**UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE FÍSICA DE SÃO CARLOS**

Marina Jeaneth Machicao Justo

**Padrões e pseudo-aleatoriedade usando sistemas
complexos**

São Carlos

2017

Marina Jeaneth Machicao Justo

**Padrões e pseudo-aleatoriedade usando sistemas
complexos**

Tese apresentada ao Programa de Pós-Graduação em Física do Instituto de Física de São Carlos da Universidade de São Paulo, para obtenção do título de Doutor em Ciências.

Área de concentração: Física Aplicada
Opção: Física Computacional

Orientador: Prof. Dr. Odemir Martinez Bruno

Versão corrigida
(Versão original disponível na Unidade que aloja o Programa)

São Carlos
2017

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Machicao, Jeaneth

Padrões e pseudo-aleatoriedade usando sistemas complexos / Jeaneth Machicao; orientador Odemir Martinez Bruno - versão corrigida -- São Carlos, 2017.
198 p.

Tese (Doutorado - Programa de Pós-Graduação em Física Aplicada Computacional) -- Instituto de Física de São Carlos, Universidade de São Paulo, 2017.

1. Padrões. 2. Pseudo-aleatoriedade. 3. Reconhecimento de padrões. 4. Sistemas complexos. 5. Teoria do caos. I. Bruno, Odemir Martinez, orient. II. Título.

Aos meus pais Alicia e Gonzalo

AGRADECIMENTOS

A Deus.

A minha mãe Alicia pelas tantas orações e pelas tantas mostras de carinho. Ao meu pai Gonzalo pelas lições de vida e por sempre ter acreditado e se orgulhado de mim. Aos meus irmãos César, Carlos, Jonathan e Fredy; pelos momentos gratos em família e sábios conselhos. Aos meus sobrinhos, Rodrigo e Catalina, por serem as novas razões para curtir da vida. Ao meu avô Miguel, por ser um exemplo de longa vida, e a toda a minha família que foi crescendo nesses últimos anos.

Com todo meu carinho, ao meu amado João por estar comigo nas boas e más, por ter se entregado a mim em todo sentido, por ter me acompanhado, ser meu ombro e bastão, por ter rido e chorado comigo todos estes anos juntos. Te amo, obrigada.

Ao meu orientador Prof. Odemir Martinez Bruno, pela orientação e mais do que isso por sempre ter acreditado em mim, mesmo quando eu não enxergava meus passos. Minha maior admiração pelas grandes qualidades como pessoa, atitudes e exemplar profissional.

Aos meus professores da graduação da Universidad Nacional de San Agustín de Arequipa, em especial ao Prof. César Beltrán Castañón e ao Prof. Juan Carlos Gutierrez, pelo apoio e incentivo a realizar este sonho.

A todos os meus professores que foram os meus verdadeiros alicerces na longa estrada inesgotável por estudar. Com especial carinho ao Prof. Javier Pariona Cardenas, por engrandecer o meu espírito de matemáticas, e a todos os docentes do Colegio Santa Rosa de Abancay.

A todos os colegas do *Scientific Computing Group*. A Mariane, Marcela, Núbia, Dalcimar, João, Wesley, Bruno, Anderson, Maurício, Marcos, Lucas, Leonardo, Humberto, Jarbas, Rayner e Alex, pelo companheirismo e ajuda em tantas dúvidas tiradas. Especialmente a minhas amigas Ângela e Gisele, pelas inúmeras vezes que me deram forças e se preocupando comigo quando fez falta.

A todos meus amigos do Peru. A Giselle, Luz Marina, Ynes, Carmen, William, Juan Herbert e Gabriel por sempre termos vivenciado gratos momentos, mesmo na distância.

Aos meus amigos brasileiros e peruanos que conheci no Brasil (Edwin, Juan, Laura, Roni, Rosita, Douglas, Paulinha e Sama), pelas reuniões, festas, e sessões de filmes com excelente culinária.

A toda a comunidade peruana em São Carlos pela acolhida afetuosa e por ser um recanto longe de casa.

Ao “alojamento VELHO-Território LIVRE” por ter me cedido um espaço quando precisei um lar.

Ao Instituto de Física de São Carlos (IFSC-USP), aos professores e funcionários. Em especial a Cristina da biblioteca e Italo da seção da gráfica, pela simpatia e ajuda nos últimos estágios de desenvolvimento deste trabalho. E a todos que, direta ou indiretamente, colaboraram neste.

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pela concessão da bolsa de Doutorado.

Por último, mas não menos importante, ao Pukis pelos gratos momentos de calma e tranquilidade que só uns bigodinhos podem oferecer.

Obrigada a todos do fundo do meu coração.

“Por vezes sentimos que aquilo que fazemos não é senão uma gota de água no mar. Mas o mar seria menor se lhe faltasse uma gota.”

Madre Teresa de Calcutá

RESUMO

MACHICAO, J. **Padrões e pseudo-aleatoriedade usando sistemas complexos.** 2017. 198p. Tese (Doutorado em Ciências) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2017.

Neste trabalho demonstramos que padrões e aleatoriedade estão intimamente relacionados, ao contrário do que intuitivamente é considerado como campos opostos. Esta abordagem visa dois propósitos: por um lado, obter vantagens das propriedades caóticas para “medir” pseudo-aleatoriedade, e por outro lado, extrair padrões de diagramas espaço-tempo como método de reconhecimento de padrões. Este trabalho centrou-se em dois métodos relacionados com sistemas complexos, como sistemas dinâmicos de tempo discreto, redes complexas, autômatos celulares (AC) e suas combinações. O primeiro método foi explorar as propriedades das profundezas do caos como fonte de pseudo-aleatoriedade a partir de sistemas dinâmicos caóticos, como o mapa logístico e o mapa da tenda. Observamos que os padrões desaparecem e a pseudo-aleatoriedade é aumentada pela remoção de k dígitos à direita da vírgula dos pontos de uma órbita original de um mapa caótico. Portanto, foi encontrada uma fonte caótica interessante para obter geradores de números de pseudo-aleatórios (PRNGs) parametrizada por k . Um segundo método foi proposto com base na incorporação de autômatos celulares na topologia de rede, também chamada de rede-autômato, visando caracterizar as redes a partir da dinâmica espaço-temporal intrínseca dessas redes. Quatro problemas de grande demanda foram explorados, tais como (i) identificar redes sociais online; (ii) identificar organismos de diferentes domínios da vida através de suas redes metabólicas; (iii) classificar padrões de distribuição de estômatos variando de acordo com diferentes condições ambientais; e (iv) o problema de identificação de autoria. Finalmente, essa mesma abordagem foi utilizada para analisar as sequências de números pseudo-aleatórios gerados pelo padrão ouro do k -mapa logístico no contexto do reconhecimento de padrões. A abordagem proposta permitiu explorar padrões e pseudo-aleatoriedade extraídos de uma miríade de sistemas com resultados bem-sucedidos em termos de acerto e boa pseudo-aleatoriedade. Além disso, este trabalho trouxe consigo progressos significativos em aplicações de reconhecimento de padrões do mundo real de um amplo ramo de campos como criptografia, criptoanálise, biologia e ciência dos dados.

Palavras-chave: Padrões. Pseudo-aleatoriedade. Reconhecimento de padrões. Sistemas complexos. Teoria do caos. Autômato celular.

ABSTRACT

MACHICAO, J. **Patterns and pseudo-randomness using complex systems.** 2017. 198p. Tese (Doutorado em Ciências) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2017.

In this work, we demonstrate that patterns and randomness are close related, contrary to what intuitively is considered as opposite fields. We aimed for a pattern recognition approach that aims for two purposes: (i) to take advantages from the chaotic properties as a source of pseudo-randomness in order to “measure” pseudo-randomness and (ii) to extract patterns from spatio-temporal diagrams obtained from complex systems models as a pattern recognition method. This work has focused on different complex systems such as discrete dynamical systems, complex networks, cellular automata (CA), and their combinations. The first method was to explore the chaotic properties in a deep-zoom manner as a source of pseudo-randomness from chaotic dynamical systems such as the logistic map and the tent map. We observed that the patterns vanish and therefore pseudo-randomness is increased by removing k right digits from the original orbit sequences. Therefore, we found an interesting chaotic source to obtain pseudo-randomness number generators (PRNGs). A second method was proposed based on the embedding of cellular automata (CA) over a network topology, also called network automata, aiming to characterize networks from the intrinsic spatio-temporal dynamics of these networks. Various on-demand problems were explored such as (i) identifying online social networks; (ii) identifying organisms from distinct domains of life through their metabolic networks; (iii) classifying stomata distribution patterns varying according to different environmental conditions; and (iv) the authorship identification problem. Finally, this same approach was used to analyze the sequences of pseudo-random numbers generated by the gold standard k -logistic map in the context of pattern recognition. So far, the proposed pattern recognition approach based on non-linear systems allowed us to explored patterns and pseudo-randomness extracted from a myriad of systems with successful results in terms of accuracy and good pseudo-randomness. The proposed method has made significant progress in real-world pattern recognition applications from a wide branch of fields such as Cryptography, Cryptanalysis, Biology and Data Science.

Keywords: Patterns. Pseudo-randomness. Pattern recognition. Complex systems. Chaos theory. Cellular automata.

LISTA DE FIGURAS

Figura 1 – Padrões simples, complexos e aleatórios gerados por sistemas complexos	27
Figura 2 – Diagrama de organização da tese	31
Figura 3 – Diagrama espaço-tempo de duas órbitas do mapa logístico	38
Figura 4 – Diagrama de fases bi- e tridimensional do mapa logístico	39
Figura 5 – Diagrama espaço-tempo, diagrama de <i>cobweb</i> , diagrama de bifurcação e curva do expoente de Lyapunov do mapa logístico para diferentes parâmetros μ	40
Figura 6 – Diagrama de fases bi- e tridimensional do mapa da tenda	41
Figura 7 – Diagrama de bifurcação do mapa da tenda	42
Figura 8 – Autômato celular elementar com regra 30 e seu respectivo diagrama espaço-tempo	45
Figura 9 – Classificação qualitativa do autômato celular elementar do Wolfram	46
Figura 10 – Exemplos de autômatos celulares da família <i>Life-Like</i>	47
Figura 11 – Exemplo da dinâmica de uma rede-autômato	49
Figura 12 – Esquema para a construção de um gerador de números pseudo-aleatórios	64
Figura 13 – Visualização de números pseudo-aleatórios gerados pelo RANDU e sua distribuição de frequências	66
Figura 14 – Histograma normalizado do mapa logístico	70
Figura 15 – A profundidade infinitesimal no caos	75
Figura 16 – Evolução temporal de duas órbitas do k -mapa logístico	80
Figura 17 – Diagrama de bifurcação e correspondente esqueleto de bifurcação do k -mapa logístico	81
Figura 18 – Diagrama de <i>cobweb</i> do k -mapa logístico	83
Figura 19 – Algoritmo de Wolf para computar o expoente de Lyapunov em séries temporais	84
Figura 20 – Expoente de Lyapunov $\lambda(k_0)$, $\lambda(k_1)$, $\lambda(k_2)$ e $\lambda(k_3)$ do k -mapa logístico	84
Figura 21 – Curvas de distribuição de frequências do k -mapa logístico	86
Figura 22 – Diagrama de Poincaré bi- e tri-dimensional do k -mapa logístico	88
Figura 23 – Diagrama de Poincaré bidimensional colorido do k -mapa logístico	89
Figura 24 – Comparação de resultados do teste DIEHARD dos quatro PRNGs: o k -mapa logístico, k -mapa da tenda, o gerador congruente linear, e o Mersenne Twister	94
Figura 25 – Espectro de potência do PRNG gerador congruencial linear RANDU e do Mersenne Twister	95
Figura 26 – Espectros de potência do PRNG k -mapa logístico	96
Figura 27 – Construção da rede-autômato <i>Life-Like</i>	101

Figura 28 – Diagrama espaço-tempo da rede-autômato <i>Life-Like</i> com regra B1357-S2468	102
Figura 29 – Distribuição das medidas de entropia de Shannon $\vec{\mu}_S$, comprimento da palavra $\vec{\mu}_W$ e complexidade de Lempel-Ziv $\vec{\mu}_L$	105
Figura 30 – Sequência de passos para o reconhecimento de padrões em redes usando a abordagem de rede-autômato LLNA.	107
Figura 31 – Distribuição das 262,144 regras da família <i>Life-Like</i> de acordo à taxa de acerto atingida para a classificação das redes sintéticas	109
Figura 32 – Análise da taxa de acerto em relação ao tempo de iterações t e das probabilidades de condições iniciais σ	110
Figura 33 – Análise das variáveis canônicas do vetor de características correspondente a 11,200 redes dos modelos: aleatório, pequeno-mundo, livre de escala e geográfica	112
Figura 34 – Análise das variáveis canônicas do vetor de características correspondente a 500 redes livre de escala do modelo Barabási-Albert para $\alpha = \{0.5, 1.0, 1.5, 2\}$ e Dorogovtsev-Mendes	114
Figura 35 – Distribuição das 262,144 regras da família <i>Life-Like</i> de acordo à taxa de acerto atingida para a classificação das redes sociais	123
Figura 36 – Análise da taxa de acerto usando o vetor de características $\vec{\mu}_L$ para as redes sociais	124
Figura 37 – Modelagem de uma rede metabólica baseado em reações enzimáticas da espécie <i>Arabidopsis thaliana</i>	125
Figura 38 – Distribuição das 262,144 regras da família <i>Life-Like</i> de acordo à taxa de acerto atingida para a classificação das redes metabólicas	128
Figura 39 – Análise das variáveis canônicas do vetor de características $\vec{\mu}_S$ correspondente às redes metabólicas	129
Figura 40 – Modelagem de uma rede estomática	133
Figura 41 – Distribuição das 262,144 regras da família <i>Life-Like</i> de acordo à taxa de acerto atingida para a classificação das redes estomáticas da espécie <i>Tradescantia zebrina</i>	135
Figura 42 – Análise das variáveis canônicas e da matriz de confusão correspondente à classificação das redes estomáticas	137
Figura 43 – Esquema de atribuição de autoria baseada no método LLNA	140
Figura 44 – Exemplo da modelagem da rede de co-ocorrência	141
Figura 45 – Distribuição das 262,144 regras da família <i>Life-Like</i> de acordo à taxa de acerto atingida para a classificação das redes textuais	142
Figura 46 – Análise dos componentes principais e matriz de confusão correspondente à classificação de redes textuais	145

Figura 47 – Construção da rede espaço-tempo a partir das órbitas geradas pelo k -mapa logístico	149
Figura 48 – Distribuição das 262,144 regras da família <i>Life-Like</i> de acordo à taxa de acerto atingida para a classificação das redes espaço-tempo do PRNG k -mapa logístico	151
Figura 49 – Distribuição do vetor de características $\vec{\mu}_L$ correspondente à classificação das redes espaço-tempo dos pares de classes do PRNG k -mapa logístico	152
Figura 50 – Construção da rede de transição do PRNG k -mapa logístico	155
Figura 51 – Visualização da matriz de probabilidades de transição	156
Figura 52 – Distribuição do vetor de características correspondente à matriz de probabilidade de transição do PRNG k -mapa logístico	156
Figura 53 – Resultados da classificação dos pares de classes k_i vs k_{i+1} -mapa logístico usando como vetor de características à matriz de probabilidade de transição	158
Figura 54 – Resumo comparativo do método proposto e as medidas estruturais de redes para as seis bases de redes estudadas	159
Figura 55 – Distribuição do espaço de regras da família <i>Life-Like</i> para as seis bases de redes estudadas	160
Figura 56 – Diagrama de bifurcação do k_0 , k_1 , k_2 e k_3 -mapa da tenda	190
Figura 57 – Diagrama de Poincaré para k_0 , k_1 , k_2 , k_3 , k_4 e k_5 usando $\delta = 2$	191

LISTA DE TABELAS

Tabela 1 – Tipos de sistemas dinâmicos.	35
Tabela 2 – Estado da arte da criptografia e a pseudo-aleatoriedade baseada na teoria do caos ao longo dos anos.	69
Tabela 3 – Resultados dos testes de aleatoriedade DIEHARD para o PRNG k -mapa logístico	91
Tabela 4 – Resultados dos testes de aleatoriedade NIST para o PRNG k -mapa logístico	92
Tabela 5 – Comparação dos resultados de testes de aleatoriedade DIEHARD dos PRNGs k_4 -mapa logístico, gerador congruente linear e Mersenne Twister	93
Tabela 6 – Diagrama espaço-tempo da rede-autômato <i>Life-Like</i> dos 4 modelos de redes sintéticas usando a regra B1357-S2468	103
Tabela 7 – Histogramas da distribuição de entropia de Shannon $\bar{\mu}_S$ para cada um dos quatro modelos de rede (aleatório, pequeno mundo, livre de escala e geográfica)	106
Tabela 8 – Bases de dados de redes sintéticas.	109
Tabela 9 – Taxa de acerto (%) obtida usando diferentes medidas e suas combinações como descritores para classificar os modelos de rede: aleatório, pequeno-mundo, livre de escala e geográfica	111
Tabela 10 – Análise da robustez do LLNA para valores diferentes de ruído ρ_N . . .	112
Tabela 11 – Taxas de acerto (%) obtida para a classificação dos 5 tipos de redes livre de escala	113
Tabela 12 – Comparação da taxa de acerto (%) para a classificação do <i>dataset-sintético</i> , <i>dataset-sintético-ruído</i> e <i>dataset-livre-escala</i> usando as medidas estruturais e o LLNA	115
Tabela 13 – Resumo de informações das bases de dados de redes apresentadas neste capítulo.	120
Tabela 14 – Taxas de acerto (%) para a classificação das redes sociais usando o LLNA	123
Tabela 15 – Comparação da taxa de acerto (%) para a classificação de redes sociais usando medidas estruturais e o LLNA	125
Tabela 16 – Lista dos 43 organismos da base de redes metabólicas divididas em três classes	127
Tabela 17 – Taxas de acerto (%) para a classificação de redes metabólicas usando o LLNA	128
Tabela 18 – Comparação da taxa de acerto (%) para a classificação das redes metabólicas usando medidas estruturais e o LLNA	129

Tabela 19 – Taxas de acerto (%) para a classificação da plasticidade fenotípica em plantas de três diferentes espécies usando o LLNA	136
Tabela 20 – Comparação da taxa de acerto (%) para a classificação da plasticidade fenotípica em plantas de diferentes espécies usando medidas estruturais e o LLNA	138
Tabela 21 – Taxas de acerto (%) obtida para classificar 8 autores usando diferentes conjuntos de lematização <i>nenhuma</i> , <i>parcial</i> e <i>completa</i> usando o LLNA	144
Tabela 22 – Comparação da taxa de acerto (%) para a identificação de autoria usando a concatenação de medidas estruturais e o LLNA	146
Tabela 23 – Taxa de acerto (%) da classificação binária de sequências geradas pelo PRNG <i>k</i> -mapa logístico usando o LLNA	151
Tabela 24 – Comparação da taxa de acerto (%) para a classificação das redes de espaço-tempo do PRNG <i>k</i> -mapa logístico usando medidas estruturais e o LLNA	153
Tabela 25 – Número de regras em comum dentro das 10,000 melhores selecionadas para a classificação das seis bases de redes.	160
Tabela 26 – Síntese das principais contribuições desta tese.	166
Tabela 27 – Descrição da bateria de testes de aleatoriedade DIEHARD.	187
Tabela 28 – Descrição da bateria de testes de aleatoriedade NIST.	188
Tabela 29 – Resultados dos testes de aleatoriedade DIEHARD para o PRNG <i>k</i> -mapa da tenda	192
Tabela 30 – Resultados dos testes de aleatoriedade DIEHARD para o PRNG <i>k</i> -mapa da tenda	193

LISTA DE ABREVIATURAS E SIGLAS

AC	Autômato celular
ACE	Autômato celular elementar
Bx-Sy	Regra para “nascer” (<i>Born</i>) e “sobreviver” (<i>Survive</i>)
BA	Barabási & Albert
CB-PRNG	<i>Chaos-based Pseudo Random Number Generator</i>
CVA	Análise de variáveis canônicas (<i>Canonical Variable Analysis</i>)
DIEHARD	Bateria de testes de aleatoriedade DIEHARD
DM	Dorogovtsev & Mendes
ER	Erdős & Renyi
GEO	Rede geográfica
kNN	Algoritmo <i>k</i> -vizinhos mais próximos (<i>k-Nearest Neighbor</i>)
LCG	Gerador congruencial linear (<i>Linear Congruential Generator</i>)
LDA	Análise discriminante linear (<i>Linear Discriminant Analysis</i>)
LE	Expoente de Lyapunov (<i>Lyapunov exponent</i>)
LLNA	Rede autômato <i>Life-Like</i> (<i>Life-Like Network Automata</i>)
MT	Gerador Mersenne Twister
NB	<i>Naive</i> Bayes
NIST	Instituto Nacional de Padrões e Tecnologia (<i>National Institute of Standards and Technology</i>)
PCA	Análise de componentes principais (<i>Principal Component Analysis</i>)
PLN	Processamento de Linguagem Natural
PRNG	Gerador de números pseudo-aleatórios (<i>Pseudorandom Number Generator</i>)
RPR	Reconhecimento de Padrões em Redes

SVM	Máquina de vetores de suporte (<i>Support Vector Machine</i>)
TRNG	Gerador de números verdadeiramente aleatórios (<i>True Random Number Generator</i>)
t-SNE	<i>t-Distributed stochastic Neighbor Embedding</i>
WS	Watts & Strogatz

LISTA DE SÍMBOLOS

S	Espaço de fase
t	Tempo
f	Sistema dinâmico
x_0	Condição inicial
x_t	t -ésima iteração
x_c	Ponto crítico
λ	Expoente de Lyapunov
μ	Parâmetro do mapa logístico
δ	Parâmetro do mapa da tenda
$\{P_n^i\}$	Esqueleto de bifurcação
$\mathcal{O}^k(\mu, x_0)$	Órbita do k -mapa logístico
x_t^k	k -dígitos à direita da vírgula de x_t
$\mathcal{C}$	Autômato celular
$\mathcal{T}$	Tesselação
c_i	Célula
s_0	Condição inicial do autômato celular
$s(c_i, t)$	Estado da célula
N	Função de vizinhança
ϕ	Função de transição ou regra
r	Raio de vizinhança
$\mathcal{G}$	Rede ou grafo
$\mathcal{V}$	Conjunto de vértices de um grafo
$\mathcal{E}$	Conjunto de arestas de um grafo
v_i	Vértice indexado por i

N	Número de vértices
M	Número de arestas
A	Matriz de adjacência
k_i	Grau ou conectividade
k_i^{in}	Grau de entrada
k_i^{out}	Grau de saída
H_{k_h}	Grau hierárquico
cc_i	Coeficiente de aglomeração
$\langle l \rangle$	Comprimento médio dos caminhos mínimos
ρ_P	Assortatividade
b_i	Intermediação
γ	Expoente da lei de potência
α	Coeficiente de ligação preferencial
$P(k)$	Distribuição de grau
$\langle \rangle$	Média
$\vec{\mathcal{F}}$	Vetor de características
$\sigma(v_i, t)$	Densidade de vizinhança
μ_S	Entropia de Shannon
μ_W	Comprimento da palavra
μ_L	Complexidade de Lempel-Ziv

SUMÁRIO

1	INTRODUÇÃO	27
2	SISTEMAS COMPLEXOS, CAOS, E RECONHECIMENTO DE PADRÕES	33
2.1	Sistemas dinâmicos não lineares	34
2.2	Teoria do caos	35
2.3	Mapas iterativos	37
2.3.1	Mapa logístico	37
2.3.2	Mapa da tenda	41
2.4	Autômato celular	43
2.4.1	Autômato celular elementar	44
2.4.2	Autômato celular da família “Life-Like”	46
2.5	Rede-autômato	48
2.6	Redes complexas	50
2.6.1	Medidas estruturais	50
2.6.2	Modelos de redes complexas	53
2.6.2.1	Rede aleatória	53
2.6.2.2	Rede pequeno-mundo	53
2.6.2.3	Rede livre de escala	54
2.6.2.4	Rede geográfica	55
2.7	Reconhecimento de padrões em redes	55
2.7.1	Modelagem de dados em redes	56
2.7.2	Extração de características	57
2.7.3	Classificação	58
2.7.3.1	Classificadores supervisionados	59
2.7.3.2	Métodos de projeção	60
3	CAOS E PSEUDO-ALEATORIEDADE	61
3.1	Geradores de números aleatórios e pseudo-aleatórios	61
3.1.1	O que é uma sequência aleatória?	61
3.1.2	Aplicações das sequências de números aleatórios	63
3.2	PRNGs clássicos	64
3.2.1	Gerador congruencial linear	65
3.2.2	RANDU: quando os PRNGs falham	65
3.2.3	Mersenne Twister	66
3.3	PRNGs baseados no caos	67

3.3.1	Desvantagens do mapa logístico	68
3.4	Testes de aleatoriedade	70
3.4.1	Limitações dos atuais testes de aleatoriedade	71
4	ALEATORIEDADE NAS PROFUNDEZAS DO CAOS	75
4.1	Mergulhando nas profundezas do caos	76
4.1.1	A sensibilidade dos dígitos menos significativos	76
4.2	k-mapa logístico	78
4.2.1	Análise da dinâmica do k -mapa logístico	79
4.2.1.1	Diagrama de evolução temporal	79
4.2.1.2	Diagrama de bifurcação	79
4.2.1.3	Diagrama de <i>cobweb</i>	82
4.2.1.4	Aproximação do Expoente de Lyapunov do k -mapa logístico	82
4.3	Propriedades pseudo-aleatórias do k-mapa logístico	85
4.3.1	Distribuição da frequência	85
4.3.2	Diagrama de Poincaré do k -mapa logístico	87
4.4	PRNG baseado no k-mapa logístico	87
4.4.1	Recomendações e limitações do PRNG k -mapa logístico	88
4.4.1.1	Análise do parâmetro de controle $\mu = 4$ e $\delta = 2$	89
4.4.2	Avaliação pela bateria de testes DIEHARD	90
4.4.3	Avaliação pela bateria de testes NIST	91
4.5	Comparação com PRNGs LCG e Mersenne Twister	93
4.5.1	Teste de densidade espectral	94
4.6	Padrão ouro PRNG $k_0, k_1, \dots, k_\infty$	97
5	PADRÕES ESPAÇO-TEMPO USANDO REDE-AUTÔMATO LLNA	99
5.1	LLNA: Rede-autômato <i>Life-Like</i>	99
5.1.1	Padrões na rede-autômato	101
5.1.2	Medidas extraídas do padrão espaço-tempo	103
5.1.3	Vetor de características	104
5.1.4	Invariância ao tamanho da rede	105
5.2	Reconhecimento de padrões em redes usando o LLNA	106
5.2.1	Dataset de redes sintéticas	106
5.2.2	Seleção da regra	108
5.2.3	Sensibilidade dos parâmetros do LLNA	109
5.2.4	Classificação dos quatro modelos de redes sintéticas	110
5.2.5	Tolerância a ruído	111
5.2.6	Classificação de modelos de redes livre de escala	112
5.2.7	Comparação com medidas estruturais de redes	113
5.3	Considerações finais	114

6	RECONHECIMENTO PADRÕES E ALEATORIEDADE EM RE-	
	DES USANDO O LLNA	117
6.1	Configurações dos experimentos	118
6.1.1	<i>Benchmark</i> de redes do mundo real	118
6.1.2	Estratégias para a seleção da regra	119
6.1.3	Estratégias para a extração de características	119
6.1.4	Estratégias de treinamento e validação	120
6.1.5	Estratégias de comparação com medidas estruturais de redes	121
6.2	Identificação de padrões estruturais em redes sociais <i>online</i>	121
6.2.1	Contextualização e aquisição de dados	121
6.2.2	Seleção de regras	122
6.2.3	Resultados da validação	122
6.2.4	Discussões	124
6.3	Identificação de organismos em redes metabólicas	125
6.3.1	Contextualização e aquisição de dados	125
6.3.2	Seleção da regra	126
6.3.3	Resultados da validação	127
6.3.4	Discussões	130
6.4	Análise da plasticidade fenotípica de plantas em redes estomáticas	130
6.4.1	Contextualização e aquisição de dados	130
6.4.2	Modelagem da rede de estômatos	131
6.4.3	Seleção da regra	134
6.4.4	Resultados da validação	134
6.4.5	Discussões	137
6.5	Reconhecimento de autoria usando redes textuais	139
6.5.1	Contextualização e aquisição de dados	139
6.5.2	Modelagem de textos como redes	141
6.5.3	Seleção da regra	142
6.5.4	Resultados da validação	143
6.5.5	Discussões	145
6.6	Classificação de sequências pseudo-aleatórias em redes espaço-tempo	147
6.6.1	Contextualização	147
6.6.2	Modelagem de séries temporais como redes espaço-tempo limiarizada	148
6.6.3	Seleção da regra	150
6.6.4	Resultados da validação	150
6.6.5	Discussões	153
6.7	Classificação de sequências pseudo-aleatórias em redes de transição	153
6.7.1	Contextualização	153
6.7.2	Modelagem de redes de transição	154

6.7.3	Resultados da validação	156
6.7.4	Discussões	157
6.8	Resumo dos resultados e discussões	157
7	CONCLUSÕES	163
7.1	Contribuições e publicações	164
7.2	Trabalhos futuros	168
	REFERÊNCIAS	169
	APÊNDICES	185
	APÊNDICE A – TESTES DE ALEATORIEDADE	187
	APÊNDICE B – K-MAPA DA TENDA	189
	APÊNDICE C – LISTA DE LIVROS E AUTORES DA BASE GU- TENBERG	195

1 INTRODUÇÃO

Físicos e matemáticos querem descobrir regularidades. Qual o uso tem [estudar] a desordem? Mas as pessoas têm que saber sobre desordem se elas vão lidar com isso. [...]
— James Yorke em entrevista com James Gleick (1, p. 64 tradução livre)

Desde o primeiro instante de nossas vidas, capturamos e processamos padrões constantemente. Aprendemos a cheirar, ouvir, sentir, ver, tocar padrões, porque de outra forma não poderíamos viver. A cada instante usamos esta informação para realizar previsões que nos permitam realizar inclusive as mais simples atividades. Graças a esta aprendizagem nossos cérebros conseguem reconhecer padrões desde os mais regulares até os mais complexos como, por exemplo, localizar um padrão imerso num meio aleatório. Apenas para aguçar nossos sentidos, o leitor é convidado a visualizar os seguintes padrões na Figura 1.

Figura 1 – Padrões simples, complexos e aleatórios gerados por diferentes regras de um a) autômato celular elementar; b) um autômato celular bidimensional *Life-Like*; c) por uma rede-autômato LLNA (embebido em uma rede aleatória, rede pequeno-mundo, rede textual e uma rede livre de escala); d) por um gerador de números pseudo-aleatórios k -mapa logístico (k_0 , k_1 , k_2 e k_3).

Fonte: Elaborada pela autora.

Sim, são padrões, desde os mais simples ou complexos até os mais aleatórios! Neles a complexidade está relacionada ao comportamento macroscópico do sistema, cujos elementos

interagem entre si e seu ambiente, exibindo equilíbrio entre a ordem e a aleatoriedade, no qual a complexidade surge no meio-termo, muitas vezes na fronteira da ordem-desordem. (2) Padrões e aleatoriedade não são uma oposição binária (presença ou ausência), uma vez que a aleatoriedade não é a ausência de padrão, mas sim a base para os padrões emergirem (3), pois muitos fenômenos naturais apresentam um certo equilíbrio entre o regular e a desordem convivendo e interagindo continuamente. (2) Embora exista uma linha tênue e difusa entre padrão e aleatoriedade, estes termos são complementares, não obstante estudados por separado.

Da mesma forma que o ser humano é uma máquina poderosa para detectar padrões e fazer previsões, computacionalmente a área de reconhecimento de padrões busca por métodos que consigam detectar padrões que permitam caracterizar objetos abstratos a fim de distingui-los dentre outras entidades de forma automatizada. Porém, é razoável que o leitor não tenha encontrado padrões visíveis nas imagens produzidas pelo gerador de números pseudo-aleatórios (PRNG). De fato, como será visto no decorrer desta tese, métodos de reconhecimento de padrões também podem ser utilizados para explorar padrões complexos e inclusive aleatórios.

A maior motivação para estudar os padrões num contexto de aleatoriedade se dá porque o que parece se comportar desordenadamente pode estar governado por uma regra geral, lei ou função que dite tal comportamento, que ainda não foi descoberto. Entretanto, especula-se da sua aleatoriedade até ser comprovado o contrário. De fato, na área da criptoanálise, que é uma área com muitos desafios a serem superados, existe uma alta demanda por métodos que ajudem a detectar certos tipos de fraquezas (padrões). Por exemplo, um PRNG deve gerar sequências sem vestígios de padrões, pois, isto representaria uma falha de segurança. Visto que, recentemente, durante o mestrado da autora, foi observado que tanto o reconhecimento de padrões como a criptoanálise possuem objetivos complementares, obtiveram-se resultados importantes que indicavam a presença de padrões nos algoritmos criptográficos. (4,5) Esta linha de pesquisa ainda é considerada pouco explorada, mas este fato tornou-se uma das motivações desta tese.

Na literatura existem três lacunas que podem ser vistas como complementares. A primeira diz que métodos de reconhecimento de padrões são pouco explorados para analisar sequências pseudo-aleatórias, pois isto representa um desafio, inclusive, esdrúxulo. A segunda, que não existe um padrão ouro da pseudo-aleatoriedade, ou seja, uma forma de avaliá-la entre mais “fraca” até a “mais forte”, já que, na literatura, apesar das presunções existentes sobre as fraquezas dos PRNGs, não existe ainda uma forma de parametrizar tal qualidade. A terceira, se existisse um padrão ouro, logo em teoria, abriria a possibilidade de usar os métodos de reconhecimento de padrões para ser possível estudar PRNGs com o fim de “medir a pseudo-aleatoriedade” em tais sistemas. Assim, em um primeiro contexto, esses novos métodos trariam contribuições na detecção de padrões, assim como o mesmo

método poderia ser usado como um teste de aleatoriedade. Já que tipicamente, os testes mais usados na comunidade são de natureza empírica (estatísticos), os quais são usados para analisar os padrões de distribuição, contudo, apenas como indicadores baseados em hipóteses de “passou” ou “falhou”. Assim sendo de grande importância novos métodos que busquem revelar a presença de distribuições não óbvias em sequências de números produzidos por um gerador.

O objetivo desta tese é propor métodos para evidenciar esta dicotomia: padrões e aleatoriedade. Assim, encontramos na teoria dos sistemas complexos e na teoria do caos as bases matemáticas para contornar estes problemas. Os sistemas dinâmicos não-lineares fornecem os modelos matemáticos para a análise de sistemas complexos, como as equações diferenciais, mapas iterativos, autômatos celulares, redes complexas, entre outras. (6) Estes modelos buscam caracterizar as estruturas subjacentes e a complexidade durante a evolução temporal de um sistema. O mapa logístico é uma das equações de recorrência mais estudadas na literatura, comumente usado como uma referência para examinar os padrões característicos no espaço de fase como, por exemplo, da cascata de bifurcações exibida pelas trajetórias na rota ao caos. (6–8)

No entanto, atualmente, com a introdução da ciência dos dados e *big data*, existe uma alta demanda por métodos de reconhecimento de padrões que tratem com grandes volumes de dados. O principal objetivo do reconhecimento de padrões é permitir a extração automatizada de representações dos dados. (9) Consequentemente, é necessário o desenvolvimento de um paradigma integrado de dados. A combinação do reconhecimento de padrões e redes surge como uma alternativa importante no cenário de *big data* para encontrar, identificar, analisar e agrupar padrões que, de outra forma, seriam inviáveis por meio da abordagem reducionista.

Em contraposição à abordagem reducionista, as redes têm sido usadas com sucesso em muitas áreas de conhecimento que cobrem praticamente todos os campos da Ciência. (10–13) A importância da modelagem em redes é devido a que estas incorporam mais informações na sua estrutura do que simplesmente um conjunto de dados tabelados ou vetoriais. Típicos exemplos de modelagens de dados com redes incluem as redes metabólicas, redes de interações protéicas e sociais, entre outras (14), porém, nas últimas décadas, modelos não estruturados como são séries temporais (15,16), imagens de contornos (17), texturas (18), texturas dinâmicas (19), entre outros, vem conseguindo modelar suas relações como redes. A organização de todos esses sistemas podem ser representados por grafos, isto é, vértices conectados por arestas.

O objetivo principal deste trabalho é demonstrar a presença e a ausência de padrões em sistemas complexos, bem como usar os sistemas complexos para encontrar padrões e analisar a qualidade da aleatoriedade. Mais especificamente, os objetivos podem ser divididos em: (i) Verificar a possibilidade de sistemas caóticos serem bons geradores de

seqüências pseudo-aleatórias; (ii) Propor geradores aleatórios de boa qualidade baseados em sistemas caóticos; (iii) Uma vez que o caos pode ser ajustado por parâmetros em sistemas caóticos (por exemplo, parâmetros do sistema podem produzir seqüências com expoentes de Lyapunov maiores ou menores), os geradores baseados em sistemas caóticos podem da mesma forma possuir parâmetros de controle de aleatoriedade o que pode produzir padrões ouro para o reconhecimento de padrões. Estes padrões ouro podem ser utilizados na pesquisa e desenvolvimento de métodos de reconhecimento de padrões; (iv) Desenvolver métodos de reconhecimento de padrões utilizando sistemas não lineares. O objetivo é combinar métodos não lineares advindos dos sistemas complexos para o desenvolvimento de métodos de reconhecimento de padrões, como é usar redes complexas, que são flexíveis o suficiente para trabalhar com diferentes tipos de dados, combinadas com autômatos celulares. Deste modo, as redes são usadas como tesselações dos autômatos, e durante sua evolução produzem uma dinâmica espaço temporal que é capaz de revelar padrões presentes nas redes; e (v) Investigar o uso dos métodos de reconhecimento de padrões produzidos neste trabalho com a finalidade de procurar por padrões em séries aleatórias, a fim de avaliar sua qualidade.

Este documento está organizado em 7 capítulos incluindo esta introdução. No Capítulo 2, são apresentadas as bases teóricas dos modelos de sistemas complexos, incluindo a teoria do caos, sistemas dinâmicos, mapas caóticos, autômatos celulares, redes complexas e sua combinação rede-autômatos. No Capítulo 3 são introduzidos os conceitos de geradores de números aleatórios e sua importância; Adicionalmente é realizado um levantamento bibliográfico da literatura dos métodos que aproveitam as propriedades pseudo-aleatórias dos sistemas caóticos. No Capítulo 4 é proposto uma abordagem de exploração nas profundezas do caos na busca por pseudo-aleatoriedade, na qual é feita uma análise da profundidade infinitesimal das órbitas do mapa logístico e proposto o k -mapa logístico. Nesse mesmo capítulo é descrito um gerador de números pseudo-aleatórios parametrizado (padrão ouro). No Capítulo 5 é proposto um modelo de autômato celular cuja tesselação é descrito pela topologia de uma rede, na qual foram analisados, desta maneira, os aspectos dinâmicos de um autômato celular do ponto de vista da formação de padrões. Nesse mesmo capítulo, a proposta é aplicada ao problema de classificação de redes sintéticas, com o intuito de calibrar os parâmetros. No Capítulo 6, são apresentados vários exemplos de aplicações de reconhecimento de padrões em redes, finalizando com uma aplicação na qual é usado o padrão ouro proposto a fim de distinguir entre qualidades de pseudo-aleatoriedade. No Capítulo 7 são apresentadas as conclusões deste trabalho.

Visto que esta tese está fortemente embasada nos sistemas complexos e na teoria do caos, é conveniente também apresentar uma forma de enxergar a organização deste documento por meio de um diagrama de bifurcação, como ilustrado na Figura 2.

Figura 2 – Diagrama de organização desta tese. Cada bifurcação representa os capítulos a seguir e a linha de raciocínio.

Fonte: Elaborada pela autora.

2 SISTEMAS COMPLEXOS, CAOS, E RECONHECIMENTO DE PADRÕES

Hoje em dia, o estudo dos sistemas complexos vem sendo entendido como um novo paradigma em contraposição ao clássico modelo reducionista, que tenta explicar os sistemas em função de suas partes constituintes. Muitos exemplos de sistemas complexos podem ser encontrados em quase todos os fenômenos da natureza, assim como também muitos modelos foram propostos para tentar estudá-los. Por exemplo, a equação logística surgiu como um modelo para descrever o crescimento populacional de insetos (20); já uma simples regra de autômato celular assemelha-se à formação do padrão de pigmentação das conchas da espécie *conus textile* (21); as interações sociais, pelo qual certos indivíduos são mais propensos a terem amizades com outras pessoas (22) ou as relações metabólicas estabelecidas por determinadas enzimas, que constituem uma funcionalidade metabólica de um organismo, podem ser modeladas por redes complexas. (23)

De fato, devido a sua capacidade de generalização e caráter multidisciplinar, as redes complexas, advindas das bases matemáticas da teoria de grafos e da mecânica estatística (24,25), têm sido usadas para modelar estruturas encontradas no mundo real, como, por exemplo, reações metabólicas, cadeias de proteínas, dados genéticos, dentre outros fenômenos, tornando-se ferramentas ubíquas. (26)

Todos esses exemplos de sistemas complexos são coleções de indivíduos que se interconectam e interagem entre si, a partir da qual emerge uma dinâmica coletiva e auto-organizada como consequência das relações não lineares entre seus indivíduos. (6) Um outro ponto em comum, é que os sistemas complexos podem ser modelados por sistemas dinâmicos determinísticos não lineares, e que além disso, como o leitor poderá perceber no decorrer desta tese, muitas vezes, a partir destes modelos é que surge o caos.

A teoria do caos estuda os sistemas não lineares e determinísticos que são altamente sensíveis às condições iniciais, isto é, quando até a mais pequena perturbação no sistema pode levar a resultados divergentes, tornando-os imprevisíveis e até aparentemente aleatórios. Em contraste, é curioso apontar que, a teoria do caos nasceu a partir da convergência dos diferentes estudos de cientistas das mais diversas áreas, como Henri Poincaré, Edward Lorenz (27), Robert May (20), Aleksandr Lyapunov, Mitchell Feigenbaum, Tien-Yien Li e James Yorke (28), Robert Devaney (7), Benoit Mandelbrot (29), entre muitos outros. Em virtude dessa unificação multidisciplinar, hoje em dia a teoria do caos vem sendo aproveitada por diversos campos como a física, a matemática, as ciências da computação, a economia, as engenharias, a meteorologia, a filosofia, a política e inclusive a arte.

Nesse contexto, todas as propriedades intrínsecas que trazem consigo os diversos modelos de sistemas complexos e da teoria do caos são parte da nossa motivação para

estudá-los e usá-los, pois, em princípio, se estes modelos levam a formação de padrões complexos e/ou até caóticos, também teriam propriedades aproveitáveis para tornar-se em fontes de pseudo-aleatoriedade (Capítulo 4), ou que as redes complexas sejam o hábitat no qual a vida artificial seja simulada e que a combinação de redes e autômatos celulares sejam ferramentas para o reconhecimento de padrões (Capítulo 5), ou que as sequências pseudo-aleatórias geradas pelo caos possam ser analisados sob visão do reconhecimento de padrões (Capítulo 6).

Neste capítulo são introduzidos vários modelos de sistemas complexos. Na primeira parte são dados os conceitos teóricos de sistemas dinâmicos e da teoria do caos, dando ênfases nos mapas iterativos (Seção 2.3). Já na segunda parte são introduzidos outros modelos, como os autômatos celulares (Seção 2.4) e rede-autômatos (Seção 2.5). Em seguida, são introduzidas as redes complexas (Seção 2.6), modelos clássicos de redes e medidas estruturais. Na última parte do capítulo (Seção 2.7) é tratado o reconhecimento de padrões em redes, bem como as principais técnicas e métodos de aprendizado de máquinas.

2.1 Sistemas dinâmicos não lineares

Em geral, um **sistema dinâmico não linear** $f : \mathbf{S} \rightarrow \mathbf{S}$ é uma representação de um sistema que evolui no tempo, que consiste de uma ou várias funções determinísticas não lineares que descrevem sua evolução (6), na qual:

- $\mathbf{S}$ é o **espaço de fase** (ou espaço de estados), cujos elementos representam todos os possíveis estados do sistema;
- t é o **tempo** ou número de iterações de evolução do sistema;
- f é a função, ou lei de evolução, que permite obter os estados futuros a partir dos estados anteriores de forma **determinística**;
- Dizemos que $\mathbf{S}$ é **invariante**, se todos os pontos do espaço de fase, após infinitas iterações, permanecerão em $\mathbf{S}$. Ou seja $f(\mathbf{S}) \subseteq \mathbf{S}$ é satisfeita;
- se f é dito **ergódico**, quando em média o comportamento ao longo do tempo é igual à média do correspondente espaço de fase, para órbitas típicas, conhecido como o teorema de Birkhoff.
- A **órbita** é a sequência de pontos espaço-tempo $\mathcal{O}(x_0) = \{x_0, x_1, \dots, x_t\}$ gerada por f a partir de uma **condição inicial** $x_0 \in \mathbf{S}$;
- x_0 é chamado de **ponto fixo** de f , se $f(x_0) = x_0$;
- x_0 é um **ponto periódico** de f se após iterado $p \geq 1$ vezes, satisfaz $f^p(x_0) = x_0$, com p o **período** da órbita.

De acordo com a variável tempo, os sistemas dinâmicos podem ser divididos em duas grandes categorias: **contínuo** ou **discreto**. No primeiro caso, a dinâmica do sistema é descrita por equações diferenciais. Já no segundo caso, é descrita por mapas iterativos da forma

$$x_{t+1} = f(x_t),$$

com $x_t \in \mathbf{S}$. Em ambos os casos, o espaço de fase é contínuo. Entretanto, também existem outros modelos de sistemas dinâmicos, como os autômatos celulares, cujo espaço de fase é discreto. Essa classificação de sistemas dinâmicos é resumida na Tabela 1. Note-se que neste trabalho, focamos especificamente nos mapas iterativos e nos autômatos celulares, os quais serão descritos neste capítulo.

Tabela 1 – Tipos de sistemas dinâmicos.

Classe	Dimensão do espaço de fase	Tipo de variável	Tempo
Eq. diferencial parcial	infinito	contínua	contínuo
Eq. diferencial ordinária	finito	contínua	contínuo
Mapas iterativos	finito	contínua	discreto
Autômato celular	finito	discreto	discreto

Fonte: Adaptada de FROYLAND. (8)

2.2 Teoria do caos

A teoria do caos nasceu por acaso. Nos seus primórdios, no ano 1963, Edward Lorenz (27) tentava fazer simulações meteorológicas usando um conjunto de equações. Assim, ele se deparou com a imprevisibilidade provocada pelos pequeníssimos erros computacionais que, sem perceber, faziam com que as trajetórias mudassem radicalmente do esperado. Por sua parte, o biólogo Robert May (20) também narrou sua experiência com a imprevisibilidade da equação logística (usada para simular populações de insetos), que para certos parâmetros desta, surgiam oscilações com aparência aleatória. (1) Entretanto, estas experiências eram bem conhecidas como o “efeito borboleta”, isto é, um simples bater de asas de uma borboleta poderia significar um tornado em outro continente.

Porém, foi somente no ano 1975, que o termo **caos** apareceu pela primeira vez em um trabalho apresentado por Li-Yorke (28), “período 3 implica caos” (ver Seção 2.3). Eles mostraram que para uma classe de sistemas dinâmicos unidimensionais com um ponto de máximo, como é o caso do mapa logístico, a existência uma órbita com período 3, é suficiente para se deduzir que o sistema apresente um incontável conjunto de órbitas de diversos períodos. (28) Hoje em dia, embora, ainda não haja muito consenso para a definição matemática do caos (7, 28, 30–32), uma das definições mais aceitas pela comunidade foi dada por Devaney (7), que observou três propriedades que caracterizam um sistema caótico:

Definição 1. Seja um mapa contínuo $f : \mathbf{S} \rightarrow \mathbf{S}$

1. f é **topologicamente transitiva** em $\mathbf{S}$ se, para quaisquer dois intervalos abertos I e J (que podem ser arbitrariamente pequenos, porém com comprimento não nulo), pode-se encontrar valores iniciais em I que quando iterados, acabará por levar a pontos em J (7, 33);
2. **Os pontos periódicos de f são densos** em $\mathbf{S}$, o que implica que cada ponto do sistema caótico é arbitrariamente próximo de algum ponto em uma órbita periódica. (7) O qual também implica que o sistema apresentará órbitas com periodicidade de todos os períodos, bem como um incontável conjunto de órbitas não periódicas e instáveis (28, 30);
3. f é **sensível às condições iniciais**, se existe $\varepsilon > 0$ tal que, para qualquer $x \in \mathbf{S}$ e qualquer ε vizinhança $\mathbf{W}$ de x , existem $y \in \mathbf{W}$ e $t > 0$ tais que

$$\|f^t(x) - f^t(y)\| > \varepsilon.$$

Ou seja, que a menor mudança nas condições iniciais levará a órbitas com divergência maior do que aquela com que foi perturbada. (7)

Particularmente para o caso dos sistemas dinâmicos unidimensionais, a sensibilidade às condições iniciais já estaria implícita com a primeira e segunda propriedades de Devaney. (34) Entretanto, muitos pesquisadores também advertem acerca da existência de outras propriedades que aparecem como consequências dos sistemas caóticos que se sobrepõem como extensão à definição de caos de Devaney, como por exemplo, os **atratores estranhos** (35) que podem apresentar sensibilidade às condições iniciais e são **fractais** e propriedades dinâmicas básicas que levam aos como da repetição do processo de **extensão e dobradura**, semelhante à “massa do padeiro”. (33, p. 496) Além disso, muitos pesquisadores, indicam que os sistemas caóticos devem possuir **expoentes de Lyapunov** positivos, $\lambda(x_0) > 0$. (36) O expoente de Lyapunov (*Lyapunov exponent*, LE) mede a taxa de divergência:

$$\|f^t(x + \varepsilon) - f^t(x)\| \approx \varepsilon e^{t\lambda(x_0)}, \quad (2.1)$$

com que duas órbitas de condições iniciais muito próximas ε , se afastam exponencialmente no espaço de fase após t iterações. (35, 36) O LE para sistemas dinâmicos contínuos e unidimensionais é definido por

$$\lambda(x_0) = \lim_{t \rightarrow \infty} \lim_{\varepsilon \rightarrow 0} \frac{1}{t} \ln \left\| \frac{f^t(x_0 + \varepsilon) - f^t(x_0)}{\varepsilon} \right\|. \quad (2.2)$$

Por outro lado, quando trata-se de sistemas dinâmicos discretos, como por exemplo, no caso dos mapas iterativos, o LE para sistemas unidimensionais é definido pela Equação (2.3). (6)

$$\lambda(x_0) = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{i=0}^{t-1} \ln \|f'(x_i)\|. \quad (2.3)$$

De forma similar, existem também na literatura outras aproximações para medir a sensibilidade às condições iniciais quando trata-se de sistemas de autômatos celulares (37), séries temporais (38), e inclusive, aproximações aplicadas à Criptoanálise. (5) Com o expoente de Lyapunov, é também possível distinguir dentre três tipos de comportamentos:

- Marginalmente estável ($\lambda(x_0) = 0$), quando as órbitas estão perto de uma bifurcação;
- Periódico e assintoticamente estável ($\lambda(x_0) < 0$), quando a órbita converge a uma órbita periódica ou a um ponto fixo;
- Caótico ($\lambda(x_0) > 0$), como detalhado anteriormente.

Por outro lado, nos sistemas não lineares também são observadas outras características como a **universalidade** e a **auto-similaridade**. No que tange à universalidade, é dito que todos os mapas unidimensionais quadráticos possuem uma constante de Feigenbaum $\delta \approx 4.66920$ (39), a qual mede a razão entre os intervalos sucessivos das bifurcações das órbitas periódicas para parâmetros que precedem ao regime caótico (ver Seção 2.3). Já a propriedade de auto-similaridade indica que os sistemas possuem padrões que prevalecem ao serem visualizados em sucessivas ampliações.

2.3 Mapas iterativos

2.3.1 Mapa logístico

O mapa logístico é um dos modelos matemáticos mais estudados no campo dos sistemas complexos assim como na teoria do caos. Originalmente, a equação logística (ou modelo de Verhulst), foi introduzida como um modelo de crescimento populacional por Pierre-François Verhulst, no ano 1845. Entretanto, somente no ano de 1976, que as propriedades caóticas do mapa logístico foram descobertas pelo biólogo matemático Robert May. (20)

Os **mapas iterativos**, equações de recorrência, ou simplesmente mapas, são sistemas dinâmicos não-lineares da forma $x_{t+1} = f(x_t)$. No modelo da dinâmica de populações do mapa logístico, x_t representa a população na t -ésima geração. Este mapa exhibe uma gama surpreendente de comportamentos, uma vez que o parâmetro da taxa de crescimento

μ é variado. O mapa logístico é representado pela Equação (2.4), cujo espaço de fase é definido em $f : [0, 1] \rightarrow [0, 1]$, para $\mu \in [0, 4]$.

$$x_{t+1} = \mu x_t(1 - x_t). \quad (2.4)$$

A Figura 3 apresenta a evolução no espaço de fase do mapa logístico para duas condições iniciais x_0 e x'_0 com diferentes valores μ . Na Figura 3a observa-se que inicialmente ambas órbitas seguem a mesma trajetória, porém, poucas iterações depois, ambas as órbitas divergem em razão da divergência exponencial, também conhecido como expoente de Lyapunov para tempo finito, é dada pela Equação (2.5).

$$\frac{1}{t} \sum_{i=1}^t \ln |\mu - 2\mu x_i|. \quad (2.5)$$

No entanto, na Figura 3b observa-se que logo após poucas iterações, as órbitas com parâmetro $\mu = 2.85$ convergem para um ponto fixo, o que corresponde a um expoente de Lyapunov $\lambda = \ln |2 - \mu| = \ln(0.85)$. Por outro lado, na Figura 3c, o parâmetro $\mu = 3.57$ apresenta regime periódico correspondente ao LE $\lambda \approx 0$.

Figura 3 – Diagrama espaço-tempo de duas órbitas do mapa logístico (Equação (2.4)) com condições iniciais $x_0 = 0.8$ e $x'_0 = 0.80000001$, para os parâmetros (a) $\mu = 4$, caótico; (b) $\mu = 2.85$, ponto fixo; e (c) $\mu = 3.57$ periódico, durante 100 e 50 iterações, respectivamente.

Fonte: Elaborada pela autora.

Por meio do **diagrama de fases** ou diagrama de Poincaré é possível visualizar a relação x_t e x_{t+1} de um sistema dinâmico. Da esquerda a direita, a Figura 4 apresenta o diagrama de fases bi- e tridimensional do mapa logístico para diferentes parâmetros μ . Observe que, devido a que o mapa logístico é correspondente a uma equação quadrática, para qualquer condição inicial em $[0, 1]$, o padrão característico da parábola invertida permanecerá em função de μ . Além disso, é interessante notar que, a parábola atinge

um valor máximo em $x_c = 0.5$ (**ponto crítico**), cujo padrão permanece no esboço tridimensional.

Figura 4 – Diagrama de fases bidimensional (esquerda) e tridimensional (direita) do mapa logístico usando parâmetros entre $\mu \in [3.6, 4]$ (escala colorida) para 5000 iterações. Os eixos horizontal e vertical mostram o espaço de fase x_t com respeito a x_{t+1} (esquerda), e x_t em relação a x_{t+1} e x_{t+2} (direita), respectivamente.

Fonte: Elaborada pela autora.

O **diagrama de cobweb** ou teia de aranha, é um gráfico utilizado para auxiliar na visualização da estabilidade ou instabilidade de um sistema dinâmico dado um parâmetro μ . Este diagrama é construído no plano x_t, x_{t+1} . Assim, dada uma condição inicial x_0 no eixo X, traça-se uma linha vertical a partir de x_0 até se encontrar com $f(x_0)$ e, em seguida, traçando uma linha horizontal desde o ponto $(x_0, f(x_0))$ até a função identidade $f(x) = x$, concluindo uma iteração, e assim sucessivamente. Desta forma pode ser obtida uma órbita da forma $x_0, f(x_0), f^2(x_0), \dots, f^t(x_0)$. Note-se que a linha reta $f(x) = x$ representa os pontos de interseção do gráfico da função, os quais também indicam a periodicidade do sistema dinâmico.

Na Figura 5 pode ser observado o diagrama de *cobweb* correspondente ao mapa logístico para diferentes parâmetros μ . Observe que as setas (azul) são usadas para seguir a sequência de pontos indo de x_t até x_{t+1} , o qual mantém-se dentro dos parâmetros da parábola invertida, razão pela qual é topologicamente transitiva. A Figura 5a exibe uma órbita assintoticamente estável que logo atinge um ponto fixo após um curto tempo. Por outro lado, na Figura 5b-e são apresentadas órbitas periódicas. Observe que, o número de vezes em que as órbitas intersectam a diagonal correspondem a um ciclo. Já na Figura 5c, é mostrado o regime caótico, neste caso $\mu = 4$, em que o topo da parábola atinge o ponto máximo do espaço de fase.

Uma outra forma de analisar a ergodicidade do sistema dinâmico é por meio do **diagrama de bifurcação**. Esta representação gráfica permite analisar o comportamento global das órbitas do sistema quando o parâmetro μ varia (eixo horizontal) com respeito aos possíveis valores de x_t (eixo vertical). A Figura 5g apresenta o diagrama de bifurcação

Figura 5 – Compilação de gráficos da dinâmica do mapa logístico para diferentes parâmetros μ . De cima para baixo, (a-f) diagrama espaço-tempo e correspondentes diagramas de *cobweb*. Da esquerda para a direita, são mostrados os parâmetros de controle (a) $\mu = 2.4$ (atrator estável), (b-e) $\mu = 3.2$, $\mu = 3.4495$, $\mu = 3.5441$, $\mu = 3.8284$ (atratores periódicos) e (f) $\mu = 4$ (atrator caótico), respectivamente. As setas e linhas (azul) são as seqüências de pontos indo de x_t até x_{t+1} . A parábola invertida da função logística é representada pelo parâmetro μ nas coordenadas $(x, f(x))$ (cinza), e a linha diagonal é a interseção do gráfico em (x, x) (cinza). (g-h) Diagrama de bifurcação e correspondente zoom. No eixo horizontal, os parâmetros $\mu \in [2.4, 4]$ (acima) e $\mu \in [3.847, 3.857]$ (abaixo), com mesma condição inicial $x_0 = 0.4679$. No eixo vertical, $x_t \in$ (g) $[0, 1]$ e (h) $[0.13, 0.18]$, respectivamente. (i) Curva do expoente de Lyapunov em função do parâmetro μ , valores $\lambda < 0$ (periódico), $\lambda \approx 0$ (estável) e $\lambda > 0$ (caótico). Para os gráficos (g-i) foram consideradas $t = 10^5$ iterações e desconsideradas as primeiras 200 (transiente).

Fonte: Adaptada de KING. (40)

do mapa logístico para $\mu = [2.4, 4]$. Analisando este diagrama, existe uma região trivial entre $0 < \mu < 1$ (não é mostrada na figura) em que as órbitas convergem para zero (ponto fixo); já na região, $1 < \mu < 3$, as órbitas convergem para $\frac{\mu-1}{\mu}$. Entretanto, a partir de $\mu > 3$, surgem as bifurcações (dobramento de período), as quais ocorrem quando uma pequena perturbação feita para certos valores dos parâmetros provocam uma mudança súbita no seu comportamento.

Por meio deste diagrama, no mapa logístico, é facilmente observado que existe

uma sequência de bifurcações levando à criação de órbitas com diferentes períodos $p = \{2, 4, 8, \dots\}$, que vão se bifurcando sucessivamente até o surgimento do caos em $\mu \approx 3.57$. Para um parâmetro um pouco maior aparece uma órbita com período $p = 3$ em $\mu = 1 + 2\sqrt{2} \approx 3.8284$. Logo depois surgem intermitências de caos, como por exemplo na região entre $\mu \approx [3.82, 3.86]$. Até que finalmente chegam no máximo regime caótico quando $\mu = 4$, com correspondente expoente de Lyapunov $\lambda(x_0) \approx 0.6931$ (Equação (2.5)).

Ainda mais interessante é que por meio deste diagrama é possível visualizar uma outra propriedade do caos, a auto-similaridade. Ao fazermos um zoom em torno de uma pequeníssima janela, como observado na Figura 5i, o padrão global parece repetir-se novamente. Adicionalmente, a curva do expoente de Lyapunov pode ser observada na Figura 5h, a qual corresponde ao diagrama de bifurcação da Figura 5g. Além disso, as cores das órbitas estão relacionadas com os diagrama de *cobweb* (Figura 5a-f). Observe que a periodicidade dessas órbitas correspondem ao número de ramificações no diagrama de bifurcação.

2.3.2 Mapa da tenda

O mapa da tenda é dado pela Equação (2.6), cujo espaço de fase é definido em $f : [0, 1] \rightarrow [0, 1]$, para $\delta \in [0, 2]$ como parâmetro.

$$x_{t+1} = f(\delta, x_t) = \begin{cases} \delta x_t, & \text{se } x_t < \frac{1}{2} \\ \delta(1 - x_t), & \text{se } x_t \geq \frac{1}{2} \end{cases} \quad (2.6)$$

O mapa da tenda, recebe este nome devido à forma de tenda, a qual pode ser observada no diagrama de fase da Figura 6 para diferentes parâmetros δ .

Figura 6 – Diagrama de fases bidimensional (esquerda) e tridimensional (direita) do mapa da tenda (Equação (2.6)) usando parâmetros entre $\delta \in [1.5, 2]$ (escala colorida) para 5000 iterações. Os eixos horizontal e vertical mostram o espaço de fase x_t com respeito a x_{t+1} (esquerda), e x_t em relação a x_{t+1} e x_{t+2} (direita).

Fonte: Elaborada pela autora.

De forma similar ao mapa logístico, as propriedades dinâmicas globais do mapa da tenda podem ser observadas pelo diagrama de bifurcação (ver Figura 7). Nele podemos

constatar o fenômeno do “período 3 implica caos”, a auto-similaridade, e que sua região mais caótica é dada quando $\delta = 2$ (Equação 2.3).

Figura 7 – Diagrama de bifurcação do mapa da tenda. O eixo horizontal apresenta o parâmetro $\delta \in [1, 2]$. O eixo vertical $x_t \in [0, 1]$ apresenta as evoluções a longo prazo para a mesma condição inicial. O sistema foi iterado por 10^5 iterações e descartadas as 200 primeiras (tempo transiente).

Fonte: Elaborada pela autora.

Um outro recurso interessante do diagrama de bifurcação é o esqueleto deste. Como visto nos diagramas correspondentes ao mapa logístico (Fig. 5g-h) e o mapa da tenda (Fig. 7), existem curvas mais escuras e notórias, as quais são chamadas de **esqueleto do diagrama de bifurcação**. Bai-lin (41) introduziu a família de funções recursivas $\{P_n(\mu)\}$, em que $n = 0, 1, \dots$ corresponde à curva do esqueleto de um sistema dinâmico discreto, mostrada em um diagrama de bifurcação, dada pela seguinte definição.

Definição 2. Considerando que um sistema dinâmico discreto pode ter um ou mais pontos críticos x_{c_i} , ou seja, onde sua primeira derivada desaparece ($f'(\mu, x_{c_i}) = 0$), várias funções compostas $\{P_n^i\}$ podem ser obtidas, onde o sobrescrito i indica o ponto relacionado. Então, começando de um ponto crítico x_c , é definido recursivamente por

$$P_{n+1} = f(\mu, P_n(\mu)). \quad (2.7)$$

A respeito do mapa logístico, dado seu único ponto crítico $x_c = 0.5$ e considerando que sua primeira derivada é $f'(\mu, x_c) = \mu - 2\mu x = 0$, a função P_n é polinomial de μ de forma que:

$$\begin{aligned}
P_0(\mu) &= x_c \\
P_1(\mu) &= \mu/4 \\
P_2(\mu) &= 4\mu^2 - \mu^3/16 \\
P_3(\mu) &= (\mu^3 - \mu^4)(\mu^3 - 4\mu^2 + 16)/256 \\
P_4(\mu) &= (4\mu^4 - \mu^5)(\mu^3 - 4\mu^2 + 16) \\
&\quad (\mu^7 - 8\mu^6 + 16\mu^5 + 16\mu^4 - 64\mu^3 + 256)/65536
\end{aligned}$$

Além disso, a respeito do mapa da tenda e seu único ponto crítico $x_c = 0.5$, os polinomiais $P_n(\delta)$ são:

$$\begin{aligned}
P_0(\delta) &= x_c \\
P_1(\delta) &= \delta/2 \\
P_2(\delta) &= \begin{cases} \delta^2/2, & \text{for } \delta < 1 \\ \delta(1 - \delta)/2, & \text{for } \delta \geq 1 \end{cases}
\end{aligned}$$

2.4 Autômato celular

Na seção anterior foram introduzidos sistemas dinâmicos modelados por mapas iterativos com um número finito de dimensões (e.g. mapa logístico e mapa da tenda, são unidimensionais) com tempo discreto, porém, com variáveis no espaço de fase contínuo. Entretanto, o autômato celular (AC) é um outro modelo de sistema dinâmico, ainda mais simples que os mapas, cujo espaço de fase e o tempo são representações discretas. (8)

Os ACs foram inspirados no conceito de “vida artificial”, no qual seus componentes (células) interagem entre si e seu meio ambiente, simulando vida ou morte das células. Estes estados são modificados ao longo do tempo de acordo com uma regra, dependendo apenas dos estados prévios da sua vizinhança, a partir dos quais padrões complexos podem emergir. (42)

Formalmente, um autômato celular $\mathcal{C}$ é definido pela forma generalizada (37)

$$\mathcal{C} = \langle \mathcal{T}, \mathbf{S}, s_0, \mathbf{N}, \phi \rangle,$$

em que,

- A **tesselação** $\mathcal{T}$ é composta por um conjunto de células c_i . Em geral, ACs são representados em tesselações regulares em espaços euclidianos n-dimensionais $\mathbb{Z}^n$;
- O espaço de fase $\mathbf{S}$ é formado por um conjunto finito de **estados**. Por exemplo, uma célula pode ter estados binários $\mathbf{S} = \{0, 1\}$, em que $s(c_i, t) = 1$ representa o estado

“vivo” e $s(c_i, t) = 0$ o estado “morto”. Note-se que a função $s(c_i, t)$ mapeia os estados da célula c_i em um determinado tempo t ;

- A **condição inicial** do AC é descrita por s_0 . Nela atribui-se a cada célula c_i um estado inicial no tempo $t = 0$, é dizer $s_0 = s(c_i, 0)$.
- A função $\mathbf{N}$ mapeia o conjunto de **vizinhança** de cada célula c_i . Além disso, em ACs bidimensionais podem ser definidas as vizinhanças de von Neumann ($|\mathbf{N}| = 4$) e Moore ($|\mathbf{N}| = 8$), em que o número de vizinhos da célula c_i é representado por $|\mathbf{N}(c_i)|$;
- A **função de transição** ou regra $\phi : \mathbf{S}^{|\mathbf{N}(c_i)|} \rightarrow \mathbf{S}$ define a dinâmica de evolução do AC. Assim, os estados futuros da célula c_i dependem da configuração atual dos estados dela mesma e/ou da sua vizinhança. Note-se que a evolução dos estados de todas as células é feita de forma simultânea e sincronizada (43);

Diferentes variações de ACs são encontrados na literatura. Por exemplo, dependendo da regra de transição, os ACs podem ser **determinísticos** ou **não determinísticos**. No último caso, é adicionada uma função de probabilidade para definir os próximos estados. Os ACs são chamados de **homogêneos**, quando há uma única regra global, comum a todas as células, que governa a dinâmica do AC, caso contrário, são chamados de **heterogêneos** ou híbridos. Os ACs também podem ser **totalistas** ou **outer-totalistas**. No primeiro caso, a função de transição depende apenas do total de células na vizinhança $\mathbf{N}(c_i)$ incluindo-se a própria célula c_i

$$s(c_i, t + 1) = \phi(s(c_{ij}, t)_{j=1}^{|\mathbf{N}(c_i)|}).$$

Já no segundo caso, existe uma sutil diferença, chama-se AC outer-totalista, quando a regra depende dos estados da sua vizinhança e do estado desta mesma como parâmetros, tal como é mostrado a seguir:

$$s(c_i, t + 1) = \phi(s(c_{ij}, t)_{j=1}^{|\mathbf{N}(c_i)|}, s(c_i, t)).$$

A seguir são introduzidos exemplos de ACs totalísticas e outer-totalísticas. No primeiro caso, o autômato celular elementar (21) e no segundo caso o AC bidimensional “jogo da vida” do John Conway (44), respectivamente.

2.4.1 Autômato celular elementar

Em 1983, Wolfram (45) propôs o autômato celular elementar (ACE). Este autômato pode dar origem a comportamentos extremamente complexos, apesar da sua simplicidade em termos de definição. Sem dúvidas, Wolfram é um dos cientistas mais influentes nesta

área, logo depois de von Neumann e Turing. Nesta seção daremos uma breve introdução ao ACEs, porém o leitor pode encontrar uma descrição mais detalhada nas obras. (46, 47)

A tesselação de um ACE $\langle \mathcal{T} \rangle$ é disposta em um arranjo linear de células e é definido por $\langle \mathbb{Z}^1, \mathbf{S} = \{0, 1\}, s_0, \mathbf{N} = (c_{i-r}, c_i, c_{i+r}), \phi : \mathbf{S}^3 \rightarrow \mathbf{S} \rangle$. Observe que, como o ACE é do tipo totalista e o raio $r = 1$, a vizinhança inclui as células da esquerda, da direita e ela mesma, também conhecido como acomplamento com o vizinho mais próximo.

As regras de transição do ACE são definidos por uma tabela, na qual são mapeadas cada uma das possíveis configurações de vizinhança em relação ao possível estado de saída. (45) Portanto, dado que o tamanho da vizinhança é $2r + 1$, existem $2^{2^3} = 256$ regras. Por outro lado, para poder visualizar o diagrama espaço-tempo, Wolfram (45) descreveu uma disposição bidimensional, isto é, a evolução das células ao longo do tempo t são empilhadas verticalmente, representando um imagem de tamanho $|\mathcal{T}| \times t$.

Na Figura 8 é ilustrado um ACE com regra 30, ou seja a função de transição indicando 00011110 em binário, a qual é a forma de representar a dinâmica do ACEs segundo o modelo de Wolfram. (45) Já que a tesselação contém 3 vizinhos, logo existem 2^3 possíveis cenários, explicados a seguir. Por exemplo, no painel superior da Figura 8, os quadrados superiores representam a vizinhança e o quadrado inferior a saída da função. Da esquerda para a direita, para uma configuração da vizinhança $\mathbf{N} = (c_{i-r}, c_i, c_{i+r})$ na qual todos as células estão vivas, seria representado por 111 em binário, o estado da célula c_i é $s(c_i, t + 1) = 0$ no estado seguinte. Para 110 o estado de c_i é $s(c_i, t + 1) = 0$, para 101 o estado de c_i é $s(c_i, t + 1) = 0$, para 100 o estado de c_i é $s(c_i, t + 1) = 1$, para 011 o estado de c_i é $s(c_i, t + 1) = 1$, e assim por diante. Nesta mesma figura, é ilustrada a evolução da célula c_i cuja vizinhança é da forma 000, pelo qual a seguinte evolução será $s(c_i, t + 1) = 0$.

Figura 8 – Autômato celular elementar com regra 30 (acima) e seu respectivo diagrama espaço-tempo (abaixo), no qual 1 corresponde a branco e 0 a preto. O eixo vertical corresponde às evoluções no tempo t e o eixo horizontal representa a tesselação $\langle \mathcal{T} \rangle$.

Fonte: Elaborada pela autora.

A partir da dinâmica global dos ACEs diferentes diagramas espaço-tempo podem surgir. Estes padrões foram classificados em 4 classes de acordo com o esquema qualitativo introduzido por Wolfram (43), os quais são: (i) Classe 1: a evolução leva a um estado

homogêneo; (ii) Classe 2: a evolução leva a um conjunto de estruturas estáveis ou periódicas; (iii) Classe 3: a evolução leva a um padrão caótico e (iv) Classe 4: a evolução leva a estruturas complexas locais, às vezes longas, embora não exista muito consenso com respeito a esta última classe. (48)

A Figura 9 apresenta exemplos de ACEs com regras 172, 7, 30 e 110, correspondente às quatro classes de Wolfram, respectivamente.

Figura 9 – Classificação qualitativa do ACE do Wolfram. Diagrama espaço-tempo de 4 ACEs correspondentes à classificação qualitativa de Wolfram. Os diagramas correspondem a um arranjo linear de 600 células (horizontal) com condição inicial s_0 aleatória evoluídas durante $t = 400$ iterações (vertical).

Fonte: Elaborada pela autora.

2.4.2 Autômato celular da família “*Life-Like*”

Um AC bidimensional é representado em uma tesselação regular $\mathbb{Z}_L^2$, em que o número de vizinhos (células) é igual a L^2 . Um clássico exemplo de AC bidimensional é o “jogo da vida” (*game of life*, em inglês) proposto por John Conway (44), cujo nome foi dado por tentar simular o ciclo reprodutivo das espécies. O jogo da vida é definido pela regra B3-S23, ou seja, uma célula **nasce** (B) quando o total de vizinhos vivos é exatamente três, ou **sobrevive** (S) se possui dois ou três vizinhos vivos. Esta regra segue a notação $Bx-Sy$, em que x e y correspondem a combinação de 0 até 8 algarismos usando diferentes dígitos entre 0 e 8, em ordem crescente. Esta mesma notação é usada para identificar todas as

regras dessa família de ACs, conhecida como AC *Life-Like*. Uma vez que cada célula pode ter um máximo de 8 vizinhos (vizinhança de Moore), logo existem em total $2^{18} = 262,144$ regras de transição ao se combinar as condições de nascimento (B) e sobrevivência (S).

A fim de ilustração, a Figura 10 apresenta diferentes regras da família *Life-Like*, nas quais diversos padrões podem ser observados a partir da evolução espaço-tempo (labirintos, coagulação da sangue, amebas, etc). Algumas das regras de transição $Bx - Sy$ foram profundamente investigadas como por exemplo a regra B1357-S2468, a qual apresenta propriedades caóticas que foram bem sucedidas para a Criptografia. (49)

Figura 10 – Exemplos de ACs da família *Life-Like*. Um AC bidimensional é representado em uma grade regular contendo 500×500 células com condição inicial s_0 aleatória. Cada coluna, apresenta os estados do AC bidimensional conforme evoluídas durante $t = \{10, 20, \dots, 50\}$ iterações (de cima para baixo), no qual diferentes padrões podem ser observados.

Fonte: Elaborada pela autora.

2.5 Rede-autômato

Como visto anteriormente, os autômatos celulares tipicamente são definidos em uma tesselação regular, na qual o número de vizinhos é fixo (vizinhança homogênea). No entanto, ao considerar ACs construídos sobre tesselações irregulares, como por exemplo uma triangulação Voronoi ou inclusive uma rede, o número de vizinhos de cada célula pode variar consideravelmente (vizinhança heterogênea). (37)

Uma **rede-autômato** é uma generalização de ACs, no qual os estados, espaço e tempo também são discretos. A principal característica de uma rede-autômato é que ela usa a rede (tesselação) como hábitat para simular vida artificial. Nessa rede, o conjunto de vértices representa as células e as arestas representam a vizinhança. Logo, atribuí-se-lhe regras que governam os estados das células, a partir da qual podem emergir propriedades intrínsecas da rede, as quais podem ser visualizadas pelos padrões **espaço-tempo** que emergem. (50)

A integração dos conceitos autômatos celulares e grafos foi introduzida pela primeira vez por Watts (51, p. 187) em 1999, tempo depois, Wolfram também associou estes termos. (47, p. 475) No ano 2009, Marr & Hütt (52) estudaram a dinâmica de evolução das redes por meio de ACs, explorando regras de transição baseadas nos critérios do problema da maioria absoluta, que é um problema clássico nos ACs, que procura pela sincronização das células para atingir a maioria de votos. Os resultados obtidos por Marr & Hütt indicaram uma forte correlação das medidas de entropia e a distribuição de grau das redes. Foi só no ano 2011, que Smith et al. (53) introduziram formalmente o nome de rede-autômato (*network automata*, em inglês).

Os padrões de evolução (diagrama espaço-tempo) da dinâmica da rede-autômato foram estudadas também por Zhou & Lipowsky (54), baseada na topologia de rede livre de escala, nesse modelo os estados binários de cada vértice (modelo Ising) são governados por regras de maioria local. Estes autores constataram que as redes livre de escala apresentam diferentes comportamentos encontrando um ponto crítico em $\gamma = 5/2$ (expoente da lei de potência). Por outro lado, Wu et al. (55) estudaram o comportamento do modelo de ACs Greenberg-Hastings em redes livre de escala. No ano 2010, Baetens & De Baets (37) analisaram a estabilidade da topologia dos ACs totalísticos irregulares por meio do expoente de Lyapunov e, que posteriormente, em 2012, foi introduzida uma generalização de ACs em tesselações em termos de redes. (56)

Formalmente, uma rede-autômato $\mathcal{R}$ pode ser definido pela tupla

$$\mathcal{R} = \langle \mathcal{T}, \mathbf{S}, s_0, \mathbf{N}, \phi \rangle .$$

A tesselação $\mathcal{T}$ do AC é representada pela topologia da rede, assim cada vértice será representado por uma célula v_i a qual está conectada com seus k_i vizinhos. $\mathbf{S}$ é o conjunto

de estados, por exemplo, estados binários $s_i = 1$ (vivo) e $s_i = 0$ (morto). A função $s(c_i, t)$ indica o estado da célula c_i no instante t . Por fim, s_0 representa a configuração inicial de todas as células e ϕ é a função de transição, i.e., a regra que governa a dinâmica da rede-autômato, a qual define como os estados das células são atualizados no tempo. (50)

Na rede-autômato, ao contrário dos ACs, o número de vizinhos de cada célula está restrito à vizinhança de cada vértice da rede. Portanto, a função $\mathbf{N}$ mapeia a conectividade de cada vértice, e a função de transição ϕ é dada como uma função da densidade da vizinhança, σ_i , definida como a proporção de vizinhos vivos. Entretanto, diferentes funções de transição podem ser definidas, como por exemplo o modelo Marr & Hütt é baseado no limiar σ que define a porcentagem de vizinhos vivos. (52)

Para fins ilustrativos, a Figura 11 apresenta uma rede-autômato com configuração de rede evoluída durante três etapas de tempo e seu respectivo diagrama espaço-tempo, onde a evolução de cada célula é empilhada de cima para baixo e cujo espaço é composto por todas os vértices da rede, colocados da esquerda para a direita de acordo com um critério, por exemplo, rótulos dos vértices, ordenação de acordo com a conectividade, entre outros.

Figura 11 – (a) Exemplo de dinâmica de evolução de uma rede-autômato com estados binários que evolui com uma regra homogênea ilustrativa. (b) Correspondente diagrama espaço-tempo. Cada coluna representa um vértice da rede e cada linha representa os estados de todos os vértices em cada etapa. (c) Ilustração de um padrão espaço-tempo para uma rede de pequeno-mundo com 500 vértices evoluída por $t = 500$ iterações.

Fonte: Elaborada pela autora.

2.6 Redes complexas

Hoje em dia, as redes complexas tornaram-se uma área multidisciplinar e robusta que surgiu da unificação dos aspectos matemáticos da teoria de grafos e da mecânica estatística. (24, 25) Com as redes complexas conseguem-se representações abstratas dos sistemas complexos (24, 57, 58), onde os componentes do sistema (vértices) são conectados (com ou sem direção) com seus vizinhos, os quais estabelecem algum tipo de relação (arestas) entre si de acordo com a abstração do problema. A seguir são discutidos os conceitos mais importantes da teoria dos grafos, as medidas estruturais mais usadas para a caracterização destas, e os principais modelos de redes encontrados na literatura.

Formalmente, as redes complexas são representadas através de grafos. Um grafo é descrito por $\mathcal{G}(\mathcal{V}, \mathcal{E})$, com um conjunto de vértices $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ interligados por um conjunto de arestas $\mathcal{E} = \{e_1, e_2, \dots, e_M\}$, de tamanho N e M , respectivamente. (14, 57) Uma rede também pode ser representada por uma matriz de adjacência A de $N \times N$, em que $A_{ij} = 1$, se v_i está conectado a v_j ou $A_{ij} = 0$ caso contrário. Por outro lado, quando trata-se de redes ponderadas, no qual cada aresta possui um peso w_{ij} , a rede pode ser representada pela matriz de pesos W .

2.6.1 Medidas estruturais

Dado que as redes complexas podem apresentar diferentes estruturas, topologias e características, diversas medidas são usadas para a sua caracterização. Uma ampla discussão e detalhes sobre as medidas estruturais pode ser encontrada no trabalho de Costa et al. (14) A seguir são introduzidas várias medidas estruturais, as quais podem ser organizadas de acordo com a seguinte divisão:

- **Medidas relacionadas à conectividade**

Uma das mais importantes medidas, é o **grau** (ou conectividade), k_i é o número de vizinhos do vértice v_i definido por

$$k_i = \sum_{j=1}^n A_{ij}. \quad (2.8)$$

Quando trata-se de **redes direcionadas**, o número de ligações de entrada e saída k_i^{in} e k_i^{out} são definidos por

$$k_i = k_i^{\text{in}} + k_i^{\text{out}} = \sum_{j=1}^N A_{ij} + \sum_{j=1}^N A_{ji}. \quad (2.9)$$

De forma similar, a conectividade das redes ponderadas é dada por

$$k_i^w = \sum_{j=1}^N w_{ij}. \quad (2.10)$$

O **grau médio** $\langle k \rangle$ da rede é definido pela Equação (2.11).

$$\langle k \rangle = \frac{1}{N} \sum k_i. \quad (2.11)$$

As redes também podem ser caracterizadas de maneira hierárquica, ou seja, analisar as vizinhanças mais distantes em base a níveis ou camadas. O **grau hierárquico** H_{k_h} considera a conectividade dos vizinhos de vértices que estão restritos ao nível hierárquico h . Pelo qual $\langle H_{k_h} \rangle$ define o grau hierárquico médio da rede, definido por:

$$\langle H_{k_h} \rangle = \frac{1}{N} \sum_i H_{k_h i}. \quad (2.12)$$

Por sua vez, a probabilidade com que dois vizinhos do vértice v_i estejam conectados entre si formando um triângulo é chamado de **coeficiente de aglomeração** (*clustering coefficient*) (59), definido por

$$cc_i = \frac{2e_i}{k_i(k_i - 1)}, \quad (2.13)$$

em que e_i é o número de pares conectados entre todos os vizinhos de v_i . Além disso, o **coeficiente de aglomeração médio** é dado por

$$\langle cc \rangle = \frac{1}{N} \sum_i cc_i. \quad (2.14)$$

Um outro conceito de conectividade, é a **distribuição de grau** $P(k)$, que é assumido como a probabilidade de um vértice escolhido aleatoriamente ter grau k . Particularmente, esta medida é bastante útil para diferenciar entre os diversos modelos de redes como a rede aleatória, a rede de pequeno-mundo e a rede livre de escala.

- **Medidas relacionadas à distância**

Ao percorrer uma rede, à sequência de vértices visitados para ir do vértice v_i ao v_j é chamada de **caminho**. Denotamos d_{ij} , como o comprimento do menor caminho entre dois vértices, em que $d_{ij} = \infty$ quando não houver um caminho entre v_i e v_j . O **comprimento médio dos caminhos mínimos** de uma rede $\langle l \rangle$ é calculado encontrando os menores caminhos entre todos os pares de vértices, definido da seguinte forma:

$$\langle l \rangle = \frac{1}{N(N-1)} \sum_{i \neq j} d_{ij}, \quad (2.15)$$

em que $\frac{1}{N(N-1)}$ é o fator de normalização para uma rede totalmente conectada.

- **Medidas da correlação de grau**

Uma outra medida importante de redes é a **assortatividade** ρ_P que quantifica a tendência dos vértices de conectarem-se a vértices com grau similar. (60) Em geral, é usado o conceito de correlação de grau, assim $\rho_P > 0$ indica maior tendência de conexão entre vértices com grau similar (rede assortativa), $\rho_P < 0$ indica que vértices com grau alto tendem a conectar vértices com grau baixo (rede não assortativa), e $\rho_P = 0$ indica a inexistência de correlação entre os graus. A medida de assortatividade de grau é definida na Equação (2.16).

$$\rho_P = \frac{\frac{1}{M} \sum_{j>i} k_i k_j A_{ij} - \left[\frac{1}{M} \sum_{j>i} \frac{1}{2} (k_i + k_j) A_{ij} \right]^2}{\frac{1}{M} \sum_{j>i} \frac{1}{2} (k_i^2 + k_j^2) A_{ij} - \left[\frac{1}{M} \sum_{j>i} \frac{1}{2} (k_i + k_j) A_{ij} \right]^2}. \quad (2.16)$$

- **Medidas de centralidade**

A medida de **intermediação** reflete a importância de um vértice em relação à carga de informação que recebe. Dado v_j e v_k , dois vértices não adjacentes, isto é, que não estão diretamente vinculados, a comunicação entre eles depende dos vértices que pertencem aos caminhos que conectam v_j e v_k . Portanto, a intermediação b_i de um vértice v_i é calculada em função do número de caminhos geodésicos passando por v_i :

$$b_i = \sum_{j,k,j \neq k} \frac{d_{jk}(i)}{d_{jk}}, \quad \text{ou normalizada como} \quad b_i = \frac{1}{N^2} \sum_{j,k,j \neq k} \frac{d_{jk}(i)}{d_{jk}}, \quad (2.17)$$

em que $d_{jk}(i)$ é o número de caminhos geodésicos que conectam v_j e v_k passando por v_i , e d_{jk} é o número de caminhos geodésicos que conectam v_j e v_k . Finalmente, N^2 é o número total de arestas possíveis.

- **Medidas espectrais**

O **espectro** é uma medida importante no estudo da similaridade entre dois grafos. (61) A teoria espectral dos grafos estuda as propriedades destes através dos autovalores e autovetores de suas matrizes associadas: a matriz de adjacência A e a matriz Laplaciana L . A matriz laplaciana é dada por $L = D - A$, em que D é a diagonal dos graus de A . Dado que a rede está ordenada, então os autovalores Laplacianos seguem $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N$.

- **Outras medidas**

Devido à heterogeneidade das redes, muitas vezes, vértices com grau baixo tendem a se conectar com vértices de grau alto, chamados de *hubs*. Por outro lado, também

existem os vértices chamados de *autoridades* que são vértices ligados a muitos *hubs*. Nesse contexto, uma outra medida importante provém da análise de redes da Internet, baseado no algoritmo de HITS*. (62) O algoritmo de HITS atribui um peso $h(v_i)$, $a(v_i)$ para os *hubs* e *autoridades* respectivamente, tal que

$$a(v_j) = \sum_{a_{ij}=1} h(v_i), \quad (2.18)$$

$$h(v_i) = \sum_{a_{ij}=1} a(v_j), \quad (2.19)$$

em que $h(v_i) = a(v_i) = 1$.

2.6.2 Modelos de redes complexas

As redes são classificadas em modelos de acordo à forma de construção destas. Diferentes modelos de redes foram propostos, os quais podem ser caracterizados pela sua topologia e suas propriedades estatísticas, como a distribuição de grau, o coeficiente de aglomeração e o comprimento médio dos caminhos mínimos. A seguir, são introduzidos alguns dos principais modelos de redes complexas encontrados na literatura.

2.6.2.1 Rede aleatória

O modelo de **rede aleatória** concebido por Erdős & Rényi (ER) (63) em 1959, constitui uma das primeiras tentativas para a modelagem de redes reais. Trata-se de um dos modelos mais simples de rede. A rede ER é construída a partir de N vértices, inicialmente desconectados. Sendo que cada vértice v_i possui $(N-1)$ diferentes possibilidades de conexão, então os pares de vértices são conectados aleatoriamente dependendo da probabilidade $p \in [0, 1]$. Logo, por meio desta construção, é gerada uma rede caracterizada por manter uma conectividade média

$$\langle k \rangle = p(N-1), \quad (2.20)$$

cujas distribuição de grau $P(k)$ segue uma distribuição Poisson. (24) Observe que, para o valor $p = 1$ obtêm-se uma rede completamente conectada, atingindo o máximo número de arestas $\frac{N(N-1)}{2}$.

2.6.2.2 Rede pequeno-mundo

Quase 40 anos depois da introdução das redes aleatórias, em 1998, Watts & Strogatz (59) perceberam que as redes do mundo real não eram aleatórias. Eles perceberam que as redes sociais eram **pequeno-mundo**, ou seja, em que qualquer indivíduo pode ser alcançado através de uma cadeia curta de amigos e pela alta probabilidade de que “os amigos dos meus amigos” estejam conectados entre si, formando triângulos (alto coeficiente de aglomeração). (14, 64)

* HITS, acrônimo de *Hypertext Induced Topic Selection*.

A construção do modelo de rede pequeno-mundo proposta por Watts & Strogatz (WS) (59) inicia-se com uma rede regular, ou um anel de n_0 vértices conectados com seus 2κ vizinhos mais próximos, evitando multiplicidades e auto-conexões. Após esse processo, as arestas são reconectadas aleatoriamente com probabilidade p , considerando que $N \gg \kappa \gg \log N \gg 1$. Observe que, obtêm-se uma rede regular quando $p = 0$, enquanto que uma rede aleatória quando $p \rightarrow 1$. (51) A rede pequeno-mundo também segue uma distribuição de grau do tipo Poisson, além disso, caracteriza-se por ter um alto coeficiente de aglomeração enquanto um pequeno comprimento médio de caminho geodésico (59) e conectividade média

$$\langle k \rangle = 2\kappa. \quad (2.21)$$

2.6.2.3 Rede livre de escala

Apenas um ano depois, em 1999, Barabási & Albert (BA) (65) descobriram que certas redes do mundo real, como por exemplo a Internet, caracterizam-se por ter propriedades pequeno-mundo e cuja distribuição de grau não é aleatória, mas que segue uma lei de potência da forma

$$P(k) \sim k^{-\gamma}, \quad (2.22)$$

na qual γ é o expoente da lei de potência, ou seja trata-se de uma rede livre de escala. (24) De fato, foi descoberto que este valor varia entre $2 \leq \gamma \leq 3$ (66,67) para diversos exemplos de redes do mundo real. Assim, estas redes são caracterizadas por possuir poucos vértices altamente conectados (*hubs*) enquanto muitos outros com pouquíssimas conexões.

Na literatura existem vários modelos de rede livre de escala, por exemplo, o modelo proposto por Barabási & Albert (BA) (65) e o modelo Dorogovtsev & Mendes (DM). (68) O modelo de rede BA é baseado nos conceitos de crescimento e ligação preferencial. Desta forma, a partir de um conjunto de vértices iniciais n_0 conectados de forma arbitrária, a cada iteração, novos vértices com m novas arestas são ligadas aos vértices mais conectados com probabilidades proporcionais (Equação (2.23)), dessa forma “o rico fica mais rico”. (66)

$$P(v_i \rightarrow v_j) = \frac{k_j}{\sum_{v_m \in \mathcal{V}} k_m}. \quad (2.23)$$

Por outro lado, as redes livre de escala podem ter coeficiente de ligação preferencial α linear e não linear. No primeiro caso, tratam-se das redes do modelo BA que seguem um ligação linear ($\alpha = 1$). Já no segundo caso, como por exemplo $\alpha = \{0.5, 1.5, 2\}$ s, tratam-se de redes que podem ser sublineares, quando a distribuição de grau da rede é esticada exponencialmente e o número de *hubs* é muito menor; ou supralineares, quando quase todos os vértices estão conectados a alguns *hubs*.

Por outro lado, a construção do modelo Dorogovtsev & Mendes (68) começa com três vértices e três arestas, formando um triângulo, em seguida, cada vez que um novo vértice é adicionado uma aresta é escolhida aleatoriamente, de tal forma que o vértice novo é ligado por duas novas arestas conectando-se as extremidades da aresta escolhida. (68)

2.6.2.4 Rede geográfica

Os três modelos apresentados anteriormente são geralmente tratados em um espaço adimensional, em que a posição espacial dos vértices não é considerada. Porém, certos modelos de redes são chamadas de **redes geográficas** (GEO), uma vez que a posição espacial influencia a evolução da rede, por exemplo, as redes de aeroportos, redes de energia, entre outros. (69)

Um dos modelos de redes mais conhecidos é o proposto por Waxman para construir uma rede GE é distribuindo N vértices aleatoriamente em uma tesselação bidimensional e interligando os vértices com uma determinada probabilidade p que decai proporcionalmente com a distância entre os vértices, tal que

$$P_{\text{wax}}(v_i \rightarrow v_j) = \beta e^{-\theta_{ij}/L\eta}, \quad (2.24)$$

no qual θ_{ij} é a distância euclidiana entre vértices, L é a distância máxima entre dois vértices, e β, η são parâmetros na faixa $(0, 1]$. (69) Valores altos de β resultam em redes com maior densidade de arestas, enquanto que pequenos valores de η implicam um aumento na densidade de arestas curtas em relação das mais compridas. (69)

2.7 Reconhecimento de padrões em redes

A literatura apresenta uma grande quantidade de aplicações baseadas na análise de redes. Em uma tentativa de agrupar as linhas de pesquisa de acordo como o objetivo do estudo, que vão desde o nível micro ao macro, destacam-se três categorias: (i) extração de informações locais da rede, (ii) mineração de dados da rede e (iii) reconhecimento de padrões em redes.

• Extração de informações locais da rede

Reúne às linhas de pesquisas que visam estudar a natureza individual dos componentes da rede. Nesta linha, as propriedades topológicas são usadas para extrair informações locais que levem à compreensão do problema em questão (25), por exemplo: a detecção de *hubs* para investigar a influência dos usuários nas redes sociais (70), ou a análise das implicações epidemiológicas dos indivíduos alvos de campanhas de sexo seguro (71); a medição dos comprimentos dos caminhos mais curtos de um vértice para outro, como foi demonstrado o fenômeno pequeno-mundo e os “seis graus de separação” (72); a remoção de vértices com alta conectividade em redes metabólicas (73), entre outros;

- **Mineração de dados da rede**

Esta linha de pesquisa tem como principal objetivo a mineração de dados das redes, como por exemplo descobrir padrões de ligações, ou ligações faltantes, para prever o surgimento de ligações futuras a partir da estrutura de rede atual (74); ou prever como um vídeo viral se espalhará pelas redes sociais (75); ou de prever a disseminação geográfica da gripe com o intuito de reduzir o impacto da doença (76), entre outros;

- **Reconhecimento de padrões em redes (RPR)**

Aqui são agrupadas as linhas de pesquisa que visam o modelamento de um problema específico usando redes em um contexto de reconhecimento de padrões. Esta linha tem como objetivo a caracterização de redes por meio da extração de informações da correlação entre vértices e sua relação com a topologia da rede para sua posterior classificação usando técnicas de reconhecimento de padrões. (24, 77) De modo geral, a metodologia de RPR segue três fases:

modelagem de dados em redes $\rightarrow$ extração de características $\rightarrow$ classificação.

2.7.1 Modelagem de dados em redes

Na literatura, existem diversas propostas para a modelagem (e/ou transformação de dados) com redes. (58) Isto porque as teorias de grafos, e redes em geral, dão suporte e flexibilidade para representar quase qualquer sistema que envolve um conjunto de componentes e suas inter-relações. Perceba que o termo “dados” pode ser usado em sentido amplo, pode tratar-se de sequências de números, sequências de palavras (textos), séries temporais, imagens (contornos, texturas), reações metabólicas, entre muitos outros tipos de abstrações. Essencialmente, a representação de dados abrange um amplo espectro de aplicações no mundo real.

Uma das áreas que usufruiu muito das vantagens do marco teórico de redes, no contexto de RPR, é a Visão Computacional. De acordo com Casanova (78), diferentes formas de modelagem podem ser usadas nos mais diversos problemas, como é a análise de texturas (18, 79), onde os pixels da imagem representam os vértices e as arestas são definidas seguindo algum critério (cor, intensidade, adjacência, raio, etc.) (78); ou a análise de contornos (17, 80, 81), onde dado um contorno de imagem, cada ponto da curva representa um vértice. Assim, dependendo de um limiar (distância euclidiana) várias redes podem ser formadas, as quais, tipicamente, apresentam características de pequeno-mundo (17)

Na área de biologia, as reações metabólicas também são modeladas de diferentes formas, onde os metabólitos representam os vértices, e a conectividade é definida seja por uma relação enzimática, substrato-produto, entre outras. (82) Um dos modelos de redes metabólicas mais estudadas foi proposto por Jeong et al. (23), que estudaram o banco de redes WIT (83), correspondentes a 43 organismos pertencente a 3 reinos (*archaea*, *bacteria*

e *eukaryote*). Esta mesma base foi usada no contexto de RPR para análise de relações evolutivas. (84)

Recentemente, a área de Processamento de Linguagem Natural (PLN) vem-se aproveitando também das vantagens da modelagem em redes. Diversas aplicações no contexto de RPR incluem o reconhecimento de autoria (85–87), classificação textual e a estilometria (88, 89), nos quais textos são modelados como redes complexas, nas quais os vértices representam palavras e as arestas representam as relações de adjacência.

Por outro lado, a modelagem de séries temporais em redes também vem sendo estudada. Por exemplo, séries temporais geradas por sistemas não lineares foram modeladas como redes de recorrência. (15) Uma outra abordagem foi a transformação das sequências de dígitos de números irracionais e transcendentais em uma rede (d, r) , cada d dígitos correspondem a um vértice e sua conectividade era determinada pela adjacência dependendo de um raio r . (90, 91) Embora, o contexto de RPR de séries temporais, ainda seja pouco explorado, acreditamos conveniente mencioná-los, pois esta modelagem serviu como inspiração para esta tese, como será visto no Capítulo 6.

2.7.2 Extração de características

Em geral, a extração de características é o segundo passo na metodologia de reconhecimento de padrões, cujo principal objetivo é a construção de um **vetor de características** $\vec{\mathcal{F}} = \{f_1, f_2, \dots, f_n\}$, que compreende n **descritores** (atributos) representativos que se assemelham para objetos da mesma categoria e que se diferenciam para objetos de diferentes categorias. (92) Além disso, é esperado que esses descritores sejam invariantes às transformações da entrada, assim como de garantir robustez ante a remoção de vértices ou arestas em uma rede.

No contexto de redes, a extração de características é tipicamente baseada em medidas estruturais, por ser uma das abordagens melhor estabelecidas e adotadas na literatura. (14, 24, 77) Isso porque, em conjunto, muitas medidas como o grau, caminhos médios, coeficiente de aglomeração, medidas hierárquicas, centralidades e inclusive medidas espectrais (61), conseguem distinguir redes dos modelos aleatória, pequeno-mundo, livre de escala e geográfica. (24, 77) Dependendo da aplicação, a escolha dos descritores pode ser crucial para uma caracterização da rede de forma adequada. (24, 77) Por um lado, no caso do modelo de rede de contornos (17), apenas as medidas de grau médio e grau máximo já conseguem discriminar contornos com muito sucesso. Por outro lado, ainda há várias limitações que devem ser consideradas. Por exemplo, a distribuição do grau $P(k)$ é útil para discriminar dentre os vários modelos teóricos de redes ER, WS, BA e GEO. Porém, quando trata-se de redes reais, essa mesma propriedade não é mais adequada, porque existem redes com topologias diferentes que podem apresentar distribuição de grau similar, portanto falhando na sua classificação correta. (24, 93)

Uma segunda limitação, discutida por Rodrigues (24), é que apenas medidas como a distribuição de grau, grau médio, coeficiente de aglomeração e menor caminho médio não são suficientes para uma caracterização completa da rede. Consequentemente, poderia pensar-se em usar um maior número de medidas que possam trazer um maior número de propriedades estruturais, porém, isto também resultaria em redundância de dados, pois algumas destas medidas são correlacionadas. (77) Ademais, na área de reconhecimento de padrões é aconselhado evitar a redundância de descritores, pois estes podem incorrer no dilema da “maldição da dimensionalidade”. (92) Para contornar a limitação da redundância, porém nem sempre com sucesso, Costa et al. (77) propuseram o uso de técnicas para a redução da dimensionalidade como a análise discriminante linear e a análise de componentes principais (LDA e PCA, pelas siglas em inglês, respectivamente). (92)

Recentemente, outros métodos mais sofisticados foram propostos diferentes das medidas estruturais clássicas. Em 2012, Gonçalves *et al.* (94) propôs um método baseado na caminhada determinística do turista para classificar modelos de redes usando a trajetória do agente sobre a topologia de rede. Seus resultados indicam uma melhora na taxa de classificação quando comparada às medidas estruturais da rede. Além disso, este método também foi usado para o reconhecimento de padrões em visão computacional, como contorno e análise de texturas. (18, 19, 95)

2.7.3 Classificação

O **reconhecimento de padrões** tem como objetivo a descoberta de padrões nos objetos abstratos por meio de algoritmos computacionais (aprendizado de máquina) que permitam sua classificação. (9) As técnicas de aprendizado de máquina, dependendo do conhecimento prévio das classes, são divididas em: supervisionadas, semi-supervisionadas e não supervisionadas (9), sendo que nesta tese foi dada especial atenção para a classificação supervisionada.

Na literatura existem diversos **algoritmos de classificação** supervisionados (9) como, por exemplo, os k vizinhos mais próximos (kNN), *Naive Bayes* (NB) e máquina de vetores de suporte (SVM, pelas siglas em inglês, *Support Vector Machine*). Todos esses são chamados de aprendizagem supervisionada porque são algoritmos que recebem um **conjunto de dados de treinamento** $\mathcal{X} = \{\vec{\mathcal{F}}_1, \vec{\mathcal{F}}_2, \dots, \vec{\mathcal{F}}_q\}$, com q o número de amostras (ou instâncias) previamente rotuladas dentro de um conjunto de **classes** $\mathcal{C}$ correspondentes a um problema de classificação, com o qual supervisionam o processo de aprendizagem. Tendo conhecimento prévio das classes, o algoritmo faz previsões iterativamente sobre os dados de treinamento e é corrigido por este até que atingido um nível aceitável de desempenho.

A estratégia de k -validação cruzada (96) é tipicamente usada para avaliar o desempenho do método de classificação. Neste método, o conjunto de dados é dividido

aleatoriamente em k partes, de modo que $k - 1$ partes são usadas para o treinamento e 1 parte para validação. Assim, este processo é repetido k vezes iterativamente até que todas as partes tenham sido testadas. De fato, ao dividir os dados em um conjunto de treinamento e um **conjunto de teste** $\mathcal{Y} = \{\vec{\mathcal{F}}_1, \vec{\mathcal{F}}_2, \dots, \vec{\mathcal{F}}_r\}$, é garantido que $\mathcal{X} \cap \mathcal{Y} = \emptyset$. Esta divisão é feita com o intuito de gerar um classificador imparcial. (96)

O desempenho do classificador é calculado pela taxa de acerto (ou taxa de erro) que verifica o número de acertos (erros) do conjunto de teste em relação ao conjunto de treinamento e vice-versa. Assim, a taxa de acerto média e desvio padrão podem ser calculados durante o processo iterativo da estratégia de k -validação cruzada, ou após realizadas várias execuções deste. Os resultados de taxa de acerto também podem ser apresentados usando matrizes de confusão M , onde cada célula mostra os valores de cada possível relação entre as classes preditas e as classes reais. Tradicionalmente, uma matriz de confusão é usada para identificar os seguintes marcadores de desempenho:

- M_{c_i, c_i} , ou seja, a diagonal principal, é o número de instâncias pertencentes à classe c_i que foram corretamente atribuídas a c_i ;
- M_{c_i, c_j} é o número de instâncias pertencentes à classe c_i que foram incorretamente atribuídas à classe c_j .

2.7.3.1 Classificadores supervisionados

O kNN é um algoritmo de classificação simples, rápido e eficaz que usa uma métrica de distância para classificar os dados. (96,97) O objetivo é atribuir um rótulo a um dado de instância desconhecido que analise dados semelhantes. A classificação é feita quando uma instância do conjunto de testes é comparada com cada instância do conjunto de dados de treinamento. Em seguida, agrupa os k mais parecidos, que são os vizinhos mais próximos de k , e em seguida olha para os seus rótulos e de acordo com um voto de maioria decide qual é o rótulo para esta nova instância.

À diferença do kNN que é baseado em distâncias, o algoritmo *Naive Bayes*, é um classificador probabilístico baseado na teoria de decisão bayesiana. (96) Para classificar uma nova instância, primeiro é estimada a distribuição de probabilidades de cada uma das classes e, em seguida, são combinadas as probabilidades condicionais com as *priori*, para assim atribuir a esta instância com a classe com máxima probabilidade a *posteriori*.

O SVM é uma técnica de classificação baseada na teoria de aprendizado estatístico. (98) Inicialmente, este algoritmo foi desenhado como classificador binário, no qual são criados hiperplanos como limites de decisão para as novas instâncias. O hiperplano ótimo fornece a separação máxima dos limites entre duas classes e é obtido pela solução de um problema de otimização quadrática, (99) no entanto, o SVM não se restringe a duas classes,

estendendo-se para problemas de classificação de múltiplas classes, como são abordados nesta tese.

2.7.3.2 Métodos de projeção

Na área de reconhecimento de padrões, os métodos de projeção são ferramentas úteis para a visualização de dados multidimensionais. Neste trabalho, temos usados três diferentes métodos: A **análise de componentes principais** (PCA) (9), a **análise de variáveis canônicas** (CVA) (100) e o método t-SNE (do inglês, *t-Distributed Stochastic Neighbor Embedding*). (101)

O PCA é uma das técnicas de redução de dimensionalidade mais usada no reconhecimento de padrões. PCA é baseado na transformação ortogonal dos dados para obter um conjunto menor de variáveis linearmente não correlacionadas chamadas componentes principais. (9)

O CVA é uma técnica supervisionada similar ao PCA, a qual considera a classe dos dados a fim de projetá-los maximizando a separação das classes. (9) A análise canônica é uma análise de regressão que fornece uma combinação linear dos descritores originais que maximiza a separação entre as classes de interesse. (100) Portanto, a primeira e a segunda variáveis canônicas correspondem aos autovetores com os autovalores mais altos. Tais autovetores correspondem à matriz que quantifica a variação dentro da classe em relação às instâncias da mesma classe (*intra-classes*) e à matriz que quantifica a variação entre as classes (*inter-classes*).

O t-SNE é uma técnica de visualização de dados de alta dimensão, proposta em 2008. (101) Nesta técnica, cada vetor de característica é projetado em duas dimensões. A diferença do PCA e do CVA, o t-SNE tem uma função objetiva não-convexa, ou seja, a função objetivo é minimizada usando uma otimização do gradiente que é iniciada aleatoriamente, pelo qual é possível obter visualizações diferentes. (102) Porém, vale ressaltar que este método garante ótima separabilidade das classes, apenas para quesitos de visualização.

3 CAOS E PSEUDO-ALEATORIEDADE

Qualquer um que considere métodos aritméticos para produzir números está, certamente, cometendo um pecado.

— John von Neumann

A estreita relação entre o caos e a pseudo-aleatoriedade vem despertando grande interesse na comunidade acadêmica e o mercado, a qual é fundamentada pelas diferentes propriedades dos sistemas caóticos. (5, 49, 103–133) Baixo essa premissa, o principal foco desta tese é estudar tais propriedades como fontes de aleatoriedade, com o objetivo de explorar um espectro de pseudo-aleatoriedade, que posteriormente foi levado em um contexto de classificação usando métodos de reconhecimento de padrões.

Nesse contexto, este capítulo apresenta uma revisão bibliográfica sobre as sequências de números aleatórios, os tipos de geradores, como estes são construídos classicamente, assim como as aplicações nas quais estas sequências são usadas. Também será comentado o potencial da teoria do caos e suas boas propriedades aproveitáveis como fontes de aleatoriedade. Finalmente, será introduzido como os números aleatórios são testados, quais conjuntos de testes estão disponíveis na literatura e a relação dos padrões em um contexto de reconhecimento de padrões.

3.1 Geradores de números aleatórios e pseudo-aleatórios

3.1.1 O que é uma sequência aleatória?

A aleatoriedade é um dos termos mais usados em quase todos os aspectos da humanidade e intuitivamente o relacionamos com a ausência de padrões aparentes ou com a incapacidade de prever qual será o valor seguinte em uma sequência.

A seguir são mostradas duas sequências U_n e Z_n contendo 6 e 20 números respectivamente. A primeira é gerada por um dado de jogo não viciado, ou seja, que possui um espaço amostral $\omega = \{1, 2, 3, 4, 5, 6\}$, enquanto a segunda é uma sequência binária ($\omega = \{0, 1\}$) gerada arbitrariamente pela autora.

- $U_n = \{3, 6, 3, 6, 3, 6, ?\}$;
- $Z_n = \{0, 1, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 1, ?\}$.

Podemos dizer que estas sequências são aleatórias?

Observando a sequência U_n , é evidente a predominância do padrão “36”, pelo qual pode ser dito que U_n não é aleatória, porém, por tratar-se de um dado não viciado, há grande dificuldade para prever qual será o seguinte valor. No entanto, pode-se afirmar

que este número terá a mesma probabilidade de $1/6$ para todos os casos (distribuição uniforme). Isto porque cada número jogado é independente dos anteriores, razão pela qual existe uma probabilidade de que esta sequência ocorra tanto quanto uma outra desse comprimento.

No que tange à sequência Z_n , esta parece não ter um padrão redundante, porém é possível afirmar que a frequência de 1s e 0s é uniforme, e pelo que podemos dizer que possi uma alta entropia de Shannon. (134) Por outro lado, nesta sequência existem até 8 blocos de informação “0|1|00|11|01|010|111|001|” o que tornaria sua complexidade de Lempel-Ziv (135) razoável. Ambas medidas mencionadas anteriormente pertencem ao contexto da teoria da informação e da teoria da complexidade, as quais serão introduzidas mais formalmente no Capítulo 5.

No entanto, no que tange à fonte de aleatoriedade ainda identifica-se uma sutil diferença. Observe que foi dito que U_n foi produzido por um dado não viciado, e Z_n foi gerada pelo ser humano—ainda existem questões psico-sociais deste assunto, as quais são bem discutidas por Figurska et al. (136) A partir disso surge uma outra questão mais filosófica: existe a verdadeira aleatoriedade? Na comunidade acadêmica existe uma longa discussão, ainda aberta, que paira sobre a existência da verdadeira aleatoriedade. Contudo, nesta tese não se pretende abrir questionamentos sobre a presença, ou não, de aleatoriedade na natureza*, e apenas focar no que pode ser contornado: a pseudo-aleatoriedade.

Existem várias definições na literatura com respeito à aleatoriedade, com diferentes perspectivas que vão desde a (i) estatística, que considera um gerador de aleatoriedade quando são produzidos números equiprováveis (distribuição uniforme) e cujas sequências são independentes uma das outras (138, 139); (ii) na teoria da complexidade algorítmica e na teoria da informação, que considera uma sequência de uns e zeros (N -bits) e chama de aleatório aquela com a maior complexidade possível, ou seja, uma sequência é dita aleatória quando for computacionalmente incompressível ou irreduzível, estudadas por Solomonoff (140, 141), Kolmogorov (142) e Chaitin. (143, p. 111)

Assim, nesta tese segue-se a definição proposta por Kenny (144):

Definição 3. Seja $U_n = \{u_1, u_2, \dots, u_N\}$ uma sequência de variáveis aleatórias, com N sendo o tamanho da sequência e u_i é uma variável aleatória binária, cujos possíveis valores são os eventos $u_i \in \{0, 1\}$. Cumpre que:

$$P(u_i = 1) = P(u_i = 0) = \frac{1}{2}.$$

De forma equivalente, a distribuição conjunta das subsequências de tamanho ω contendo 2^ω pontos, em cada espaço amostral ω , deve ser $1/2^\omega$.

* Mesmo quando existem leis determinísticas na natureza, há também incertezas em relação a conhecer a posição e velocidade de uma partícula (Princípio da incerteza de Heisenberg). Passagens discutidas pelo físico e cosmólogo Stephen Hawking em palestra intitulada *Does God play Dice?* sobre predição do futuro. (137)

Embora a definição considere apenas números binários, esta tese não tratará somente de sequências binárias $\{0, 1\}$, mas também com as sequências de números inteiros (Seção 3.2) e de sequências uniformes entre $[0, 1]$, dependendo da aplicação e do propósito desejado. Isto porque a transformação em qualquer outro intervalo, de forma adequada, pode ser considerada aleatória também.

3.1.2 Aplicações das sequências de números aleatórios

Com o advento da era digital a necessidade de usar números aleatórios tem crescido nas mais diversas áreas como: (i) na estatística e na teoria da probabilidades, para a amostragem de dados; (ii) na teoria de decisão, como estratégia para a otimização de algoritmos computacionais; (iii) no cálculo numérico, como por exemplo, no método de integração de Monte Carlo (145); (iv) na simulação e na modelagem de sistemas, como ferramenta para simular comportamentos estocásticos, como em fenômenos da natureza (146); (v) na indústria de jogos e do entretenimento, para renderização de cenários e texturas em filmes, por exemplo; (vi) linguagens de programação e quaisquer algoritmo ou heurística que precise da amostragem como em reconhecimento de padrões, inteligência artificial, algoritmos genéticos, aprendizagem profunda e, até em cenários mais críticos como (vii) na criptografia, onde os geradores de números aleatórios criptograficamente seguros (147, 148) são de suma importância para os algoritmos de encriptação que atuam em comunicações militares, ferramentas de inteligência, sistemas bancário, funcionamento do comércio eletrônico, entre outros.

Em geral, espera-se que um gerador produza sequências de números com propriedades *ideais*, independentes do ponto de vista estatístico (sendo avaliadas por uma bateria de testes de aleatoriedade) e que sejam uniformemente distribuídos. (127, 138, 149)

Usualmente, um gerador possui uma fonte de aleatoriedade da qual extraem-se números de forma aleatória. De acordo com este fonte, existem duas abordagens na literatura:

- Geradores de números verdadeiramente aleatórios (TRNGs, pela siglas em inglês): Usam os fenômenos físicos como fonte de aleatoriedade, razão pela qual estes geradores podem possuir alta entropia, serem aperiódicos e imprevisíveis. Exemplos de TRNGs abrangem desde o lançamento de um dado ou uma moeda (150) até processos mais sofisticados como o decaimento radioativo (151, 152), o ruído térmico das resistências (153), o ruído atmosférico (154, 155), lâmpada de lava (156) e física quântica (157, 158);
- Geradores de números pseudo-aleatórios (PRNGs, pela siglas em inglês), recebem esse nome pois baseiam-se em algoritmos determinísticos para gerar sequências com propriedades aproximadas às de verdadeira aleatoriedade.

Em um contexto computacional, a maioria dos TRNGs têm várias desvantagens em comparação com um bom PRNG. A maior desvantagem dá-se em termos de implementação. Existem diversos desafios incluindo equipamentos especiais adicionais e circuitos (muitas vezes necessitando de refrigeração ou altas tensões), velocidade limitada, taxas de produção restritas, e em certos casos pode ocorrer degradação das propriedades estatísticas quando implementadas em hardware e software. (159)

Por outro lado, a principal vantagem dos PRNGs em comparação aos TRNGs é em termos de eficiência, pois, como mencionado anteriormente, muitos dos campos de aplicação precisam de grande volume de números aleatórios. Uma outra vantagem é o determinismo, que para diversos fins, é uma característica desejável, especialmente na simulação de experimentos, assim como na criptografia e funções unidirecionais (*hashing*). (148) Além disso, muitos PRNGs são portáteis e podem ser facilmente implementados. No entanto, uma das desvantagens dos PRNGs em comparação aos TRNGs, é que são computacionalmente periódicos, como citado por von Neumann. (138, p. 1) Porém, com os avanços tecnológicos há formas de garantir uma periodicidade extremamente longa. (160, p. 27)

A Figura 12 apresenta um esquema generalizado para a construção de um PRNG. Primeiro, é necessário introduzir uma semente u_0 (dentro do domínio da função geradora), a partir da qual a função encarrega-se de extrair números “aleatoriamente” e transformá-lo (se necessário) para formar uma sequência de números na forma $U_n = \{u_1, u_2, u_3, \dots, u_N\}$.

Figura 12 – Esquema para a construção de um gerador de números pseudo-aleatórios.

Fonte: Elaborada pela autora.

Na literatura existem diversos PRNGs. Os PRNGs clássicos (introduzidos na Seção 3.2) são baseados em recorrências lineares e se apoiam em operações bit a bit (deslocamentos de bits, operadores abelianos) e conceitos algébricos. Como alternativa aos algoritmos tradicionais, na Seção 3.3 é feita uma revisão dos principais PRNGs baseados no caos determinístico (CB-PRNG, pelas siglas em inglês). Neste capítulo também serão apresentadas as propriedades em comum que os relacionam fortemente com a criptografia.

3.2 PRNGs clássicos

Um dos PRNGs mais conhecido e amplamente estudado é o gerador congruencial linear (LCG, pela siglas em inglês) e suas diversas variações. (161) Outro grupo de geradores são baseados no registro de deslocamento com retroalimentação linear (LFSR, siglas em

inglês), o qual funciona por meio do deslocamento de bits. (162) Nessa mesma linha, no ano de 1998 surgiu o Mersenne Twister (163), o qual é usado oficialmente pela linguagem Java (164), e diferentes variações dele como a família de geradores *Xorshift*. (165)

3.2.1 Gerador congruencial linear

Este algoritmo foi introduzido em 1951 por Lehmer (138, 139), baseado na Equação (3.1), que segue uma distribuição uniforme entre $[1, m]$.

$$x_{n+1} = (ax_n + c) \bmod m, \quad (3.1)$$

em que $0 \leq a < m$ é o multiplicador, $0 \leq c < m$ é o incremento e $m > 0$ o módulo, todas as constantes são inteiras. Na equação 3.1, x_0 é chamado de semente inicial, com $0 \leq x_0 < m$ e $n \geq 0$. Desta forma um número x_{n+1} é gerado usando o valor anterior, x_n . Além disso, os números gerados podem ser normalizados entre $[0, 1]$, dividindo-se da forma $\frac{x_n}{m}$.

Existem diversos princípios para a escolha dos parâmetros a , c e m , como por exemplo usar números primos e coprimos (números primos entre si). (138, 139) Em geral, a distribuição dos números gerados pela família de LCGs é uniforme e passa na maioria dos testes de aleatoriedade, embora esses PRNGs não sejam recomendados para criptografia. Atualmente, programas como Excel (166) e linguagens de programação como Java (164) e PHP (167), entre outras, tem bibliotecas nativas que usam LCGs combinadas.

3.2.2 RANDU: quando os PRNGs falham

RANDU, descrito pela Equação (3.2), foi o primeiro PRNG implementado no computador IBM System/360, nos anos 60s, e esteve em vigência por mais de uma década. Este gerador foi um dos casos mais polêmicos da história, devido ao fato de que os parâmetros de a e m tornam-o vulnerável. (168, 169)

$$x_{n+1} = (65539x_n) \bmod 2^{31}. \quad (3.2)$$

A falha deste algoritmo torna-se evidente ao visualizar os números gerados em 3 dimensões, no qual observam-se 15 planos paralelos (ver Figura 13a), o que implica que valores consecutivos (tuplas x_n , x_{n+1} e x_{n+2}) permitem prever exatamente as saídas futuras, apesar de possuir uma distribuição uniforme (como pode ser visto na Figura 13b). Repare que, este comportamento não é visível quando visualizado em duas dimensões, na qual aparenta ser aleatório. De acordo com Marsaglia (168), isso acontece devido à correlação em série entre valores sucessivos da sequência x_n .

A falha do gerador RANDU pode ser detectada usando o teste espectral (FFT) da bateria de testes NIST. (171) O caso do RANDU é um forte indício de que nem todos os PRNGs são bons geradores, pois cumprem parcialmente os requisitos de um

Figura 13 – (a) Diferentes ângulos de visualização dos 4×10^4 números pseudo-aleatórios gerados pelo RANDU (normalizado) dispostos em forma de recorrência (x_n, x_{n+1}, x_{n+2}) , no qual R é o grau de rotação vertical e I a inclinação horizontal. (b) Distribuição das frequências dos números gerados em (a), divididos em 10 bins.

Fonte: Adaptada de PASHLEY; AMODEO. (170)

PRNG adequado. Além disso, também é um forte indício de que projetar, desenvolver e implementar geradores não é uma tarefa trivial, como apontado por Hellekalek. (149) Segundo este autor, novos esforços devem ser considerados para desenvolver propostas alternativas de PRNG que satisfaçam a necessidade de boas fontes de pseudo-aleatoriedade.

3.2.3 Mersenne Twister

O Mersenne Twister (MT)[†], projetado por Matsumoto & Nishimura (163) em 1998, é atualmente um dos PRNGs mais utilizados comercialmente, dado que muitas linguagens possuem bibliotecas com suas implementações adaptadas. Este algoritmo possui um período bastante grande de $2^{19937} - 1$, e, cuja diferença do RANDU garante até 624 planos. (163)

O MT gera sequência de números de 32-bits (palavra) de forma uniforme entre 0 e $2^{32} - 1$ (ou normalizada, se dividida pelo máximo valor), dada pela recorrência linear apresentada na Equação (3.3), em que $x_k \in \mathbb{N}$ é o k -ésimo valor da sequência e as constantes $n = 624$ e $m = 397$.

$$x_{k+n} = x_{k+m} \oplus (x_k^u | x_{k+1}^l) A. \quad (3.3)$$

O Mersenne Twister usa as operações em bits $\oplus$ e $|$. O operador abeliano $\oplus$ é usado para

[†] O Mersenne Twister recebeu seu nome devido a sua periodicidade $2^{19937} - 1$, a qual corresponde a um número primo Mersenne (i.e. um número da forma $M_n = 2^n - 1$, com n um número primo) que atinge os melhores parâmetros para este algoritmo.

torcer os bits da palavra anterior com a palavra resultante do processo de concatenação. O operador $|$ concatena duas palavras parciais x_k e x_{k+1} para formar uma palavra resultante de 32-bits. Assim x_k^u é a parte superior de $32 - r$ bits de x_k , enquanto x_{k+1}^l é parte inferior de r bits de x_{k+1} , com $r = 31$, em outras palavras $x_k^u | x_{k+1}^l$ corresponde a um número inteiro (ou binário) formado pelos bits mais significativos de x_k e dos 31 bits menos significativos de x_{k+1} . Observe que, no MT é necessário ter 624 palavras para semear a sequência que precisam ser armazenadas na memória e diferente de zeros. Finalmente, A é um valor discreto predefinida, cuja forma é escolhida para que a multiplicação seja rápida. Por outro lado, da mesma forma que o LCG, o Mersenne Twister não é criptograficamente seguro.

3.3 PRNGs baseados no caos

Ao longo dos últimos 30 anos, muitos cientistas observaram a forte relação entre o caos determinístico e a pseudo-aleatoriedade. Como foi introduzido no Capítulo 2, esta relação é fundamentada pelas diferentes propriedades dos sistemas caóticos como: a sensibilidade às condições iniciais, ergodicidade, transitividade, complexidade, imprevisibilidade, instabilidade, equações relativamente simples e dinâmica determinística (119, 123, 172, 173), as quais são entendidas como características adequadas para serem fontes de pseudo-aleatoriedade de qualidade.

Aqui são agrupadas as linhas de pesquisa que aproveitam-se das propriedades pseudo-aleatórias dos sistemas caóticos, com as quais são propostos construção de geradores de números aleatórios, algoritmos de *hash*, algoritmos de encriptação, entre outros. O maior enfoque será dado no que tange aos geradores de números aleatórios baseados no caos (CB-PRNGs, pela siglas em inglês), porém salientando que ambas as construções destes métodos são fundamentadas nos mesmos quesitos.

De modo geral, a metodologia dos CB-PRNGs, é dada da seguinte forma:

semente $\rightarrow$ sistema caótico $\rightarrow$ sequência binária.

Os CB-PRNGs são implementados com base em uma equação caótica que, muitas vezes produz valores com propriedades aleatórias quando inicializada por uma semente (condição inicial). De fato, progresso significativo foi relatado, por exemplo, com os CB-PRNGs baseados em equações de diferenças e mapas de recorrência (119, 124–126, 131, 132, 174–176) assim como em autômatos celulares caóticos. (49, 177, 178)

Há três décadas foi proposta a teoria do caos como uma nova abordagem para criar sistemas de criptografia, a partir disso, muitos pesquisadores têm se interessado na relação entre o caos e a criptografia, aproveitando-se de muitas das propriedades dos sistemas caóticos com a finalidade de empregar estas vantagens para criar sistemas de criptografia.

Na Tabela 2 é apresentada uma linha de tempo na com as principais contribuições na criptografia, seja como métodos criptográficos ou PRNGs, assim como as contribuições que quebraram estes.

3.3.1 Desvantagens do mapa logístico

Um dos trabalhos mais marcantes foi introduzido no começo dos anos 90s. Em 1998, Baptista (109) propôs um algoritmo de encriptação que aproveita as propriedades do mapa logístico. O algoritmo proposto divide o espaço de fase em ε -intervalos associadas ao número de unidades de um alfabeto. Para criar o texto cifrado, considera-se o número de iterações realizadas pelo mapa logístico, de modo que a órbita passe da condição inicial (chave) e alcance ε -intervalos associado a esse caractere. Porém, como percebido da Tabela 2, devido ao jogo da criptografia/criptoanálise, foram detectadas várias falhas na implementação deste algoritmo. (115) Eis que soa mais forte o ditado: “todo algoritmo é presumido inocente até comprovado o contrário”. (138, p. 38)

Naquele trabalho, a relação entre caos e criptografia foi questionada, criticada e desencorajada após a descoberta de certas falhas desse sistema. (115, 130) Alguns pesquisadores expressaram certas dúvidas sobre as propriedades do mapa logístico no contexto de criptografia, como (i) distribuição de probabilidade não uniforme (padrão U), onde uma distribuição platô é esperada (115); ii) dependência do parâmetro de controle, consequentemente levando a janelas periódicas (115); iii) amostras suficientemente grandes de texto cifrado que podem estimar o parâmetro (115); e (v) órbitas de comprimentos de ciclo curtos onde é esperada longa periodicidade dependendo das limitações da máquina (130); (vi) degradação de problemas do sistema caótico digital. (180)

A Figura 14a mostra o diagrama de bifurcação do mapa logístico para parâmetros $\mu \in [3.2, 4]$. Nesta figura as cores representam a normalização das frequências na qual os valores das órbitas caem em cada compartimento. A normalização foi feita apenas considerando os valores para cada μ correspondente. Para ilustrar ainda mais, a Figura 14b apresenta as curvas de distribuição de frequência para certos parâmetros $\mu = \{3.2, 3.4, 3.6, 3.8, 4\}$. Observe que para o caso em que $\mu = 4$ surge o clássico padrão U que segue a função densidade de probabilidade invariante do mapa logístico (36, p. 32), dada por

$$\rho(x) = \frac{1}{\pi\sqrt{x(1-x)}}. \quad (3.4)$$

Neste caso em particular, pode ser observada uma maior frequência nas bordas $x \in [0, 0.1]$ e $x \in [0.9, 1]$ do que na região central.

Tabela 2 – Estado da arte da criptografia e a pseudo-aleatoriedade baseada na teoria do caos ao longo dos anos.

1986	Wolfram usou o autômato celular elementar com regra 30 como sistema criptográfico. (103) Atualmente a linguagem de programação Wolfram Mathematica usa esta mesma estratégia. (179)
1989	Matthews (104), propôs o primeiro PRNG baseado no mapa logístico com propósitos de criptografia.
1989	Wheeler (105), quebrou o algoritmo de Matthews (104) ao encontrar certa degradação nele.
1991	Wheeler & Matthews (106), juntaram esforços para propor um novo algoritmo de encriptação caótico.
1991	Habutsu et al. (107), propuseram um cifrador de bloco baseado no mapa da tenda.
1991	Biham (108), quebrou o algoritmo do Habutsu et al. (107), por meio de um ataque de texto cifrado.
1998	Baptista (109), propôs um algoritmo de encriptação usando o mapa logístico aproveitando da propriedade de transitividade.
1999	Álvarez et al. (110), propuseram um algoritmo de encriptação usando o mapa da tenda.
2000	Álvarez et al. (111), quebraram o próprio algoritmo proposto em 1999. (110)
2001	Shujun Li et al. (112), melhoraram o algoritmo de Álvarez et al. (110) contra a criptoanálise.
2001	W. Wong et al. (113), propuseram uma otimização do algoritmo do Baptista. (109)
2002	K. Wong (114), propôs uma outra variante no algoritmo do Baptista (109) usando tabela dinâmica.
2003	Álvarez et al. (115) quebraram o algoritmo do Baptista. (109)
2004	Álvarez et al. (116) quebrou o algoritmo do W. Wong et al. (113) No mesmo ano, o mesmo time Álvarez et al. (117) também quebraram o algoritmo de K. Wong. (114)
2006	Addabbo et al. (118), propuseram um PRNG baseado no mapa da tenda.
2009	Patidar et al. (119) propuseram um PRNG baseado no mapa logístico.
2009	Ali & Nejib (120) estudaram o mapa logístico como um PRNG.
2009	Cecen et al. (121) também estudaram o mapa logístico, e aprofundaram na análise da região $\mu = [3.5599, 4)$.
2010	Pareek et al. (122) usaram a combinação de dois mapas da tenda com diferentes parâmetros acoplados.
2010	Marco et al. (123) propuseram um algoritmo de encriptação baseado no atrator de Lorenz.
2011	Dabal & Pelka (125) propuseram um sistema criptográfico baseado no mapa logístico usando $\mu = 4$ implementado em FPGA.
2011	Machicao et al. (49), propuseram um algoritmo de criptografia baseado nas propriedades caóticas do autômato celular da família <i>Life-Like</i> .
2011	Behnia et al. (126) propuseram um PRNG baseado numa generalização do mapa logístico. Nesse mesmo trabalho criticaram as falhas do mapa logístico original.
2011	Addabbo et al. (127) fizeram uma revisão dos principais CB-PRNGs e propuseram um gerador baseado na congruência não linear.
2011	Luca et al. (128) analisaram o mapa da tenda como PRNG.
2012	Li et al. (129) propuseram uma melhora no gerador usando o mapa logístico por meio de reinserção de sementes (condição inicial).
2012	Persohn & Povinelli (130) estudaram a periodicidade do mapa logístico como PRNG como consequência da precisão finita.
2013	Lequan et al. (131) fizeram uma análise comparativa dos principais PRNGs da literatura.
2014	François et al. (132) propuseram um PRNG baseado na combinação de três mapas caóticos.
2015	Machicao et al. (5) encontraram padrões nos modos de operação usados pelos sistemas criptográficos.
2017	Machicao & Bruno (133), melhoraram a pseudo-aleatoriedade do mapa logístico (proposta desta tese).

Fonte: Elaborada pela autora.

Figura 14 – (a) Histograma normalizado do mapa logístico em 3D. O eixo vertical (eixo X) representa o espaço de fase $x_t \in [0, 1]$. O eixo horizontal corresponde ao parâmetro $\mu \in [3.2, 4.0]$. Ambos os eixos foram divididos em 40 compartimentos, respectivamente. As barras coloridas correspondem às intensidades das frequências de 10^5 valores de x_t com respeito a μ . (b) Histograma normalizado para vários parâmetros μ . As médias e desvio padrão (linha contínua e barra de erro sombreada) foram calculadas a partir de 100 condições iniciais aleatórias. Para ambas as figuras foram descartadas as primeiras 1000 iterações (tempo transitente).

Fonte: Elaborada pela autora.

3.4 Testes de aleatoriedade

Tipicamente, muitos dos requisitos para avaliar as propriedades aleatórias dos PRNGs são feitas por meio de testes de aleatoriedade: sejam de natureza empírica (estatísticos), ou teóricos. (138, p. 80) ou por visualização (por exemplo teste espectral). Entre os testes de aleatoriedade mais usados e reconhecidos como padrão da indústria estão o DIEHARD (181) e NIST[‡]. (171) Estes testes são usados para analisar os padrões de distribuição das sequências de números gerados pelos PRNGs.

Tanto o DIEHARD como o NIST possuem uma coleção (bateria ou *suítes*) de testes desenhados para analisar sequências de bits, seja de forma local ou parcial, de maneira que (i) processam cada bit em um fluxo de bits, (ii) processam blocos de m -bits e (iii) processam m -bits partes. Estes testes também preocupam-se com questões de sobreposição (ou não) dos bits e/ou dos janelamentos, além disso, quase todos os testes são parametrizados, seja

[‡] O *National Institute of Standards and Technology* (Instituto Nacional de Padrões e Tecnologia, em português) é uma agência governamental não regulatória da administração de tecnologia do Departamento de Comércio dos Estados Unidos. Em 1997, o NIST (pelas siglas em inglês) recomendou usar a bateria de testes estatísticos compilados no relatório técnico *NIST Special Publication 800-22: A statistical test suite for random number generator for cryptographic applications* (171) para verificar se um sistema se porta como um gerador de números aleatórios.

pelo comprimento de bits ou dos blocos.

Um teste estatístico é formulado para testar uma hipótese nula (H_0 : a sequência binária sendo testada é dita suficientemente aleatória, do ponto de vista estatístico), associada a uma hipótese alternativa (H_1 : a sequência é dita não-suficientemente aleatória). Cada teste retorna uma probabilidade p -valor entre $[0, 1]$ fornecida pelos dados contra a hipótese nula sob um nível de significância (α). Para cada teste aplicado, é derivada uma decisão que aceita ou rejeita a hipótese nula. Os testes consideram que *passou* quando o p -valor está entre $0 \ll p \ll 1$, ou seja, a sequência é dita como aleatória; caso contrário, *falhou* quando o p -valor está próximo de 1 ou 0.

Já a bateria de testes DIEHARD, proposta por Marsaglia (181), consta de 18 testes estatísticos, os quais são: *Birthday spacings*, *Overlapping permutations*, *Ranks 31x31*, *Ranks 32x32*, *Ranks 6x8*, *Monkey 20bits words*, *OPSO*, *OQSO*, *DNA*, *Count 1s stream*, *Count 1s specific*, *Parking lot*, *Minimum distance*, *RandomSpheres*, *Squeeze*, *Overlapping sums*, *Runs* e *Craps*; cujas descrições são detalhadas na Tabela 27 do Apêndice A.

A bateria de testes NIST fornece uma suíte de 15 testes estatísticos (171), e funciona da maneira similar ao DIEHARD. Os testes são: *Frequency*, *Block Frequency*, *Cumulative sums*, *Runs*, *Longest run*, *Rank*, *FFT*, *Non-overlapping template*, *Overlapping template*, *Universal*, *Approximated entropy*, *Random excursions*, *Random excursions* (variante), *Serial* e *Linear complexity*, cujas descrições são detalhadas na Tabela 28 do Apêndice A.

3.4.1 Limitações dos atuais testes de aleatoriedade

O descaso na construção de um bom gerador de números pseudo-aleatórios (PRNG) pode tornar-se um problema crucial. (147, 169) Sobretudo quando trata-se de aplicações mais rigorosas como é a simulação de Monte Carlo, criptografia e segurança. (148) Consequentemente, os testes de aleatoriedade são de grande importância para avaliar se o gerador é suficientemente aleatório.

No entanto, os próprios desenvolvedores das baterias de testes DIEHARD (181) e NIST (171) enfatizam que os testes devem ser usados apenas como *indicadores* qualitativos de aleatoriedade. O matemático Hellekalek (149) considera que nenhum destes testes pode certificar um PRNG como apropriado para alguma aplicação em particular. Estas observações só levantam mais dúvidas que muitas vezes passam despercebidas e/ou são ignoradas. A seguir são citadas várias limitações e diferentes sugestões encontradas na literatura para que essas deficiências sejam resolvidas. Muitas destas perguntas foram levantadas por Kenny (144) e continuam abertas. Além disso, baseado na experiência obtida ao longo deste doutorado, a autora aponta alguns comentários nos próximos parágrafos.

- Qual teste de aleatoriedade usar?

Atualmente, diversas baterias de testes são encontradas na comunidade como o: DIEHARD (181) e NIST (171), mencionadas anteriormente, assim como ENT (182), TestU01 (183), DIEHARDER (184), entre outras. Além disso, muitos dos testes incluídos nestas baterias, são similares ou compartilham suas estratégias, por exemplo, o DIEHARDER é uma versão otimizada dos testes DIEHARD e NIST. Já o ENT, é um teste de aleatoriedade obsoleto hoje em dia, baseado em entropia, qui-quadrado e simulação Monte Carlo dos valores de π .

Observe que, embora o NIST não tenha lançado uma versão mais atualizada desde 2001, esta bateria é uma das mais respeitadas no mercado por ser a mais rigorosa em termos de criptografia e segurança. Analogamente, já que o DIEHARD é uma das baterias mais citadas na literatura, os resultados estatísticos poderiam ser diretamente comparados com outros referentes na literatura. Porém, as justificativas para a escolha dos testes são mais em termos de praticidade do que em termos de eficiência ou qualidade para avaliar os PRNGs em si. Nesse contexto, comumente, recomenda-se usar vários testes, com o intuito de que se um gerador passa por uma grande variedade de testes, a confiança em sua pseudo-aleatoriedade aumenta.

Além disso, não há limites para o número de testes que podem ser concebidos. Se dada uma sequência, sob avaliação, se comporta aleatoriamente em relação aos testes $T_1, T_2, T_3, \dots, T_n$ não se pode ter certeza de que não falhará quando submetido a um teste adicional T_{n+1} . No entanto, cada teste aceito produz mais confiabilidade de que a sequência é pseudo-aleatória (138, p. 38), na prática, são aplicados vários tipos diferentes de testes estatísticos, e se estes passam satisfatoriamente, é aceita como aleatória, ou seja, prevalece o ditado “é presumido inocente até comprovado o contrário”. (138, p. 38)

- O que deve ser avaliado: as sequências ou o gerador como um sistema?

É comum encontrar na literatura que a construção de PRNGs clássicos baseiam-se no uso de S-boxes, shifts, rotações de bits, XORs, somas, e muitas vezes de forma nada intuitiva, pois, carecem de explicações ou lógica, além de serem operações intrincadas. No entanto, os testes de aleatoriedade apenas encarregam-se de avaliar a aleatoriedade nas sequências geradas e não na função geradora, ou seja, os testes apenas avaliam amostras de sequências geradas, um subconjunto muito pequeno em comparação ao espaço de amostragem do gerador.

- Qual tamanho da sequência a ser analisada?

No caso do DIEHARD (181), é requerida sequências de comprimento maior que 2.5 milhões de inteiros de 32-bits, o que corresponde a 10MB de dados. Entretanto, NIST requer 800 milhões de bits, o que corresponde a 100MB. Esta diferença entre DIEHARD e NIST é devido ao tipo de avaliação no nível de fluxo de bytes e de

bits. Além disso, outra diferença nos tamanhos de dados é que no NIST é requerido diferente fluxo de bits para cada teste específico, além disso, este divide o arquivo em m -bits partes. Entretanto, surge a dúvida se apenas 10MB, ou 100 MB são suficientes. Muitas aplicações hoje em dia requerem facilmente mais do que GBs de dados.

- Quantas sequências testar? É bom testar apenas um arquivo ou vários?

Em geral, os testes de aleatoriedade destinam-se a avaliar o gerador, e não arquivos de números possivelmente aleatórios. Por esta razão, várias sequências devem ser geradas e avaliadas, para ter um resultado qualitativo do gerador na média. Nesta tese é adotado o critério de usar várias sementes aleatórias e dar como resposta a média de um número total de arquivos.

- Há maneiras de se fazer uma estimativa quantificável da aleatoriedade?

Visto que os testes estatísticos são baseados em argumentos hipotéticos binários (falhou/passou) e não em uma forma de quantificá-los. Muitos PRNGs podem passar em todos os testes de aleatoriedade, porém, não há maneiras de indicar qual PRNGs é melhor que outro. Isto pode ser entendido como se existisse uma régua que mede em escala de metros, no entanto, sua precisão não atinge o nível de centímetros, pelo que há falta de novas estratégias para classificar geradores.

Por todas estas razões, é de grande importância desenvolver métodos que tentem revelar a presença de distribuições não óbvias em sequências de números produzidos por um gerador.

Vários dos critérios usados na suíte de testes NIST foram questionados e estão abertos à discussão (144) como por exemplo o número de repetições em que cada teste deve ser executado, a realização do teste do qui-quadrado nos p -valores, e portanto também questiona-se sua eficácia. Além disso, vários erros encontrados no manual do NIST, bem como a falta de clareza na forma da implementação, ou que muitos dos testes específicos da suíte carecem de explicações razoáveis, assim como a interpretação dos resultados precisa ser melhor esclarecida.

Hoje em dia, sabe-se que tais testes de aleatoriedade não devem ser considerados como um indicador final, mas sim como um detector de desvios de aleatoriedade. Uma vez que esses testes são suportados estatisticamente em valores de aprovação e falha em vez de valores em numéricos que podem levar a uma medida de uma fraqueza para uma qualidade de força. Portanto, ainda há uma falta de métodos que mostram uma medida de uma qualidade PRNG em termos de testes de pseudo-aleatoriedade. (4) Porém, em trabalho recente (5) foi sugerido que certos algoritmos contemporâneos de criptografia possam produzir padrões. Assim, esta tese visa oferecer uma nova alternativa para explorar os padrões, e por consequência as fraquezas de sistemas similares.

Contudo certos PRNGs conseguem qualidades ideais para os propósitos para os quais foram desenhados, porém existe uma falta de medidas quantitativas que possam indicar o espectro de um gerador “fraco” até um “forte”, e tais vantagens motivam a busca por boas fontes de aleatoriedade, razão pela qual esta tese encaixa-se na abordagem dos PRNGs e o reconhecimento de padrões que será discutido no Capítulo 6.

4 ALEATORIEDADE NAS PROFUNDEZAS DO CAOS

Antes de começar este capítulo, o leitor é convidado a fazer parte de um experimento simples. Observe o diagrama (na parte superior) da Figura 15, a qual mostra o bem conhecido conjunto de Mandelbrot (29)*.

Figura 15 – A profundidade infinitesimal no caos. De cima para baixo, e da esquerda para direita é mostrado o conjunto de Mandelbrot. Cada janela (branco) indica o escalamento (zoom) mostrando diferentes padrões que não se repetem.

Fonte: Elaborada pela autora.

* O conceito de fractal foi introduzido por Mandelbrot no 1997, para descrever padrões complexos que não possuem uma dimensão euclidiana e sim um fractal. Os fractais caracterizam-se pela auto-similaridade em diferentes escalas, por exemplo árvores, rios, zonas costeiras, montanhas, nuvens, etc. Assim, os fractais estão relacionados com o caos porque são sistemas complexos que têm propriedades definidas. O conjunto de Maldelbrot segue a equação $z_{n+1} = z_n^2 + c$, com $z_0 = 0$.

4.1 Mergulhando nas profundezas do caos

Como o leitor deve ter percebido, na imagem original do conjunto de Mandelbrot é exibido um padrão interessante e bem conhecido, mas limitado a esse padrão de visualização. No entanto, quando esse padrão é ampliado repetidamente, uma grande quantidade de padrões complexos começam a ser distinguidos e, efetivamente, são nessas ampliações onde a dissimilaridade aparece e há caos no conjunto de Mandelbrot. Tomando este exemplo como motivação, é necessária uma maior precisão computacional para explorar as profundezas de um sistema caótico e, conseqüentemente, investigar as propriedades de pseudo-aleatoriedade destes.

O objetivo deste capítulo é desenvolver os conceitos de profundezas no caos por meio de uma análise dos dígitos de precisão dos pontos gerados pelo sistema caótico. Acreditamos que a partir desta visualização é possível compreender melhor como surge a aleatoriedade deste. Este capítulo está dividido em duas partes. Na primeira parte foi proposto o modelo chamado k -mapa logístico, o qual analisa as escalas de profundidade nos dígitos de precisão no mapa logístico. O k -mapa logístico foi estudado por meio de diferentes ferramentas de visualização como o diagrama de evolução temporal, o diagrama de bifurcação, o diagrama de *cobweb*, o diagrama de Poincaré e histograma de frequências.

Com base neste modelo de análise das profundezas do caos, na segunda parte do capítulo foi proposto um gerador de números pseudo-aleatórios, que por meio do parâmetro k , consegue parametrizar a qualidade pseudo-aleatória desse sistema. O PRNG baseado no k -mapa logístico, chamado de “padrão ouro”, servirá para a modelagem de sequências pseudo-aleatórias que foram analisadas no contexto de reconhecimento de padrões no Capítulo 6. Adicionalmente, esta mesma abordagem foi aplicada no mapa da tenda, cujos resultados são bastante similares ao do mapa logístico, a qual é detalhada no Apêndice B.

4.1.1 A sensibilidade dos dígitos menos significativos

É bem conhecido que a sensibilidade às condições iniciais é um dos pilares da teoria do caos (Capítulo 2) e essa mesma é uma das propriedades básicas para a geração de números pseudo-aleatórios (Capítulo 3). Paradoxalmente, o verdadeiro potencial do caos está sendo descartado. Hoje em dia, muitos pesquisadores negligenciam que o caos, numericamente, depende da profundidade infinitesimal dos dígitos de precisão, pelo qual a aritmética de ponto flutuante é uma questão de grande repercussão para a teoria do caos, mas ignorada computacionalmente. Certos sistemas ditos sombreáveis geram trajetórias numéricas que embora erradas sombreiam as trajetórias verdadeiras por um longo tempo. Os dígitos menos significativos no entanto ignorados, justamente este esta tese mostrará, podem ser explorados para criar sistemas criptográficos mais aplicantes. De fato, diversos trabalhos na literatura (130, 185–189) focaram nos diferentes quesitos com respeito aos dígitos de precisão nos sistemas dinâmicos.

As limitações computacionais (discretização e erros de truncamento) podem fazer com que as órbitas computadas se afastem radicalmente, mas como consequência do erro numérico, e não da sensibilidade dinâmica do sistema. Para exemplificar estas limitações, veja, por exemplo, a seguinte órbita $\mathcal{O}(\mu, x_0)$ com parâmetro $\mu = 4$ e condição inicial x_0 , usando o mapa logístico com Equação (4.1) (apresentado também no Capítulo 2).

$$x_{t+1} = \mu x_t(1 - x_t). \quad (4.1)$$

$$x_0 = 0.36,$$

$$x_1 = 0.92160,$$

$$x_2 = 0.289013760,$$

$$x_3 = 0.82193922612264960,$$

$$x_4 = 0.585420538734197957594151502479360,$$

$$x_5 = 0.97081332624943754555426458029594392767188359506591627109109596160.$$

Observe que o número de dígitos de precisão diferentes de zero expande-se com relação ao tempo t , isto porque trata-se de uma equação quadrática. Por exemplo, x_0 contém 2 dígitos decimais após a vírgula, x_1 contém 4, x_2 contém 8, x_3 contém 16, e assim por diante. É evidente que, computacionalmente, é inviável calcular as órbitas com a mesma precisão analítica após um certo número de iterações. Porém, apesar dessas limitações poderem ser contornadas, é importante explorar o comportamento e a dinâmica dos sistemas caóticos considerando que os dígitos menos significativos dos pontos das órbitas caóticas são de fato importantes, como o leitor pode ter percebido pelo experimento inicial.

Definição 4. Sejam k os dígitos de precisão de cada ponto x^t de uma órbita original $\mathcal{O}(\mu, x_0) = \{x_0, x_1, \dots, x_t\}$ gerada por um sistema dinâmico determinístico. Especificamente, os k dígitos à direita do separador decimal de cada valor x_t são descartados a fim de compor um novo valor x_t^k . Por exemplo, a partir de 0.847392 é possível compor: 0.47392, 0.7392, 0.392, 0.92 e 0.2, usando $k = 1, 2, \dots, 5$ -dígitos, respectivamente.

Definição 5. Seja $\mathcal{O}^k(\mu, x_0) = \{x_0^k, x_1^k, \dots, x_t^k\}$ a órbita resultante derivada da órbita original $\mathcal{O}^{k=0}(\mu, x_0)$, ou simplesmente $\mathcal{O}^{k=0}$. O novo valor x_t^k é formado mantendo os k -dígitos à direita do separador decimal correspondentes ao ponto da órbita original $x_t \in \mathcal{O}^{k=0}$ (dado na definição anterior). Logo, a forma generalizada é dada pela Equação (4.2), em que $\lfloor \cdot \rfloor$ é a função piso.

$$x_t^k = \phi_k(x_t) = x_t 10^k - \lfloor x_t 10^k \rfloor, \quad (4.2)$$

Para exemplificar o conceito dos k -dígitos, considere dois pontos $x_t = 0.5614275$ e $x_{t+1} = 0.9466392483$ de uma órbita original $\mathcal{O}^{k=0}$. Assim, $\phi_{k=1}(x_t) = 0.614275$, pois, deve-se avançar um algarismo à direita após o separador decimal, enquanto que $\phi_{k=1}(x_{t+1}) =$

0.466392483. Um outro exemplo é $\phi_{k=3}(x_t) = 0.4275$, $\phi_{k=3}(x_{t+1}) = 0.6392483$, e assim por diante.

Claramente, se deve garantir em todo momento que k seja menor que a precisão computacional e esta última deve ser a maior possível, de tal forma a garantir que k dígitos sejam extraídos das órbitas. Observe que para todos os experimentos realizados neste capítulo, e nos subsequentes com respeito ao caos, foram usados 500 dígitos de precisão por meio da biblioteca Apfloat (190) que oferece alta precisão computacional.

Com base na Equação (4.2) é proposta uma abordagem geral, a qual pode ser aplicada diretamente a quase qualquer mapa caótico uni-dimensional no intervalo unitário (ou seja, $x_t \in [0, 1]$). Esta tese focou especialmente no mapa logístico (ver Seção 4.2) e no mapa da tenda (ver Apêndice B.1), isto porque ambas são equações simples e bem conhecidas na teoria do caos. No entanto, esta abordagem pode ser adaptada para outros sistemas caóticos que não satisfaçam a condição do espaço unitário, como o mapa de Hénon, o mapa Gaussiano, o atrator de Lorenz, entre outros. Para isso, devem ser ajustados adequadamente os respectivos espaço de fase para o intervalo unitário, o que implicaria em compor novos valores x_t^k no mesmo intervalo do espaço de fase.

4.2 k -mapa logístico

Definição 6. Definimos o k -mapa logístico como uma função para obter órbitas geradas a partir de uma órbita original do mapa logístico $\mathcal{O}^{k=0}(\mu, x_0) = \{x_0^k, x_1^k, \dots, x_t^k\}$, previamente obtida, usando a Equação (4.1), e que a partir desta podem ser calculadas órbitas resultantes $\mathcal{O}^k(\mu, x_0)$. Desta forma, a nova órbita preserva o parâmetro $\mu \in [0, 4]$ e o espaço de fase no intervalo unitário $[0, 1]$. Para simplificar as notações, o termo $k = i$ ou k_i é usado indistintamente quando refere-se a uma órbita do k -mapa logístico, em que k_0 é a órbita original.

A seguir são analisadas as propriedades dinâmicas do k -mapa logístico, dando ênfase nos aspectos teóricos, por meio da análise de comportamento assintótico e técnicas de visualização como o diagrama de evolução temporal, o diagrama de bifurcação, o diagrama de *cobweb*, o diagrama de Poincaré e histograma de frequências. Por fim, também são analisadas as órbitas periódicas e seus períodos, transientes e a análise de sensibilidade às condições iniciais através do expoente de Lyapunov, o que pode ser usado como um indicador de estabilidade e do tipo de órbitas gerada pelo k -mapa logístico, o que a seguir chamamos de estabilidade de Lyapunov.

4.2.1 Análise da dinâmica do *k*-mapa logístico

4.2.1.1 Diagrama de evolução temporal

Apesar da simplicidade da Equação (4.2) e a Definição 6, a dinâmica do *k*-mapa logístico pode mudar em termos da sua estabilidade de Lyapunov para cada valor de *k*. Veja na Figura 16, na qual é mostrada a dinâmica de evolução do *k*-mapa logístico, correspondente às órbitas de k_0 (original) até k_4 durante as primeiras $t = 100$ iterações. Nesta figura, cada gráfico representa as evoluções temporais para diferentes parâmetros de controle μ (de cima para baixo) que correspondem aos três comportamentos de estabilidade de Lyapunov, ou seja, assintoticamente estáveis, periódico e caótico, respectivamente.

A Figura 16a mostra as órbitas com parâmetro $\mu = 2.85$ que são assintoticamente estáveis, no qual o número de iterações antes da convergência para um ponto fixo é aumentado sucessivamente conforme *k* aumenta. Em seguida na Figura 16b, é interessante observar que quando a órbita original (k_0) é periódica, as órbitas derivadas também mantêm essa periodicidade, porém com maior comprimento das janelas (cinza) conforme *k* aumenta. Finalmente, na Figura 16c, em cada gráfico são mostradas duas órbitas com condições iniciais x_0 e x'_0 muito próximas, usando o parâmetro $\mu = 4$ que leva ao regime caótico. Nela pode ser observado que o tempo transiente τ , em que ambas as curvas divergem, é reduzido aproximadamente de 31 (k_0) para 17 (k_4) iterações. Em conclusão, esta figura mostra que o *k*-mapa logístico tem um transiente mais longo que o mapa original (Figura 16a), que as órbitas periódicas aumentam seu período (Figura 16b) e a sensibilidade às condições iniciais aumenta (Figura 16c).

4.2.1.2 Diagrama de bifurcação

A caracterização da dinâmica global das órbitas geradas pelo *k*-mapa logístico pode ser visualizada por meio do diagrama de bifurcação deste. Um fenômeno interessante é observado ao comparar o diagrama de bifurcação original (k_0) com os diagramas de bifurcação k_1 , k_2 , e k_3 apresentados na Figura 17. Nele observa-se que os padrões caóticos intercalados predominantemente nas regiões $\mu \in [3.57, 3.828]$ e $\mu \in [3.86, 4]$ seguem um padrão zigue-zague (k_1) e começam a desaparecer gradualmente até se tornarem aparentemente aleatórios (k_3). No entanto, as “janelas periódicas”, que podem ser facilmente visualizadas como as tiras brancas do diagrama de bifurcação (por exemplo, intervalos entre $[3.4, 3.57]$ e $[3.828, 3.86]$), permanecem para todas as demais órbitas do *k*-mapa logístico. Portanto, o *k*-mapa logístico preserva a periodicidade.

Em relação à região de dobramento de período, pode ser observado que os padrões ficam mais preenchidos conforme *k* aumenta, isto como consequência de que os períodos do *k*-mapa logístico são maiores. Deve-se mencionar que esses fenômenos também foram observados no diagrama de evolução temporal (ver Figura. 16b). O comportamento em zigue-zague também pode ser visto na bifurcação do esqueleto na Figura 17b, uma vez que

(a) $\mu = 2.85$ (b) $\mu = 3.57$ (c) $\mu = 4$

Figura 16 – Evolução temporal de duas órbitas do k -mapa logístico (Equação (4.2)) com condições iniciais $x_0 = 0.4587525281$ (linha preta) e $x'_0 = 0.4587525282$ (linha cinza) usando os parâmetros (a) $\mu = 2.85$ (b) $\mu = 3.57$ e (c) $\mu = 4.0$, evoluídas durante $t = 100$ iterações (eixo horizontal), considerando k_0 até k_4 (de cima para baixo). As marca d'água (cinza) representam (b) o comprimento da periodicidade.

Fonte: Adaptada de MACHICAO; BRUNO (133)

suas curvas seguem um padrão em zigue-zague (k_1 e k_2) e vão saturando-se rapidamente enquanto k aumenta (k_3).

A análise da evolução temporal (Figura 16) e dos diagramas de bifurcação (Figura 17) mostram que o k -mapa logístico preserva a dinâmica global do mapa logístico original. Isto porque as regiões caóticas, periódicas e estáveis são preservadas, indepen-

Figura 17 – (a) Diagrama de bifurcação do k -mapa logístico correspondente às órbitas k_0 (original), k_1 , k_2 e k_3 com condição inicial $x_0 = 0.4679$. O eixo horizontal representa o parâmetro de controle entre $3.4 \leq \mu \leq 4.0$ (passos de 0.001). O eixo vertical mostra os pontos x_t ao longo do tempo, a partir da mesma condição inicial, durante 10^5 iterações (desconsiderando o tempo transiente $t = 200$) para cada parâmetro. (b) As curvas sobrepostas $P_n(\mu)$ correspondem ao esqueleto da bifurcação (linhas coloridas), correspondente à Equação (2.7).

Fonte: Adaptada de MACHICAO; BRUNO (133)

dentemente do parâmetro k . Porém, repare na diminuição do número de iterações para a divergências ser notada, pois, dentro das regiões caóticas, o número de iterações antes de divergir diminui com k . Ao mesmo tempo que, dentro das regiões estáveis e das regiões de bifurcação, o número de iterações que convergem para um ponto fixo, tanto quanto o comprimento do ciclo das regiões periódicas, aumentam com k . Esta preservação da dinâmica nas regiões caóticas, periódicas e estáveis do k -mapa logístico é muito útil, pois indicam que qualquer sistema dinâmico pode ser explorado de forma convencional. Além disso, as propriedades caóticas são preservadas e inclusive são enriquecidas em termos de pseudo-aleatoriedade como será demonstrado na Seção 4.3.

4.2.1.3 Diagrama de *cobweb*

Uma outra forma de visualizar a dinâmica do k -mapa logístico, é por meio do diagrama de *cobweb*, com o qual é possível de traçar relações com as figuras mostradas anteriormente. É interessante observar este diagrama, pois, nele pode ser percorrida as trajetórias das órbitas.

A Figura 18 mostra o diagrama de *cobweb* do k -mapa logístico (de cima para baixo) para diferentes parâmetros de controle $\mu = 2.85$ (fixo), $\mu = 3.57$ (periódico) e $\mu = 4$ (caótico) para $t = 25$ iterações. A diferença do diagrama *cobweb* convencional (apresentado no Capítulo 2), nesse diagrama é mostrada a função $\phi_k(f(x_t))$ em lugar de $f(x_t)$, o qual corresponde aos limites pelos quais as órbitas transitam. Repare que para k_0 , a função do mapa logístico corresponde à parábola invertida do respectivo parâmetro de controle μ , porém, conforme k é aumentado, esta muda radicalmente (este fenômeno será melhor analisado na Seção 4.3.2). Observando os três casos mostrados em ordem crescente de k , é possível ver como as rotas dos pontos das órbitas colidem com as bordas da função do mapa.

Na Figura 18a, observa-se um ponto fixo estável que corresponde a uma espiral interna. Já que foram mostradas poucas iterações, esta espiral é melhor observada em k_0 . Nela é possível perceber que a sequência de iterações (seguindo as setas) converge para os respectivos pontos fixos com uma velocidade que depende de k . Conforme k aumenta, o número de iterações para corrigir ao ponto fixo aumenta. O k -mapa logístico apresenta pontos periódicos que são menos estáveis que no mapa original. De forma similar, com o parâmetro de controle $\mu = 3.57$ (ver Figura 18b), o tamanho dos períodos aumenta conforme k aumenta. Finalmente, na Figura 18c, quando $\mu = 4$, no ponto caótico máximo do mapa logístico, os pontos divergem rapidamente, mostrando uma espiral externa que comporta-se arbitrariamente. Repare também, nesta figura, que as trajetórias pelas quais as órbitas transitam parecem mais aleatórias conforme k aumenta, esta propriedade especificamente será tratada nas seguintes seções. Além disso, nos três casos, é observado que o número de iterações para estabilizar um ponto de equilíbrio, ou em um ponto periódico aumenta conforme k é aumentado. A sensibilidade às condições iniciais aumenta no regime caótico. Note que, de forma similar ao diagrama de evolução temporal (Figura 16), a dinâmica global permanece.

4.2.1.4 Aproximação do Expoente de Lyapunov do k -mapa logístico

Em geral, para medir a sensibilidade às condições iniciais é usado o expoente de Lyapunov λ , que pode ser calculado analiticamente usando a Equação (2.3), introduzida no Capítulo 2. Porém, para poder estimar o LE para o k -mapa logístico, uma outra estratégia foi assumida. Aqui é proposta uma forma de aproximar o expoente de Lyapunov para a nova órbita construída $\mathcal{O}^k$ usando o método de Wolf (38), que vem do campo de análise

Figura 18 – Diagrama de *cobweb* do k -mapa logístico (azul) com condição inicial $x_0 = 0.4587$ para a órbita original $\mathcal{O}^{k=0}$ iteradas por $t = 25$ iterações. De cima para baixo k_0 (original) k_1 , k_2 , e k_3 . Da esquerda para a direita são mostrados diferentes parâmetros de controle (a) $\mu = 2.85$ (atrator de ponto fixo), (b) $\mu = 3.57$ (atrator periódico) e (c) $\mu = 4.0$ (atrator caótico), respectivamente. Além disso, os pontos na curva de função são mostrados pela coordenada $(x_0, f(x_0))$ (cinza). Linha diagonal com coordenadas $(f(x_0), f(x_0))$ (cinza).

Fonte: Elaborada pela autora.

de séries temporais, como mostrado na Figura 19.

Definição 7. Dadas duas órbitas no espaço de fase com separação inicial da forma

$$d_t = |\phi_k(f(x_t + \epsilon)) - \phi_k(f(x_t))|,$$

que divergem ao longo prazo em razão da taxa dada por $\epsilon e^{t\lambda(x_0)}$. Definimos o expoente de Lyapunov para mapas unidimensionais da forma $\lambda(\mu, x_t^k)$ seguindo a Equação (4.3):

$$\lambda(\mu, x_t^k) = \frac{1}{T\Delta t} \sum_{t=1}^T \ln \left(\frac{d_{t+\Delta t}}{d_t} \right), \quad (4.3)$$

Figura 19 – Algoritmo de Wolf para computar expoentes de Lyapunov a partir de uma série temporal original (vermelho) e sua correspondente trajetória fiduciária (preto) que pertencem à vizinhança mais próxima d da órbita original.

Fonte: Adaptada de WOLF. (38)

em que Δt é o número de passos de tempo na trajetória fiduciária e T é o número de vezes que verificamos a divergência definida por

$$d_{t+\Delta t} = \left| \phi_k(f^{\Delta t}(x_t + \epsilon)) - \phi_k(f^{\Delta t}(x_t)) \right|.$$

Observe que o tempo no intervalo de comprimento Δt foi amostrado, de modo que o número total de passos seja $T\Delta t$ (38), isto a fim de realizar cálculos numéricos mais adequados.

A Figura 20 mostra a aproximação do expoente de Lyapunov para o mapa logístico original (k_0), e as correspondentes k_1 , k_2 e k_3 na região $\mu \in [3.4, 4]$ usando a Equação 4.3. Esta figura apresenta os três tipos de comportamentos dinâmicos, ou seja, caótico ($\lambda > 0$), marginalmente estável ($\lambda \approx 0$) e periódico ($\lambda < 0$), os quais são detalhados a seguir.

Figura 20 – Expoente de Lyapunov $\lambda(k_0)$, $\lambda(k_1)$, $\lambda(k_2)$ e $\lambda(k_3)$ do k -mapa logístico. O eixo vertical representa o expoente de Lyapunov em função do parâmetro de controle na região $\mu \in [3.4, 4]$ (eixo horizontal) com passos de 0.0006. Os expoentes de Lyapunov foram calculados usando $T = 1000$ iterações e $\Delta t = 50$ a partir da distância inicial $d_0 = 10^{-16}$.

Fonte: MACHICAO; BRUNO (133)

Por um lado, nas regiões caóticas $\mu \in [3.57, 3.828]$ e $\mu \in [3.86, 4]$, as curvas do LE correspondentes a k_1 , k_2 e k_3 mostram que $\lambda(\mu, x_0) > 0$. De fato, houve um aumento no expoente de Lyapunov até atingir o limite superior encontrado analiticamente.

Primeiro, observando especificamente nas janelas das regiões caóticas entre $\mu \in [3.57, 3.828]$ e $\mu \in [3.86, 4]$, ou seja, nos pontos onde as curvas dos expoentes de Lyapunov são positivos, é possível perceber que as curvas correspondentes a k_1 , k_2 e k_3 vão atingindo esse limite, em quase todos os pontos dessa região, o que indica que houve um incremento em termos de caos para todas as curvas do expoente de Lyapunov $\lambda(\mu, x^k)$ quando $k > 0$.

Por outro lado, nas regiões estáveis, periódicas e na região de dobramento de período (exceto nas janelas periódicas nas regiões caóticas), o k -mapa logístico apresenta LE μ negativo. Este fato pode ser observado em todas as curvas $\lambda(\mu, x^k)$. Contudo, estes resultados só reforçam nossa suposição, anteriormente mostrada no diagrama de bifurcação e no diagrama de evolução temporal (ver Figura 16-17), de que as órbitas do k -mapa logístico se mantêm assintoticamente estáveis, periódicas, ou tornam-se altamente caóticas para $k > 0$.

4.3 Propriedades pseudo-aleatórias do k -mapa logístico

A motivação do estudo da dinâmica caótica do k -mapa logístico é que o caos fornece uma ótima fonte de pseudo-aleatoriedade. Assim, todos os experimentos subsequentes foram focados numa das regiões mais caóticas do k -mapa logístico fornecido pelo parâmetro $\mu = 4$ que corresponde ao maior expoente de Lyapunov, como foi introduzido no Capítulo 2 e Capítulo 3.

No contexto de PRNGs baseados no caos é comum associar os expoentes de Lyapunov positivo $\lambda(x_0) > 0$ com as propriedades pseudo-aleatórias. Esta mesma relação foi inclusive demonstrada em trabalho de Lee et al. (191) Por conseguinte, baseados nesse argumento, acessar a medida da sensibilidade às condições iniciais é de grande interesse para gerar números pseudo-aleatórios. De fato, usando a Equação (4.3) foi demonstrado que várias regiões caóticas do k -mapa logístico possuem expoente de Lyapunov positivo, como foi visto na Figura 20. Contudo, o expoente de Lyapunov positivo não é o único critério necessário para usar um sistema caótico como fonte de pseudo-aleatoriedade, pois outros critérios como a distribuição das frequências e análise de correlação x_t , x_{t+1} e x_{t+2} também devem ser considerados, os quais são detalhados nas seguintes seções.

4.3.1 Distribuição da frequência

Vale lembrar que num dos primeiros sistemas criptográficos baseado no caos, introduzido por Baptista (109), foram exploradas várias das propriedades do mapa logístico como fonte de entropia. No entanto, alguns pesquisadores (115, 130) levantaram dúvidas sobre as propriedades pseudo-aleatórias do mapa logístico, pois esta apresenta uma dis-

tribuição de probabilidade não uniforme, que para propósitos de uma boa construção de um PRNG, espera-se uma distribuição o mais platô possível. (172, 173) O padrão U desta distribuição fica evidente ao observar o primeiro gráfico da Figura 21a, a qual representa a curva da frequência da órbita original k_0 usando o parâmetro $\mu = 4$. Este padrão U é bem conhecido na teoria de sistemas dinâmicos, pois o mapa logístico segue uma função de densidade de probabilidade invariante da forma $\rho(x) = 1/\pi\sqrt{x(1-x)}$. (36, p. 32) Por isso, nas regiões fronteiras entre $x \in [0, 0.1]$ e $x \in [0.9, 1]$ observam-se frequências mais altas do que na região central.

No entanto, de cima para baixo, na Figura 21a pode ser observado que esta distribuição torna-se mais uniforme conforme k aumenta, ou seja, as propriedades pseudo-aleatórias do k -mapa logístico vão tornando-se, apesar da redundância, mais aleatórias conforme $k \gg 1$. Além disso, este processo de uniformização é ilustrado na Figura 21c, que mostra a diferença Δ entre a frequência mais alta e a mais baixa obtida para os $k_0, k_1, \dots, k_{20}$ -mapa logístico, que mostra como rapidamente que esta oscilação varia muito pouco a partir de k_4 . (133)

Figura 21 – (a) Curvas de distribuição de frequências do k -mapa logístico correspondente às órbitas k_0, k_1, k_2 e k_3 com $\mu = 4$. O eixo horizontal representa o espaço de fase $[0, 1]$ (passos de 0.002). O eixo vertical mostra as frequências para 10^4 iterações. As primeiras 1000 iterações foram descartadas (tempo transiente). As médias e desvio padrão foram calculadas a partir de 100 condições iniciais aleatórias (linha contínua e barra de erro sombreada). (b) Sobreposição das curvas de distribuição mostradas em (a) combinadas na região $[0, 0.03]$. (c) Variação da frequência entre o valor mais alto e o mais baixo (Δ) calculada a partir da distribuição de frequência das órbitas para cada valor de $k_0, k_1, \dots, k_{20}$. O gráfico interno mostra uma ampliação do mesmo gráfico na região k_3 a k_{11} .

Fonte: Adaptada de MACHICAO; BRUNO (133)

4.3.2 Diagrama de Poincaré do k -mapa logístico

Esta **melhora da pseudo-aleatoriedade** (133) também pode ser observada por meio do diagrama de fases, ou diagrama de Poincaré. A Figura 22 apresenta a relação bi- e tri-dimensional de x_t^k com respeito a x_{t+1}^k e x_{t+2}^k , respectivamente, para as órbitas do k -mapa logístico com parâmetro de controle $\mu = 4$ (a versão colorida é ilustrada na Figura 23).

Na Figura 22 observa-se que o diagrama de fases da órbita original k_0 mostra a clássica parábola invertida do mapa logístico com um valor máximo em $x = 0.5$, cujo padrão remanesce no esboço tridimensional. Em seguida, de cima pra baixo, são apresentados os diagramas de fase de k_1 até k_4 , pode-se observar a transformação do padrão de parábola em padrões zigue-zague, os quais vão sendo progressivamente esfumados até sumirem tornando-se visualmente aleatórias, como pode ser observado a partir de k_2 em diante. De fato, observa-se que o espaço de fase está sendo preenchido enquanto k aumenta, ou seja, que o k -mapa logístico produz quase todos os valores possíveis entre $[0, 1]$, como espera-se de um bom gerador de números pseudo-aleatórios (149) e também se torna em mapeamento cada vez mais não-inversível. Portanto futuro e passado ficam cada vez mais decorrelacionados.

4.4 PRNG baseado no k -mapa logístico

Nesta tese foi proposto um PRNG baseado no k -mapa logístico usando uma das regiões mais caótica deste sistema $\mu \rightarrow 4$, ou seja, quando o parâmetro de controle tende a 4. O k -mapa logístico gera pontos entre $[0, 1]$, portanto estes valores podem ser diretamente usados como números pseudo-aleatórios, enquanto que certas transformações podem ser aplicadas para obter valores em diferentes versões, como sequências de bits, inteiros, entre outros.

As etapas do algoritmo para gerar uma sequência de n números pseudo-aleatórios de 2^b bits são as seguintes:

1. Passo 1: Escolher uma semente x_0 entre $]0, 1[$, como a condição inicial da Equação (4.1).
2. Passo 2: Iterar o sistema por um tempo transiente até garantir uma órbita caótica;
3. Passo 3: Gerar um número pelo k -mapa logístico usando a equação (4.2);
4. Passo 4: Normalizar o valor s usando uma máscara de b bits da seguinte forma:

$$s = \lfloor x_t^k (2^b - 1) \rfloor; \quad (4.4)$$

ou seja, os valores de $[0, 1]$ são convertidos a números inteiros de b bits.

Figura 22 – Diagrama de Poincaré bidimensional (coluna à esquerda) e tridimensional (coluna à direita) do k -mapa logístico para $\mu = 4$ e (a) k_0 , (b) k_1 , (c) k_2 , (d) k_3 e (e) k_4 . Os eixos horizontal e vertical mostram o espaço de fase $[0, 1]$ de x_t com relação a x_{t+1} e x_{t+2} . Cada órbita contém 10^4 pontos a partir de uma condição inicial aleatória, as primeira 1000 foram descartadas (tempo transiente).

Fonte: Adaptada de MACHICAO; BRUNO (133)

5. Passo 5: Se o tamanho da sequência ainda não foi satisfeito, volte para o Passo (2) caso contrário, pare.

4.4.1 Recomendações e limitações do PRNG k -mapa logístico

Em princípio, o PRNG baseado no k -mapa logístico pode usar qualquer parâmetro de controle que garanta regime caótico, evitando assim quaisquer região periódica ou estável. Além disso, algumas recomendações e limitações devem ser discutidas com respeito a este parâmetro. Por exemplo, é sugerido não usar $\mu = 4$, por dois motivos que seriam

Figura 23 – Diagrama de Poincaré bidimensional colorido do k -mapa logístico para $\mu = [3.6, 4]$ e k_0 , k_1 e k_2 , da esquerda para a direita, respectivamente. Os eixos horizontal e vertical mostram o espaço de fase $[0, 1]$ de x_t com relação a x_{t+1} . Cada órbita contém 10^4 pontos a partir de uma condição inicial aleatória, as primeira 1000 foram descartadas (tempo transiente).

Fonte: Elaborada pela autora.

contraproducentes para a geração de números pseudo-aleatórios. O primeiro motivo é para evitar os pontos de superposição $f(\mu, 0) = f(\mu, 1) = 0$ do mapa logístico assim como do ponto crítico $f(4, 0.5) = 1$, pois estes podem levar a valores triviais. O segundo motivo é devido à propriedade de conjugação do mapa logístico ($\mu = 4$). (36, 192)

4.4.1.1 Análise do parâmetro de controle $\mu = 4$ e $\delta = 2$

Como introduzido no Capítulo 2, os parâmetros de controle $\mu = 4$ e $\delta = 2$ do mapa logístico e do mapa da tenda, respectivamente, correspondem às regiões mais caóticas destes mapas. Porém, é fortemente sugerido que os PRNGs devem usar como parâmetros os valores mais próximos possíveis a estes, ou seja, $\mu \rightarrow 4$ e $\delta \rightarrow 2$, respectivamente. A razão pela qual sugerimos esses valores é que o mapa logístico com parâmetro $\mu = 4$ é topologicamente conjugado tanto para o mapa do deslocamento (*shift-map*) como do mapa da tenda para o parâmetro $\delta = 2$. (36, 192)

Essa conjugação significa que, para cada ponto no mapa do deslocamento G (ou com o mapa da tenda), existe um valor de conjugação especificado no mapa logístico F que reflete exatamente sua dinâmica. (192) Portanto, teoricamente, se a função de conjugação H é conhecida, existe uma possibilidade de obter valores futuros $G^n = H^{-1} \circ F^n \circ H$, o que seria contraproducente em termos de segurança criptográfica.

Desta forma, o mapa logístico e o mapa da tenda exatamente com parâmetros $\mu = 4$ e $\delta = 2$, respectivamente, não são recomendados para propósitos criptográficos. Porém, esse fato apenas afeta a órbita original, e portanto para afetar as órbitas do k -mapa logístico seria necessário saber porções maiores das órbitas x_t^k . Além disso, esses valores x_t^k já estão escondendo informação dos dígitos à esquerda, portanto se torna difícil recuperar os valores x_t da órbita original. Mas o que acontece para outros parâmetros de controle $\mu < 4$? Teoricamente, se existe tal função H , que conjugua o mapa logístico, então existe uma chance de quebrar o algoritmo. No entanto, encontrar uma conjugação H não é sempre possível. Por exemplo, para o mapa logístico não existe uma conjugação

útil para a maioria dos valores do parâmetro μ tal como argumentado por Alligood et al. (192, p. 114–124)

Com base na justificativa anterior, no âmbito dos objetivos do PRNG, limitamos o PRNG baseado no k -mapa logístico usando o parâmetro de controle $\mu \rightarrow 4$ por ser um protótipo para o regime caótico, pois, representa o máximo expoente de Lyapunov, que foi conjecturada por Collet & Eckman. (193) Assim, para os subsequentes experimentos foi considerado usar $\mu = 3.99999999$.

4.4.2 Avaliação pela bateria de testes DIEHARD

Para a avaliação do PRNG proposto para vários valores do parâmetro k foram gerados 100 arquivos, cada um contendo 11.2 MB, equivalente a 28×10^5 números gerados para cada $k = \{0, 1, \dots, 9\}$. Observe que o tamanho de arquivo usado é devido a restrições do próprio DIEHARD (181). Cada amostra foi gerada a partir de uma semente aleatória, e cada valor x_t^k de cada órbita foi transformado em inteiros usando uma máscara de 32 bits (discretização), por meio da conversão dada na Equação (4.4).

Assim, a bateria de testes DIEHARD (181) foi aplicada ao conjunto de arquivos gerados, considerando a hipótese nula “ H_0 : é uma sequência aleatória” para quando os testes retornassem resultados entre $0.0001 < \text{p-valor} < 0.9999$ (181).

Os resultados dos 18 testes estatísticos correspondentes aos 100 arquivos das configurações $k_0, k_1, \dots, k_9$ estão reportados na Tabela 3. Cada coluna, corresponde ao número de arquivos que passaram nos testes DIEHARD. Nesta tabela foram destacados em cor cinza os testes que falharam em pelo menos 50 arquivos, o qual pode ser observado no caso de k_0, k_1, k_2 e k_3 -mapa logístico. Mais especificamente, para alguns dos testes (*rankings6x8*, *monkey20bits*, *count1sStream* e *count1sSpecific*), o mapa logístico de k_0 a k_3 falhou completamente. No entanto, a maioria dos 100 arquivos foram passados com sucesso quando $k \geq 4$. Em relação aos testes com falha, por exemplo, o teste *monkey20bits* trata as sequências de bits como “palavras”, portanto, conta as palavras sobrepostas procurando por segmentos de 26 bits que não aparecem da posição 0 (à esquerda) a 31. Portanto, como o mapa logístico de k_0 a k_3 mantém uma distribuição de probabilidade (padrão em forma de U), esses segmentos de bits não são codificados no fluxo, o que, conseqüentemente, influenciou a falhar na prova.

Percebemos também que para o teste *OverlappingPermutations*, o número de arquivos que passaram oscila entre 95 até 99, e apenas atinge 100 com k_9 . Este teste é caracterizado por analisar as sequências de cinco números consecutivos, e avaliar a distribuição de todas as possíveis combinações, garantindo que estas sejam uniformes.

Além disso, percebe-se que para as sequências geradas pelo $k \geq 4$ -mapa logístico passa em todos os testes DIEHARD. Observe que estes resultados condizem com os

Tabela 3 – Resultados dos testes de aleatoriedade DIEHARD para 100 arquivos gerados pelos diferentes k -mapa logístico com sementes aleatórias. Cada coluna indica o número de seqüências (arquivos) que passaram. Testes que falharam são mostrados em cinza. Todos os testes passaram usando o intervalo $0.0001 < p\text{-valor} < 0.9999$.

Testes DIEHARD	k_0	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9
<i>BirthdaySpacings</i>	100	100	100	100	100	100	100	100	100	100
<i>OverlappingPermutations</i>	99	97	98	95	98	96	98	98	99	100
<i>Ranks31x31 matrices</i>	100	100	100	100	100	100	100	100	100	100
<i>Ranks32x32 matrices</i>	100	100	100	100	100	100	100	100	100	100
<i>Ranks6x8 matrices</i>	0	0	25	99	100	100	100	100	100	100
<i>Monkey20bits Words</i>	0	99	100	100	100	100	100	100	100	100
<i>OPSO</i>	98	99	100	100	100	100	100	100	100	100
<i>OQSO</i>	98	100	100	100	100	100	100	100	100	100
<i>DNA</i>	100	100	100	100	100	100	100	100	100	100
<i>Count1sStream</i>	0	0	0	98	100	100	100	100	100	100
<i>Count1sSpecific</i>	0	0	0	0	94	100	100	100	100	100
<i>ParkingLot</i>	100	100	100	100	100	100	100	100	100	100
<i>MinimumDistance</i>	96	100	100	100	100	100	100	100	99	100
<i>RandomSpheres</i>	100	100	100	100	100	100	100	100	100	100
<i>Squeeze</i>	100	100	100	100	100	100	100	100	100	100
<i>OverlappingSums</i>	100	100	100	100	100	100	100	100	100	100
<i>Runs (up)</i>	100	100	100	100	100	100	100	100	100	100
<i>Runs (down)</i>	100	100	100	100	100	100	100	100	100	100
<i>Craps (wins)</i>	100	100	100	100	100	100	100	100	100	100
<i>Craps (throws/game)</i>	100	100	100	100	100	100	100	100	100	100

Fonte: Adaptada de MACHICAO; BRUNO (133)

k -mapa logísticos mostrados no diagrama de bifurcação (ver Figura 17) assim como na uniformização do histograma apresentada na Figura 21a.

4.4.3 Avaliação pela bateria de testes NIST

Assim como foi feito com a bateria de testes DIEHARD, o PRNG baseado no k -mapa logístico também foi avaliado seguindo as recomendações fornecidas pela bateria de testes NIST [Publicação Especial 800-22 Revisão 1a]. (171) Neste caso, a hipótese nula é aceita quando retorna um p -valor ≥ 0.01 , ou seja, a seqüência binária sendo testada é dita suficientemente aleatória.

Para este teste foram gerados 10 arquivos contendo 100MB, equivalente a 25×10^6 números gerados usando o PRNG k -mapa logístico para cada $k = \{0, 1, \dots, 9\}$. A partir deste conjunto de arquivos, 10 amostras contendo seqüências binárias de comprimento 10^6 bits foram divididas de cada arquivo, procedimento que é feito pela própria bateria de testes, sendo em total um conjunto de 100 amostras.

O número de amostras que passaram os testes do NIST para cada k -mapa logístico são relatados na Tabela 4. Desta Tabela, é possível observar que os PRNGs k_0 , k_1 e k_2 falham parcialmente os testes do NIST, mais especificamente, de cima para baixo nos testes, *Runs*, *LongestRun*, alguns sub-testes da *non-overlapping template*, *overlapping template*, *universal*, *approximate entropy* e *serial*. No entanto, a partir de $k \geq 3$, a maioria das 100 amostras são passadas satisfatoriamente. A diferença do DIEHARD, que consegue passar em todos os teste a partir de $k \geq 4$, razão pela qual consideramos apenas esta

última como o maior parâmetro que passa em todos os testes de aleatoriedade.

Tabela 4 – Resultados dos testes de aleatoriedade NIST para 100 arquivos gerados pelos diferentes k -mapa logístico com sementes aleatórias. Cada coluna indica o número de sequências (arquivos) que passaram. Os testes falhados são mostrados em cinza. Todos os testes passaram para o nível de significância $\alpha = 0.01$.

Testes NIST	k_0	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9
<i>Frequency</i>	98	99	99	99	99	99	99	99	99	99
<i>BlockFrequency</i> ($m = 128$)	0	1	66	95	98	98	99	100	99	99
CumulativeSums										
Forward sums	97	98	99	99	98	99	99	99	99	99
Reverse sums	97	99	99	99	99	99	99	99	99	99
<i>Runs</i>	0	0	14	91	98	99	99	99	99	100
<i>LongestRun</i>	0	0	15	89	98	99	98	100	99	99
<i>Rank</i>	99	100	99	99	99	99	99	99	99	99
<i>FFT</i>	77	98	99	99	99	99	99	99	99	99
<i>Non-overlappingTemplate</i>										
000000001	0	0	48	97	99	99	99	100	99	99
000000011	0	3	87	98	99	99	99	99	99	99
000000101	0	41	94	98	98	99	99	99	98	99
000000111	0	46	95	99	99	99	99	99	99	99
000001001	2	82	98	99	99	99	99	99	99	99
000001011	6	86	97	99	99	99	99	99	99	99
000001101	39	94	98	99	99	100	99	100	99	99
000001111	0	74	97	99	99	99	99	99	99	99
000010001	68	95	99	98	99	99	99	100	99	99
000010011	76	95	98	99	99	99	99	99	98	99
000010101	94	97	99	99	99	99	99	99	99	99
000010111	64	95	98	99	99	99	99	99	98	99
000011001	96	98	99	99	100	100	99	99	98	99
000011011	94	98	99	99	99	99	99	99	99	99
000011101	96	98	98	99	99	99	99	99	99	99
<i>OverlappingTemplate</i>	0	0	11	93	98	99	98	99	99	99
<i>Universal</i>	0	68	97	98	99	99	99	99	99	99
<i>ApproxEntropy</i> ($m = 10$)	0	0	64	98	99	99	99	100	99	99
RandomExcursions										
$x = -4$	90	98	99	99	98	99	99	99	99	100
$x = -3$	91	97	99	99	99	99	99	99	99	99
$x = -2$	94	99	98	99	99	98	99	99	99	99
$x = -1$	95	99	99	98	99	99	99	99	99	100
$x = 1$	95	99	99	99	99	99	99	98	99	100
$x = 2$	93	98	99	99	98	100	98	98	99	99
$x = 3$	89	97	98	99	99	99	98	98	99	99
$x = 4$	85	97	97	98	98	99	99	98	98	100
<i>RandExcurs Var</i>										
$x = -9$	99	100	99	99	100	99	99	99	99	100
$x = -8$	99	99	99	99	99	99	99	99	99	100
$x = -7$	100	99	99	99	99	99	99	99	99	100
$x = -6$	100	99	99	99	99	99	98	99	99	99
$x = -5$	100	99	99	99	99	99	99	99	99	99
$x = -4$	99	100	99	99	99	99	99	99	99	99
$x = -3$	99	100	99	98	99	99	99	99	99	99
$x = -2$	99	99	99	98	99	99	99	100	99	99
$x = -1$	99	99	99	99	99	99	99	99	99	99
$x = 1$	100	100	99	100	100	99	99	99	99	99
$x = 2$	99	99	99	100	99	99	99	99	99	100
$x = 3$	99	99	98	99	99	99	99	99	99	100
$x = 4$	99	99	99	100	99	99	99	99	99	99
$x = 5$	99	99	98	100	98	99	98	99	99	99
$x = 6$	99	98	98	99	98	99	99	99	99	99
$x = 7$	99	98	99	100	98	100	99	99	99	99
$x = 8$	99	99	100	99	99	99	99	100	99	99
$x = 9$	99	98	100	99	98	99	99	100	99	99
<i>Serial</i> ($m = 16$)										
<i>Serial 1</i>	0	1	82	96	98	99	99	98	99	99
<i>Serial 2</i>	10	81	95	98	98	99	99	99	98	99
<i>LinearComplexity</i> ($M = 500$)	99	98	99	99	99	99	99	98	99	99

Fonte: Adaptada de MACHICAO; BRUNO (133)

4.5 Comparação com PRNGs LCG e Mersenne Twister

Para comparar o desempenho das propriedades de pseudo-aleatoriedade do k -mapa logístico e os PRNGs clássicos, foram selecionados dois dos PRNGs mais populares: o gerador congruente linear (138, 139) e o Mersenne Twister. (163) como introduzido no Capítulo 3 O LCG foi implementado usando os parâmetros $a = 11$, $c = 25214903917$, $m = 2^{48}$, os quais são usados pela linguagem de programação Java (164). Para o PRNG Mersenne Twister foi usada a implementação em linguagem Java disponibilizada em. (194)

A comparação do número de arquivos que passaram ao teste DIEHARD para os PRNGs k_4 -mapa logístico, LCG e Mersenne Twister é resumido na Tabela 5. Como pode ser observado o LCG não consegue passar em todos os testes de aleatoriedade, pois apresentou dificuldade com os sub-testes *OPSO*, *OQSO* e *DNA*.

Tabela 5 – Comparação dos resultados de testes de aleatoriedade DIEHARD do k_4 -mapa logístico, gerador congruente linear (LCG) e Mersenne Twister (MT). Cada célula mostra o número de arquivos que passaram ao teste, dentre um total de 100 arquivos gerados com sementes aleatórias. Testes severamente falhados são mostrados em cinza. Todos os testes são passados no intervalo $0.0001 < p\text{-valor} < 0.9999$

Testes Diehard	k -mapa logístico ($k = 4$)	LCG	MT
<i>BirthdaySpacings</i>	100	100	100
<i>OverlappingPermutations</i>	98	99	94
<i>Ranks31x31 matrices</i>	100	100	100
<i>Ranks32x32 matrices</i>	100	100	100
<i>Ranks6x8 matrices</i>	100	100	100
<i>Monkey20bits Words</i>	100	100	100
<i>OPSO</i>	100	58	100
<i>OQSO</i>	100	0	100
<i>DNA</i>	100	0	100
<i>Count1sStream</i>	100	100	100
<i>Count1sSpecific</i>	94	100	100
<i>ParkingLot</i>	100	100	100
<i>MinimumDistance</i>	100	100	100
<i>RandomSpheres</i>	100	100	100
<i>Squeeze</i>	100	100	100
<i>OverlappingSums</i>	100	100	100
<i>Runs (up)</i>	100	100	100
<i>Runs (down)</i>	100	100	100
<i>Craps (wins)</i>	100	100	100
<i>Craps (throws/game)</i>	100	100	100

Fonte: Adaptada de MACHICAO; BRUNO (133)

Adicionalmente, na Figura 24 é observada a comparação do número médio de arquivos que passaram ao teste DIEHARD dos quatro geradores sendo avaliados: k -mapa logístico, k -mapa da tenda (Apêndice B) desde k_0 até k_{100} , o gerador congruencial linear e o Mersenne Twister. Além disso, as médias de todos os resultados do teste DIEHARD

foram comparadas usando o teste t de *Student* com um nível de confiança de 95%. Foi observado que para $k \geq 4$ e $k \geq 6$ do mapa logístico e do mapa da tenda, respectivamente, passam em todos os testes de pseudo-aleatoriedade. Além disso, no que diz respeito ao gráfico interno na Figura 24, ao comparar a abordagem proposta com o LCG e o Mersenne Twister, observou-se que a partir de $k \geq 3$ o mapa logístico obteve melhores propriedades pseudo-aleatórias do que o gerador congruente linear, enquanto que para $k \geq 3$ o mapa logístico tem se desempenhado tão bem quanto o gerador Mersenne Twister.

Figura 24 – Comparação do número médio de arquivos que passaram o teste DIEHARD entre os quatro PRNGs: o k -mapa logístico e k -mapa da tenda com k_0 a k_{100} , o gerador congruente linear (LCG) e o Mersenne Twister (MT). O eixo vertical indica o número médio enquanto o eixo horizontal os diferentes valores de k -dígitos. As curvas representam a média e desvio padrão (barra de erro sombreada). O gráfico interno mostra uma ampliação na região entre $k \in [0, 9]$.

Fonte: Adaptada de MACHICAO; BRUNO (133)

Desta análise comparativa, observamos que o PRNG baseado no k -mapa logístico pode ser comparado ao popular PRNG Mersenne Twister em termos da qualidade de pseudo-aleatoriedade fornecida pelos testes estatísticos DIEHARD e NIST, embora essas qualificações não sejam consideradas como um indicador final, mas sim como um detector de desvios da aleatoriedade.

4.5.1 Teste de densidade espectral

Na criptografia, um teste interessante para avaliar se uma imagem criptografada é suficientemente aleatória é usando o espectro de potência. De fato, alguns trabalhos mostraram sua utilidade para detectar possíveis periodicidades contidas em uma imagem criptografada. (49, 123) O espectro de potência é calculado a partir da transformada de Fourier da série original. No contexto de imagens, a transformada de Fourier n -dimensional, técnica bastante usada na visão computacional, é usada para procurar as frequências que compõem uma série que pode estar misturada com ruídos no domínio do tempo. Assim, intuitivamente, a densidade espectral auxilia na identificação de periodicidades. No

contexto de PRNGs, quando o gerador produz números considerados como ruído branco, com banda de frequência limitada, este é caracterizado por possuir picos constantes em uma banda finita de frequência.

Na Figura 25 é mostrada a análise espectral do gerador congruencial linear RANDU, introduzido no Capítulo 2, que foi quebrado por apresentar periodicidade quando observada em três dimensões, ou seja, x_n , x_{n+1} e x_{n+2} . Ao observar esta figura, fica evidente que esta possui 15 planos, os quais podem ser visualizados no espectro de potência, no qual aparecem 15 pontos brancos dispostos paralelamente. Por outro lado, na mesma figura também é ilustrada a análise espectral do gerador Mersenne Twister, nas mesmas condições, o qual não apresenta padrões visíveis, dado pelo pico máximo no centro.

Figura 25 – Espectro de potência da relação x_n , x_{n+1} e x_{n+2} do (a) gerador congruencial linear RANDU e do (b) PRNG Mersenne Twister, usando uma sequência de 150^3 números pseudo-aleatórios.

Fonte: Elaborada pela autora.

Na Figura 26, da esquerda para direita, pode ser observado o diagrama de Poincaré, que relaciona as sequências x_t^k , x_{t+1}^k e x_{t+2}^k , o espectro de potência visualizada em duas e três dimensões. De cima para baixo, é mostrado esta análise espectral para diferentes valores do parâmetro k . Para o mapa logístico original (k_0) é evidenciado os traços paralelos que indicam a presença de padrões, porém, conforme o valor k é aumentado, a densidade espectral vai aproximando-se de uma constante no centro, claro indicador da ausência de padrões.

Figura 26 – Espectros de potência calculado a partir da relação de x_t, x_{t+1} e x_{t+2} para 150^3 números gerados pelo PRNG k -mapa logístico, com k_0, k_1, k_2, k_3 e k_4 (de cima para baixo).

Fonte: Elaborada pela autora.

4.6 Padrão ouro PRNG $k_0, k_1, \dots, k_\infty$

Ao longo deste capítulo, foi observado que as propriedades de pseudo-aleatoriedade de um mapa caótico podem ser melhoradas à medida que k aumenta. De fato, por meio de todas as ferramentas de visualização (diagrama de bifurcação, diagrama de Poincaré, histograma de frequência), de análise pelo expoente de Lyapunov, dos testes de aleatoriedade, inclusive análise espectral; sugerem que o desempenho de qualidade do PRNG proposto se estabeleça em $k \geq 4$ e $k \geq 6$ para o mapa logístico e o mapa da tenda, respectivamente.

Mais detalhadamente, a análise do diagrama de bifurcação e do diagrama de Poincaré revelaram que à medida que k aumenta as qualidades pseudo-aleatórias do PRNG aumentam também. Além disso, foi observada uma intensificação do comportamento caótico, corroborado pelo expoente de Lyapunov, já que todos os parâmetros que levam à comportamento caótico tendem a atingir o limite máximo de Lyapunov, ou que por conseguinte indicam que o k -mapa logístico é sensível às condições iniciais, propriedade útil em termos de sistemas criptográficos.

Em relação à análise de distribuição de frequência, também foi observado uma rápida melhora da pseudo-aleatoriedade conforme k aumenta, uma vez que o padrão U obteve uma distribuição mais platô, o que, em princípio, indica que as órbitas visitam todas as regiões do espaço de fase (topologicamente transitiva). Em todas essas análises, descobrimos que as propriedades pseudo-aleatórias do k -mapa logístico podem ser visivelmente melhoradas quando k é aumentado. Na verdade, os padrões tornam-se cada vez mais difundidos até se tornar visualmente indistinguível ($k \geq 4$). Com relação aos testes de pseudo-aleatoriedade, também foi corroborado que as sequências geradas ao usar o mapa $k \geq 4$ -mapa logístico e $k \geq 6$ -mapa da tenda passaram com sucesso para os testes de aleatoriedade de DIEHARD e NIST.

Com tudo que foi exposto, somos conduzidos a formular a seguinte conjectura: conforme o parâmetro k aumenta, a pseudo-aleatoriedade é melhorada, indo entre o padrão regular (k_0) até o mais aleatório (k_∞). Obviamente, em termos computacionais, k_∞ seria impossível de provar, porém não pretende-se explorar esse lado da conjectura, e sim de se aproveitar que conforme k aumenta a pseudo-aleatoriedade é melhorada criando assim uma ferramenta útil: um **padrão ouro**. Com este padrão ouro, é possível gerar conjuntos de dados produzidos com diferentes parâmetros k que fornecem as classes distintas o que permite o estudo e desenvolvimento de métodos voltados ao reconhecimento de padrões e séries temporais não-lineares, como será visto no Capítulo 6.

5 PADRÕES ESPAÇO-TEMPO USANDO REDE-AUTÔMATO LLNA

Este capítulo é dividido em três partes. Na primeira parte é proposto um modelo de autômato celular cuja tesselação é dada pela topologia de redes e cujas regras são inspiradas na família de regras do “jogo da vida”, aqui denominado como rede-autômato *Life-Like* (LLNA, pelas siglas em inglês). (50)

Na segunda parte é proposto um método para o reconhecimento de padrões em redes baseado no LLNA. Por meio deste método é possível caracterizar a formação de padrões espaço-tempo, obtida pela própria dinâmica do autômato, por meio de medidas como entropia, comprimento da palavra e complexidade de Lempel-Ziv. Tais descritores tiveram sua performance testadas primeiro em bases de redes sintéticas (modelos teóricos bem conhecidos na literatura usadas aqui como *benchmark*).

Na terceira parte do capítulo, a rede-autômato LLNA proposta como método de reconhecimento de padrões em redes será validada usando um conjunto de redes sintéticas cujas classes são conhecidas previamente e desta classificação são observadas as taxas de acertos obtidas por métodos de aprendizado de máquinas conhecidos. Além disso, os parâmetros do método proposto e a robustez deste foram analisadas usando-se diferentes conjuntos de redes, que incluem redes de diferentes tamanhos e grau médio, redes livres de escala obtidas por meio de diferentes modelos teóricos e redes acrescidas de ruídos estruturais. Os resultados obtidos são então comparados com medidas estruturais de redes como o grau médio ($\langle k \rangle$), o grau hierárquico médio de nível 2 ($\langle H_{k_2} \rangle$), o grau hierárquico médio do nível 3 ($\langle H_{k_3} \rangle$), o coeficiente de aglomeração médio ($\langle cc \rangle$), o comprimento médio do caminho ($\langle l \rangle$) e o grau de assortatividade (ρ_P), além disso, também considerou-se a combinação destas [$\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \langle l \rangle, \rho_P$].

5.1 LLNA: Rede-autômato *Life-Like*

Nesta tese foi proposto um modelo de autômato celular em rede ou rede-autômato. Este modelo é descrito pela quintupla

$$\mathcal{R} = \langle \mathcal{T}, \mathbf{S}, s_0, \mathbf{N}, \phi \rangle,$$

no qual a tesselação $\mathcal{T}$ é representada pela topologia de uma rede $G(V, E)$. Cada vértice é representado por uma célula v_i que está conectada com seus $|\mathbf{N}(v_i)|$ vizinhos. Os estados das células foram restringidos a binários $S = \{0, 1\}$, assim $s(v_i, t) = 1$ representa uma célula viva, enquanto que $s(v_i, t) = 0$ representa uma célula morta para $s(v_i, t) \in S$ no tempo t , em que $s(v_i, 0)$ é a configuração inicial do autômato. Note-se que para um modelo mais generalizado o espaço de estados pode ser um número finito, porém neste trabalho está restringido a valores binários.

Quando trata-se com ACs sob uma tesselação de rede, onde o grau e a distribuição de grau variam consideravelmente, como, por exemplo, a rede aleatória ou livre de escala (ver Seção 2.5), as regras ou funções de transição dos ACs podem tambémm serem modeladas de maneira mais heterogênea que em ACs com topologia regular. Portanto, a função de transição $\phi : s(c_i, t) \rightarrow s(c_i, t + 1)$ pode ser implementada de diferentes maneiras. Uma primeira abordagem é usando a densidade da vizinhança $\sigma(v_i, t)$, que representa a porcentagem de vizinhos **vivos** de cada célula (52), assim como também é possível usar um limiar θ da forma $\sigma(v_i, t) > \theta$ proposto por Baetens e De Baets. (56) Nesta tese foi proposta uma forma mais generalizada da função de densidade, a qual pode ser tratada em intervalos como, por exemplo, $0.22 \leq \sigma(v_i, t) < 0.33$. A função da densidade de vizinhança $\sigma(v_i, t)$ do vértice v_i é a proporção de vizinhos com estado vivo, descrita pela Equação (5.1)

$$\sigma(v_i, t) = \sigma_i = \frac{1}{k_i} \sum_{j=1}^N A_{ij} s(v_j, t), \quad (5.1)$$

no qual A_{ij} é a matriz de adjacência da redes e k_i o grau definido na Equação (2.8) apresentada no Capítulo 2.

Nesta tese foi criada uma correspondência da função de transições com as regras da família do AC bidimensional *Life-Like*, descrito pela notação Bx-Sy (ver Seção 2.4). Já que trata-se de um AC outer-totalístico bidimensional é baseado na vizinhança de Moore, definido por uma célula central e seus oito vizinhos mais próximos, ou seja, em total existem $r = 9$ que é responsável por todas as combinações da família *Life-Like*, razão pelo qual existem $2^{9+9} = 262,144$ possíveis regras de transição. A função de transição do LLNA é definida como:

$$s(c_i, t + 1) = \begin{cases} 1, & \text{se } s(c_i, t) = 0 \text{ e } x/r \leq \sigma_i < (x + 1)/r \Rightarrow \text{nasce (B)} \\ 1, & \text{se } s(c_i, t) = 1 \text{ e } y/r \leq \sigma_i < (y + 1)/r \Rightarrow \text{sobrevive (S)} \\ 0, & \text{caso contrário.} \end{cases} \quad (5.2)$$

Na equação (5.2), as variáveis x e y representam as condições da regra *Life-Like* que estão indicadas na forma Bx₀x₁...x₈-Sy₀y₁...y₈, em que B e S representam “nascer ” e “sobreviver”, respectivamente; e, x_i e y_i variam de 0 a 8, como foi detalhado no Capítulo 2 (nomenclatura da família *Life-Like*).

Por exemplo, a regra B3-S23 é compreendida da seguinte forma: “se o estado da célula $s(c_i, t) = 0$ (morta) e a porcentagem de vizinhos vivos está entre $\frac{3}{9} \leq \sigma(v_i, t) < \frac{4}{9}$ então no tempo $t + 1$ o estado será $s(v_i, t + 1) = 1$ (nascerá). Por outro lado, se o estado é $s(v_i, t) = 1$ (viva) e a porcentagem de vizinhos vivos está entre $\frac{2}{9} \leq \sigma(v_i, t) < \frac{4}{9}$ então $s(v_i, t + 1) = 1$ (sobreviverá), caso contrário $s(v_i, t + 1) = 0$ (morrerá)”.

Este mesmo exemplo é ilustrado na Figura 27, na qual observa-se uma rede contendo $N = 21$ vértices. Na parte superior esquerda é apresentada a configuração inicial da rede, no qual a cor preta indica células com estado igual a zero, e na cor branca as células com estado igual a um. Para o exemplo, considerando a célula na região sombreada c_i , no tempo $t = 1$ possui estado $s(c_i, t) = 0$. Se cumprir com a condição estabelecida na Equação (5.2), logo será aplicado a regra de nascimento. Já que c_i possui 5 vizinhos, dos quais 2 estão vivos e 3 mortos, logo corresponde a uma densidade de vizinhança $\sigma(v_i, t) = \frac{2}{5} = 40\%$, e portanto, satisfaz a condição e o seguinte estado será $s(c_i, t + 1) = 1$ (nascer). Em seguida, todos os estados das outras células também são calculadas e sincronizadas. No tempo $t = 2$, dessa vez, o estado de célula é $s(c_i, t) = 1$, portanto é habilitada a regra de sobrevivência S23, porém, já que a vizinhança inteira está morta, logo a densidade de vizinhança é $\sigma(v_i, t) = \frac{0}{5} = 0\%$, pelo qual não satisfaz a condição e no seguinte estado esta célula morre.

Figura 27 – Construção da rede-autômato *Life-Like*. No lado esquerdo, de cima para baixo, é visualizada a evolução do autômato. Os estados são representados por branco (vivo) e preto (morto). No lado direito é detalhada a função de transição para a regra B3-S23.

Fonte: Elaborada pela autora.

5.1.1 Padrões na rede-autômato

A seguir serão analisados os diagramas espaço-tempo da rede-autômato *Life-Like*. Estes diagramas fornecem uma visão bidimensional dos padrões temporais formados pela evolução do autômato. Note que os diagramas espaço-tempo apresentados no Capítulo 2, tratam-se de tesselações uni e bidimensionais cuja vizinhança é espacial. No caso da

rede-autômato a topologia de rede é uma estrutura adimensional. Assim, por questões de visualização podem ser usadas diversas estratégias, por exemplo, simplesmente apresentando as células uma do lado da outra, da esquerda para a direita, sem nenhuma ordem em particular. (52) Porém pode ser útil o fato de observar os padrões em relação à alguma característica em específico, como por exemplo à tendência da distribuição de grau, à formação dos *hubs*, aos vértices centrais ou aos vértices da periferia, etc.

Nesta tese, optou-se por apresentar o diagrama espaço-tempo, da esquerda à direita, ordenado pelos graus dos vértices da correspondente rede. Dessa forma, mais à esquerda encontram-se os vértices com menor grau k_i e, mais à direita, encontram-se os vértices com grau k_i maior. Na Figura 28 é ilustrado o padrão espaço-tempo para uma rede livre de escala com $N = 500$ vértices e grau médio $\langle k \rangle = 8$ usando uma regra qualquer, como por exemplo a regra B1357-S2468. Nessa figura podem ser distinguidos três tipos de comportamentos (padrões). Na parte mais à esquerda, os vértices de menor grau tendem a morrer após poucas iterações (estável), já os vértices intermediários apresentam um padrão repetitivo (periódico), enquanto que os vértices de maior grau, mais à direita, apresentam um comportamento aparentemente aleatório (caótico).

Figura 28 – Diagrama espaço-tempo da rede-autômato *Life-Like* com regra B1357-S2468, para uma rede livre de escala de $N = 500$, $\langle k \rangle = 8$ evoluída por $t = 500$ iterações. As colunas representam os estados dos vértices vivos (branco) e mortos (preto) e as linhas representam uma iteração ou evolução no tempo t .

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

Na Tabela 6 são apresentados os diagramas espaço-tempo para quatro modelos de redes sintéticas com $N = 500$ com diferente grau médio $\langle k \rangle$ usando a regra B1357-S2468. A presença de diferentes padrões que variam com o grau médio é evidente, além disso, foi observado que estes padrões são invariantes em relação ao tamanho da rede (diagramas não mostrados). Entretanto, observou-se que para as redes com $\langle k \rangle > 8$ muitos destes padrões tendem ao comportamento caótico, isto pode ser devido ao grande número de vizinhos de cada célula que morrem e vivem sem previsão, evitando a propagação dos efeitos da função

de transição. Também foi observado a formação de diferentes padrões quando usadas outras regras, podendo ou não apresentarem esses comportamentos descritos anteriormente.

Tabela 6 – Diagrama espaço-tempo da rede-autômato *Life-Like* dos 4 modelos de redes sintéticas com $N = 500$ vértices e diferente grau $\langle k \rangle$, evoluídas por $t = 500$ iterações usando a regra B1357-S2468 com condições iniciais aleatórias. As colunas de cada diagrama representam os estados dos vértices vivos (branco) ou mortos (preto) e as linhas representam uma iteração ou evolução no tempo t .

Rede	$\langle k \rangle = 4$	$\langle k \rangle = 6$	$\langle k \rangle = 8$	$\langle k \rangle = 10$
Aleatória
Pequeno-mundo
Geográfica
Livre de escala

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

5.1.2 Medidas extraídas do padrão espaço-tempo

A dinâmica de uma rede-autômato fornece um padrão espaço-temporal global de evolução, como os exibidos na Tabela 6. Assim, cada vértice da rede pode ser analisado como uma sequência de uns e zeros e várias medidas podem ser usadas para extrair propriedades quantitativas de cada vértice. Um conjunto de medidas, como a entropia de Shannon, o comprimento da palavra e a complexidade de Lempel-Ziv.

- **Entropia de Shannon** (μ_S) de uma sequência binária é definida como

$$\mu_{S_i} = -(p_i^0 \log_2 p_i^0 + p_i^1 \log_2 p_i^1), \quad (5.3)$$

em que p_i^1 e p_i^0 é a probabilidade de ter uns e zeros para a célula c_i , respectivamente (195). A entropia de Shannon é um valor entre $[0, 1]$, assim valores altos de entropia indicam padrões oscilatórios e complexos, enquanto que valores mais baixos representam padrões mais regulares ou menos alternantes. Por

exemplo, a entropia das seguintes sequências “ $s_1 = (010101010101010101)$ ” e “ $s_2 = (01001101010111001001)$ ” é $\mu_S = 1$, respectivamente;

- **Comprimento da palavra** (μ_W) descreve a homogeneidade da sequência com respeito ao comprimento das “palavras” (aqui definidos como subsequência de uns limitados por zeros), por exemplo (0011101100) possui uma palavra de comprimento três e uma palavra de comprimento dois. Observe que o número de evoluções t da rede-autômato determina o maior comprimento de palavra possível, que é obtido quando uma célula (vértice) inicia a evolução viva e nunca muda seu estado.
- **Complexidade de Lempel-Ziv** (μ_L), é uma medida entrópica que, diferentemente da entropia de Shannon, a grosso modo, conta o número mínimo de subsequências distintas g que segmentem uma sequência binária por completo (135) Por exemplo, a decomposição da sequência s_1 em blocos mínimos é “0|1|01|010|10|101|0101”, portanto, o número de cadeias possíveis é $g = 7$ e cujo comprimento de sequência é $L = 20$, assim a complexidade Lempel-Ziv é calculada como:

$$\mu_{Li} = \frac{g \log L}{L}, \quad (5.4)$$

obtendo $\mu_L = 1.198$, da mesma forma s_2 é decomposto em “0|1|00|11|01|010|111|001” com $g = 8$, e a complexidade Lempel-Ziv para s_2 é $\mu_S = 1.348$. Semelhante à entropia, padrões estáveis tendem a diminuir as complexidades de Lempel-Ziv, enquanto padrões mais caóticos ou aleatórios tendem a valores mais altos. No entanto, algumas diferenças podem ser encontradas entre as duas medidas. Por exemplo, comparando as sequências s_2 e $s_3 = (01010101)$, em ambos os casos, as sequências possuem máxima entropia, entretanto a complexidade de Lempel-Ziv é de 1.30 e 1.35, respectivamente, uma vez que estes contêm 4 e 8 blocos mínimos.

5.1.3 Vetor de características

A construção de vetores de características é um passo fundamental na área de reconhecimento de padrões. Com esse propósito, um vetor de características pode ser obtida a partir da extração de medidas do padrão espaço-tempo da dinâmica do LLNA. Nesta tese consideram-se os vetores de distribuição da entropia de Shannon $\vec{\mu}_S$, distribuição do comprimento da palavra $\vec{\mu}_W$ e a distribuição de Lempel-Ziv $\vec{\mu}_L$, detalhadas a seguir.

O primeiro vetor de característica $\vec{\mu}_S$ consiste na distribuição da entropia de Shannon. A entropia de Shannon é obtida para cada vértice e, a partir desses valores, foi obtido um histograma dividindo o intervalo $[0, 1]$ em n compartimentos (*bins*). Portanto, $\vec{\mu}_S$ é composto por n descritores, que representam as respectivas frequências.

O segundo vetor de características $\vec{\mu}_W$ é composto pela distribuição de comprimento de palavra. Por cada vértice é calculada o número de frequências em que aparecem os

padrões $\{1, 11, 111, 1111 \dots\}$, sendo limitado por um comprimento de 40 caracteres, já que sequências maiores a esta são menos frequentes. Logo, estas frequências são normalizadas pelo número total de palavras, e logo é dividida em n compartimentos.

O terceiro vetor de características $\vec{\mu}_L$ contém a distribuição de complexidade Lempel-Ziv dividida em n compartimentos, este vetor foi normalizado pelo valor máximo alcançado entre o grupo de amostras.

Na Figura 29 são ilustradas as distribuições das três medidas para diferentes modelos de redes usando a regra B135678-S03456. As distribuições foram divididas em $n = 20$ compartimentos normalizadas pelo número de vértices N . Desta figura, é evidente a separação entre os quatro modelos de rede.

Figura 29 – Distribuição das medidas (a) entropia de Shannon $\vec{\mu}_S$, (b) comprimento da palavra $\vec{\mu}_W$ e (c) complexidade de Lempel-Ziv $\vec{\mu}_L$ extraídas da evolução de uma rede aleatória, pequeno-mundo, livre-escala e geográfica com $N = 1000$, $\langle k \rangle = 4$, evoluída $t = 350$ iterações usando a regra B135678-S03456.

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

5.1.4 Invariância ao tamanho da rede

Aqui é analisada a influência do número de vértices de rede (N) em relação do padrão espaço-temporal obtido do LLNA. A Tabela 7 apresenta os histogramas da entropia de Shannon, $\vec{\mu}_S$, usando a regra B01678-S0457 para os quatro modelos de redes: aleatória, pequeno-mundo, livre de escala e geográfica. Claramente, observou-se que o parâmetro N é independente de $\vec{\mu}_S$. Os histogramas para valores distintos de N são muito semelhantes, dado o mesmo modelo de rede.

Tabela 7 – Histogramas da distribuição de entropia de Shannon $\tilde{\mu}_S$ para cada um dos quatro modelos de rede (aleatório, pequeno mundo, livre de escala e geográfica) para diferentes valores do número de vértices $N = \{500, 1000, 1500, 2000\}$. Todas estas redes têm $\langle k \rangle = 8$ e foram iteradas durante $t = 350$ iterações usando a regra B01678/S0457.

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

Com base nos diferentes gráficos mostrados tanto na Figura 29 como na Tabela 7, foi observado que o modelo de rede-autômato proposto permite distinguir entre os vários modelos de redes sintéticas, o que nos conduziu a propor um método de reconhecimento de padrões usando o LLNA, que será descrito na seguinte seção.

5.2 Reconhecimento de padrões em redes usando o LLNA

A abordagem do LLNA foi desenvolvida para o reconhecimento de padrões em redes, sendo composta pelos seguintes passos: (i) seleção da regra, que busca pelas regras que atinjam as maiores taxas de acerto para classificar uma amostra de redes representativas para um problema específico; (ii) extração de características dos padrões espaço-temporais gerados pelo autômato-rede e finalmente; (iii) o reconhecimento de padrões por meio da classificação de redes, como mostra a Figura 30.

5.2.1 Dataset de redes sintéticas

Para mostrar a robustez do LLNA em problemas de reconhecimento de padrões foi feita uma análise do problema de classificação dos 4 modelos de redes sintéticas. Por isto, os experimentos foram conduzidos em três conjuntos de dados: *dataset-sintético*, *dataset-sintético-ruído* e *dataset-livre-escala*. Informações detalhadas sobre esses conjuntos

Figura 30 – Sequência de passos para o reconhecimento de padrões em redes usando a abordagem de rede-autômato LLNA.

Fonte: Elaborada pela autora.

de dados são descritas a seguir.

- *Dataset-sintético*: composto por 4 categorias de redes sintéticas geradas de acordo com os seguintes modelos: (a) aleatório (modelo Erdős-Renyi (63)), com probabilidade de conexão entre dois vértices de $p = \langle k \rangle / N$; (b) pequeno-mundo (modelo Watts-Strogatz (59)), com probabilidade de reconexão de $p = 0.1$; (c) livre de escala (modelo Barabási-Albert (65)), com conexão preferencial linear e (d) geográfica (modelo de rede geográfica (69)). Para cada modelo, foram consideradas redes com grau médio $\langle k \rangle = \{4, 6, 8, 10, 12, 14, 16\}$ e com número de vértices $N = \{500, 1000, 1500, 2000\}$. Foram geradas 100 redes cada uma com as 28 combinações de grau médio e número de vértices $\langle k \rangle$ em função de N . Portanto, o número total de redes neste conjunto de dados é 11200, e há 2800 de cada modelo;
- *Dataset-sintético-ruído*: Esta base de redes contém as mesmas redes da base *Dataset-sintético*, porém cada rede foi modificada pela remoção/adição aleatória de arestas de acordo com uma taxa de ruído ρ_N . Por exemplo, uma taxa de ruído de 20% ($\rho_N = 20\%$) indica a adição de 10% de arestas, em relação ao número total de arestas e, a remoção de 10% das arestas existentes. Logo, à medida que essa taxa aumenta, maiores serão as mudanças estruturais da topologia da rede;
- *Dataset-livre-escala*: Composto por redes livres de escala geradas de acordo com os modelos propostos por Dorogovtsev-Mendes (68) e por Barabási-Albert (65) com conexão preferencial linear e não linear $\alpha = \{0.5, 1.0, 1.5, 2.0\}$. Foram geradas 100 redes com cada uma das cinco variações de conexão preferencial. Portanto, este conjunto de dados possui cinco classes, contendo 100 redes para cada uma destas com $N = 1000$ vértices e grau médio $\langle k \rangle = 8$;

As redes aleatórias, pequeno-mundo e livres de escala do modelo Barabási & Albert foram geradas usando a biblioteca *igraph* (196), que é um pacote para análise de redes. As redes livre de escala do modelo Dorogovtsev & Mendes e as redes geográficas foram implementadas de acordo com seus correspondentes modelos propostos em (68, 69), respectivamente.

5.2.2 Seleção da regra

A seleção da regra consiste na busca por um conjunto de regras (soluções), que atinjam as maiores taxas de acerto quando apresentado um subconjunto de redes representativo para um problema específico. Este é um processo importante, pois uma mesma regra pode apresentar diferentes comportamentos para redes de diferentes classes, portanto, não é trivial prever qual seria a melhor regra para um problema específico.

É enfatizado que, como a seleção da regra é feita através de um procedimento de otimização para um problema específico (50), deve-se assumir que o conjunto de soluções (conjunto de regras selecionadas) também trazem algumas regras que não cumprem a expectativa quando apresentado novo conjunto de dados. Por este motivo, cada uma das 262,144 regras da família *Life-Like* deve ser avaliada. Isto é, por cada regra foi construída uma rede-autômato, logo extraídos padrões e obtida a taxa de acerto da classificação (porcentagem de instâncias classificadas corretamente) para finalmente obter um espaço de regras reduzido com as quais será feita a validação do método.

Como a escolha da melhor regra engloba a indução e avaliação de 262,144 classificadores, o custo computacional deve ser considerado, tanto no aspecto do método classificador quanto das configurações experimentais. Nesta fase, é recomendado a utilização do método kNN. Escolheu-se particularmente este método porque, em geral, gera melhores resultados, mantendo um excelente tempo de processamento. (96) Nota-se que a aplicação de outros métodos nesta fase, tais como redes neurais ou SVM, seria impraticável devido à complexidade do tempo associada a esses métodos. (96) Além disso, também devem ser usadas configurações da rede-autômato com o número de iterações razoáveis e número de amostras representativas do problema para permitir uma avaliação mais rápida.

Por outro lado, a escolha das melhores regras deve ser realizada em um conjunto de dados diferente, de tal forma a obter resultados imparciais. (96) Assim, preliminarmente, observe que um subconjunto de dados é usado para a seleção de regras, enquanto que um outro, exclusivo, é usado para avaliar o desempenho dos classificadores. Este procedimento garante que diferentes conjuntos de dados sejam empregados para a seleção da regra e para a validação dos classificadores, e que este último processo também seja imparcial ao usar o esquema de validação cruzada para as etapas de treinamento e teste.

Assim, apenas para a seleção de regras, foi utilizado um outro conjunto de redes (*dataset-seleção-regra*) contendo redes dos quatro modelos teóricos com os mesmos parâmetros de geração que o *dataset-sintético*. No entanto, este conjunto de dados contém apenas 50 redes com $N = 500$ vértices para cada uma das 7 combinações de grau médio $\langle k \rangle$. Note que as instâncias do *dataset-seleção-regra* são totalmente diferentes do *dataset sintético*, isto é para garantir uma verdadeira generalização do classificador. (96) A Tabela 8 apresenta um resumo das características gerais das bases utilizadas neste capítulo. Observe que bases

independentes foram usadas para a validação do método proposto, como para a seleção da regra.

Tabela 8 – Bases de dados de redes sintéticas.

	#Classes / Legendas	Validação		Seleção da regra	
		#Redes × classe		#Redes × classe	
Redes sintéticas					
Sintética	4 ER, WS, BA, GEO	11200	2800	1400	350
Sintética+ruído	4 ER, WS, BA, GEO	11200	2800	*	*
Livre-escala	5 BA $\alpha = \{0.5, 1.0, 1.5, 2.0\}$, 500 DM	500	100	*	*

Fonte: Elaborada pela autora.

A Figura 31 apresenta o histograma de frequências das taxas de acerto obtidas pelas 262,144 regras da família *Life-Like*, usando o conjunto de dados *dataset-seleção-regra*. Nesse experimento foi usada apenas a métrica da entropia de Shannon ($\vec{\mu}_S$) durante $t = 100$ iterações e o classificador k -NN. Observa-se que aproximadamente 11% do total de regras atingiram taxas de acerto acima de 90%. A partir deste conjunto, selecionou-se 10 regras que forneceram as maiores taxas de acerto para então serem usadas nos subseqüentes experimentos deste capítulo.

Figura 31 – Distribuição das 262,144 regras da família *Life-Like* de acordo à taxa de acerto atingida para a classificação das redes sintéticas (aleatória, pequeno-mundo, livre-escala e geográfica) usando o vetor de características composto pela entropia de Shannon ($\vec{\mu}_S$) durante $t = 100$ iterações e o classificador k -NN.

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

5.2.3 Sensibilidade dos parâmetros do LLNA

Note-se que os parâmetros de tempo t e a escolha da regra são termos empíricos. Por este motivo, foram avaliadas as taxas de acerto para esses dois parâmetros: tempo t e configuração das condições iniciais dos estados σ , mostrados na Figura 32.

Na Figura 32a pode-se observar a taxa de acerto com respeito ao tempo de evolução da rede-autômato $t = \{100, 150, \dots, 450\}$. Aqui é observado que a taxa de acerto tende a aumentar à medida que os valores de t também aumentam. A taxa de acerto inicial já é alta, dado que as três regras ilustrativas estão entre as dez regras selecionadas anteriormente. Existe também uma convergência rápida das taxas de acerto, que foi observada para as três regras analisadas, embora este comportamento não possa ser assumido para todas as regras. No entanto, para padrões que não convergem, um aumento no número de iterações tempo pode fornecer mais detalhes sobre a evolução da topologia.

Por outro lado, na Figura 32b é observada a taxa de acerto com respeito ao número de células vivas na configuração inicial do autômato. Foi realizado o mesmo experimento de classificação de rede, dessa vez considerando diferentes valores de $\sigma = \{10\%, 20\%, \dots, 90\%\}$. Neste experimento é possível observar que os valores de σ mais próximos de uma distribuição uniforme de estados, ou seja, $\sigma = 50\%$, fornecem taxas de acertos mais altas.

Figura 32 – (a) Taxa de acerto médio (%) e desvio padrão (barra de erro) em relação ao tempo de iterações t para as três regras destacadas (B01678-S0457, B035678-S023456 e B1245678-S3456). (b) Acerto médio (%) e desvio padrão (barra de erro) para diferentes probabilidades de condições iniciais de células vivas (σ) usando a regra B135678-S03456 e classificador SVM.

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50).

Com base no comportamento observado dos parâmetros t e σ , foi adotado os seguintes valores $t = 350$ e $\sigma = 50\%$ nos experimentos subsequentes.

5.2.4 Classificação dos quatro modelos de redes sintéticas

A Tabela 9 apresenta as taxas de acerto médio e desvio padrão para a classificação dos quatro modelos de redes sintéticas (ER, SW, SF e GEO). Neste experimento foram avaliadas as 10 melhores regras da família *Life-Like*, escolhidas durante a seleção da regra. Para cada regra de transição, as 11,200 redes do *dataset-sintético* foram evoluídas durante $t = 350$ iterações. A classificação foi feita utilizando o esquema de 10-validação cruzada junto ao algoritmo de classificação supervisionado SVM.

Por outro lado, o desempenho da classificação foi avaliado usando as as distribuições da entropia de Shannon $\vec{\mu}_S$, o comprimento da palavra $\vec{\mu}_W$ e a complexidade de Lempel-

Ziv $\vec{\mu}_L$. A distribuição dessas medições foram usadas como vetores de características, respectivamente, bem como a combinação deles $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$. Adicionalmente, os valores médios dessas medições como vetores de características $[\langle\mu_S\rangle, \langle\mu_W\rangle, \langle\mu_L\rangle]$ também foi considerada.

No lado esquerdo da Tabela 9, as colunas $\vec{\mu}_S$, $\vec{\mu}_W$ e $\vec{\mu}_L$ mostram as taxas de acerto para cada regra, respectivamente. A maior taxa de acerto foi obtida pela regra B35678-S03456 com 99.9 (± 0.004)% foi obtida de $\vec{\mu}_L$. No lado direito da Tabela 9 são mostradas as taxas de acerto por cada regra usando o vetor de características composto pela combinação das três distribuições $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$, assim como da combinação dos valores médios das mesmas medidas como descritores: $[\langle\mu_S\rangle, \langle\mu_W\rangle, \langle\mu_L\rangle]$. A melhor taxa de acerto foi obtida pela regra B135678-S03456 com 99.992 (± 0.002)% que supera a taxa de acerto mesmo quando usadas separadamente. Na última linha da Tabela é mostrada o número de descritores (ND) que contém cada vetor de características.

Além disso, é possível observar, que em geral obtiveram-se altas taxas de acerto para a maioria dos vetores de características, e relativamente menores para o vetor das médias de cada medida $[\langle\mu_S\rangle, \langle\mu_W\rangle, \langle\mu_L\rangle]$. Já quando analisados separadamente, as distribuições também forneceram altas taxas de acertos, especialmente a distribuição da entropia de Shannon e de Lempel-Ziv.

Tabela 9 – Taxa de acerto (%) obtida usando diferentes medidas e suas combinações como descritores para classificar os modelos de rede: aleatório, pequeno-mundo, livre de escala e geográfica. As redes foram construídas com $N = 500, 1000, \dots, 2000$ e grau médio variando de $\langle k \rangle = 4, 6 \dots 16$. A última linha apresenta o número de descritores (ND) de cada vetor de características.

Regras	$\vec{\mu}_S$	$\vec{\mu}_W$	$\vec{\mu}_L$	$[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$	$[\langle\mu_S\rangle, \langle\mu_W\rangle, \langle\mu_L\rangle]$
B01678-S0457	99.37 (± 0.01)	98.87 (± 0.01)	99.55 (± 0.01)	99.946 (± 0.003)	97.34 (± 0.02)
B035678-S023456	98.91 (± 0.02)	99.33 (± 0.02)	98.96 (± 0.02)	99.89 (± 0.01)	84.31 (± 0.03)
B1245678-S3456	97.40 (± 0.03)	96.09 (± 0.01)	97.67 (± 0.02)	99.80 (± 0.01)	95.65 (± 0.01)
B0157-S457	97.19 (± 0.04)	87.74 (± 0.03)	94.60 (± 0.03)	98.67 (± 0.04)	82.60 (± 0.03)
B35678-S03456	98.77 (± 0.02)	99.26 (± 0.03)	99.900 (± 0.004)	99.98 (± 0.01)	68.61 (± 0.05)
B1245678-S03456	96.24 (± 0.04)	96.67 (± 0.03)	95.57 (± 0.02)	99.91 (± 0.01)	95.65 (± 0.01)
B35678-S13456	97.70 (± 0.05)	99.03 (± 0.03)	99.92 (± 0.01)	99.94 (± 0.01)	70.28 (± 0.08)
B01245678-S3456	97.73 (± 0.03)	97.93 (± 0.02)	91.63 (± 0.04)	99.68 (± 0.01)	94.57 (± 0.05)
B035678-S0123456	98.39 (± 0.02)	98.93 (± 0.02)	98.34 (± 0.03)	99.79 (± 0.01)	84.67 (± 0.05)
B135678-S03456	99.47 (± 0.02)	99.912 (± 0.003)	99.76 (± 0.01)	99.992 (± 0.002)	79.30 (± 0.06)
ND	20	20	20	60	3

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

Para ilustrar a diferenciação entre as classes obtida com o método proposto, na Figura 33a é ilustrada a projeção da análise canônica em duas dimensões usando a regra B135678-S03456 e o vetor de características $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$. Nesta imagem, observa-se que existe uma clara separação entre os quatro modelos.

5.2.5 Tolerância a ruído

A robustez do método LLNA foi avaliada quanto à tolerância ao ruído. A topologia da rede foi modificada pela remoção e adição de arestas de acordo com uma taxa de

Figura 33 – Análise das variáveis canônicas (CVA) do vetor de características $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$, usando a regra B135678-S03456, representando as 11,200 redes correspondentes aos modelos: aleatório, pequeno-mundo, livre de escala e geográfica.

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50).

ruído ρ_N . À medida que essa taxa aumenta, mais mudanças estruturais são realizadas na topologia da rede. Uma taxa de ruído de 20 % ($\rho_N = 20\%$) indica a adição de 10% de arestas, em relação ao número total de arestas e, a remoção de 10 % das arestas existentes. Ambos os conjuntos de arestas são selecionados aleatoriamente. Na Tabela 10 é apresentada a taxa de acerto para a classificação dos quatro modelos de rede (ER, SW, SF e GEO) ao considerar diferentes valores de ruído ρ_N e usando o classificador SVM. Em todos os casos, foi usado o vetor de características composto pela combinação das distribuições, $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$. Em geral, foi observado que o aumento de ruído ρ_N não influencia na performance obtida para cada regra, uma vez que as mudanças das taxas de acerto foram relativamente pequenas.

Tabela 10 – Análise da robustez do LLNA. Taxa de acerto (%) com valores diferentes de ruído ρ_N usando $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$ como vetor de características e classificador SVM.

Regras	$\rho_N = 0\%$	$\rho_N = 10\%$	$\rho_N = 20\%$	$\rho_N = 30\%$
B01678-S0457	99.946 (± 0.003)	99.85 (± 0.01)	99.70 (± 0.01)	99.84 (± 0.01)
B035678-S023456	99.89 (± 0.01)	99.90 (± 0.01)	99.79 (± 0.01)	99.95 (± 0.01)
B1245678-S3456	99.80 (± 0.01)	99.79 (± 0.02)	99.51 (± 0.02)	99.73 (± 0.01)
B0157-S457	98.67 (± 0.04)	99.44 (± 0.02)	99.73 (± 0.02)	99.85 (± 0.01)
B35678-S03456	99.98 (± 0.01)	99.96 (± 0.01)	99.91 (± 0.01)	99.990 (± 0.003)
B1245678-S03456	99.91 (± 0.01)	99.80 (± 0.01)	99.61 (± 0.02)	99.68 (± 0.02)
B35678-S13456	99.94 (± 0.01)	99.94 (± 0.01)	99.93 (± 0.01)	99.97 (± 0.01)
B01245678-S3456	99.68 (± 0.01)	99.54 (± 0.01)	99.48 (± 0.02)	99.62 (± 0.02)
B035678-S0123456	99.79 (± 0.01)	99.86 (± 0.01)	99.84 (± 0.01)	99.93 (± 0.01)
B135678-S03456	99.992 (± 0.002)	99.983 (± 0.004)	99.97 (± 0.01)	99.95 (± 0.01)

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

5.2.6 Classificação de modelos de redes livre de escala

Da mesma forma que nos experimentos anteriores, aqui foram usadas as mesmas regras encontradas durante a seleção da regra, assim como da configuração dos experimentos usando o classificador SVM. A Tabela 11 apresenta a taxa de acerto obtida na classificação

de diferentes modelos de redes livre de escala, ou seja, redes com conexão preferencial linear e não linear ($\alpha = \{0.5, 1.0, 1.5, 2.0\}$) usando o modelo de Barabási & Albert (65), e, redes geradas de acordo com o modelo de Dorogovtsev & Mendes (68) (*dataset-livre-escala*).

O vetor de características formado pela combinação das distribuições também forneceu as taxas de acerto mais altas para este conjunto de dados, cuja maior taxa foi de 98.3 (± 0.2)% usando a regra B0157-S457. O desempenho da entropia Shannon e as distribuições da complexidade Lempel-Ziv também podem ser realçados. No entanto, existe uma heterogeneidade em relação ao desempenho dos vetores de características para cada regra, por exemplo, o vetor $[\langle\mu_S\rangle, \langle\mu_W\rangle, \langle\mu_L\rangle]$ teve uma boa performance para a regra B035678-S0123456 (89.5 *pm*0, 2 %). Em contraste, o mesmo vetor de características forneceu uma taxa de acerto de 70 (± 1)% para a regra B01678-S0457. Os resultados obtidos indicam que mesmo as redes com topologias similares podem fornecer um padrão espaço-temporal específico, que pode ser usada para caracterizar redes em um contexto de reconhecimento de padrões.

Tabela 11 – Taxas de acerto (%) obtida para a classificação dos 5 tipos de redes livre de escala. A última linha apresenta o número de descritores de cada vetor de características.

Regras	$\vec{\mu}_S$	$\vec{\mu}_W$	$\vec{\mu}_L$	$[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$	$[\langle\mu_S\rangle, \langle\mu_W\rangle, \langle\mu_L\rangle]$
B01678-S0457	85.4 (± 0.2)	83.4 (± 0.3)	88.8 (± 0.4)	94.9 (± 0.2)	70 (± 1)
B035678-S023456	82.77 (± 0.08)	75.8 (± 0.5)	83.7 (± 0.4)	85.8 (± 0.5)	79.200 (± 0.001)
B1245678-S3456	81.5 (± 0.4)	79.5 (± 0.1)	86.9 (± 0.2)	87.0 (± 0.2)	79.800 (± 0.001)
B0157-S457	96.5 (± 0.2)	67.6 (± 0.4)	95.3 (± 0.2)	98.3 (± 0.2)	79.8 (± 0.6)
B35678-S03456	89.0 (± 0.2)	86.2 (± 0.2)	81.0 (± 0.8)	88.4 (± 0.3)	83.8 (± 0.2)
B1245678-S03456	82.4 (± 0.3)	81.0 (± 0.3)	87.61 (± 0.08)	87.5 (± 0.1)	79.800 (± 0.001)
B35678-S13456	84.2 (± 0.2)	91.81 (± 0.03)	81 (± 1)	92.6 (± 0.1)	86.5 (± 0.3)
B01245678-S3456	79.9 (± 0.2)	79.7 (± 0.2)	85.7 (± 0.1)	85.5 (± 0.1)	79.80 (± 0.02)
B035678-S0123456	83.6 (± 0.1)	84.0 (± 0.3)	84.1 (± 0.5)	94.0 (± 0.2)	89.5 (± 0.2)
B135678-S03456	90.5 (± 0.2)	87.8 (± 0.2)	92.0 (± 0.2)	92.9 (± 0.2)	88.82 (± 0.06)
ND	20	20	20	60	3

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

Na Figura 34 é apresentada a projeção bidimensional obtida pela análise de variáveis canônicas para o vetor características $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$ com a regra B0157-S457 que obteve a maior taxa de acerto. Desta figura fica evidente que existe uma separação clara entre os três modelos de rede que corroboram as altas taxas de acertos e uma interseção entre os modelos livre de escala com $\alpha = 1.5$ e $\alpha = 2.0$.

5.2.7 Comparação com medidas estruturais de redes

A performance do método proposto foi comparada com as taxas de acerto das medidas estruturais de redes. Observe-se que as medidas cobrem diversas propriedades das redes como são as medidas relacionadas à conectividade, à distância e à correlação de grau, como introduzidas no Capítulo 2. Dentre as várias medidas estruturais foram escolhidas as seguintes medidas para compor o vetor de características: grau médio ($\langle k \rangle$), grau hierárquico de nível 1 médio ($\langle H_{k_1} \rangle$), grau hierárquico de nível 2 médio ($\langle H_{k_2} \rangle$),

Figura 34 – Análise das variáveis canônicas (CVA) representando as 500 redes livre de escala dos modelos Barabási-Albert para $\alpha = \{0.5, 1.0, 1.5, 2\}$ e Dorogovtsev-Mendes, usando a regra B0157-S457 e $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$ como vetor de características.

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

coeficiente de aglomeração médio ($\langle C \rangle$), menor caminho médio ($\langle l \rangle$) e assortatividade de grau (Γ). Sendo em total 6 descritores, que foram avaliados tanto isoladamente como agrupadas.

Na tabela 12a apresenta a comparação entre as medidas estruturais e as melhores taxas de acerto obtidas pelo LLNA. No que tange à classificação das 4 classes de redes sintéticas, ambas abordagens forneceram resultados semelhantes (100% de acerto considerando o desvio padrão), isto quando considerado o vetor de características combinado $[\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \langle l \rangle, \rho_P]$, pois de forma isolada o LLNA é bastante diferenciado. Isso pode ser explicado pelo fato de que as redes usadas neste experimento foram geradas a partir de modelos teóricos clássicos que apresentam propriedades conhecidas que podem ser caracterizadas por várias medidas. Porém, comparando as medidas estruturais de forma isolada, as taxas de acerto do LLNA é bastante diferenciado.

Com respeito à robustez ao ruído, as medidas tradicionais também são comparadas com o LLNA, pode ser observado similares resultados aos obtidos com o *dataset-sintético*, ou seja, que ambas abordagens são robustas ao ruído. Por fim, na última coluna da Tabela 12 é apresentada a comparação com as medidas estruturais para a classificação do *dataset-livre-escala*. Neste caso, o LLNA ultrapassa a taxa de acerto obtida com a combinação de medidas estruturais ($96.20 \pm 0.04\%$), oferecendo uma melhoria de $2.1 \pm 0.06\%$.

5.3 Considerações finais

Neste capítulo foi proposto o modelo de rede-autômato *Life-Like* (LLNA). Por meio do diagrama de espaço-tempo foram analisados os padrões das diferentes topologias de redes. Foi proposto também um método de reconhecimento de padrões, o qual segue os

Tabela 12 – Comparação da taxa de acerto (%) para a classificação de três conjuntos de dados (*dataset-sintético*, *dataset-sintético-ruído* e *dataset-livre-escala*) usando a concatenação de medidas estruturais de rede [$\langle k \rangle$, $\langle H_{k_2} \rangle$, $\langle H_{k_3} \rangle$, $\langle cc \rangle$, $\langle l \rangle$, ρ_P] e os resultados obtidos com o LLNA usando a combinação de melhor regra e vetor de características.

Descritor	<i>dataset-sintético</i>	<i>dataset-sintético-ruído</i>				<i>dataset-livre-escala</i>
	$\rho_N = 0\%$	$\rho_N = 10\%$	$\rho_N = 20\%$	$\rho_N = 30\%$		
$\langle k \rangle$	26.2 (± 0.4)	26.0 (± 0.5)	26.1 (± 0.4)	26.2 (± 0.4)	—	
$\langle H_{k_2} \rangle$	43.4 (± 0.4)	43.5 (± 0.5)	43.5 (± 0.5)	44.2 (± 0.5)	90.6 (± 0.1)	
$\langle H_{k_3} \rangle$	44.3 (± 0.2)	44.0 (± 0.3)	43.5 (± 0.4)	43.1 (± 0.4)	86.7 (± 0.2)	
$\langle cc \rangle$	76.11 (± 0.02)	75.64 (± 0.02)	75.92 (± 0.02)	75.79 (± 0.02)	76.3 (± 0.2)	
$\langle l \rangle$	50.46 (± 0.05)	43.82 (± 0.04)	42.7 (± 0.2)	41.6 (± 0.3)	69.6 (± 0.2)	
ρ_P	75.2 (± 0.2)	83.72 (± 0.03)	85.87 (± 0.03)	85.90 (± 0.03)	88.4 (± 0.3)	
agrupado	100.0 (± 0.0)	100.0 (± 0.0)	100.0 (± 0.0)	100.0 (± 0.0)	96.20 (± 0.04)	
LLNA	99.992 (± 0.002) (B135678-S03456)	99.983 (± 0.004)	99.97 (± 0.01)	99.95 (± 0.01)	98.3 (± 0.2) (B0157-S457)	
			B135678-S03456			

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

passos (i) seleção da regra, (ii) evolução da rede-autômato, (iii) extração de características e (iv) classificação. Além disso, foi proposto o uso de medidas da teoria da informação e teoria da complexidade como é a entropia de Shannon, comprimento da palavra e complexidade de Lempel-Ziv. Os resultados preliminares mostraram que a dinâmica da evolução da rede-autômato pode ser usada como uma ferramenta eficaz para o reconhecimento de padrões. Em seguida, para demonstrar a aplicabilidade geral do método serão discutidos no Capítulo 6 problemas de reconhecimento de padrões em problemas reais.

6 RECONHECIMENTO PADRÕES E ALEATORIEDADE EM REDES USANDO O LLNA

Este capítulo visa cobrir dois propósitos principais que estão inter-relacionados. O primeiro propósito é demonstrar o potencial de aplicabilidade geral do *Life-Like network automata*, para o qual são apresentados diferentes cenários que podem ser modelados como redes e que precisem de métodos de reconhecimento de padrões. Enquanto o segundo propósito, é de abordar o problema de classificação da qualidade pseudo-aleatória das sequências geradas pelo PRNG baseado no k -mapa logístico (padrão ouro) usando o LLNA. Além disso, percebe-se que também foi proposta uma solução alternativa a este problema que consegue superar os resultados obtidos com o LLNA.

Assim, na primeira parte, as análises realizadas das redes decorrem de problemas práticos, nos quais a identificação automática continua a ser um problema importante para cada contexto específico. Desta forma, as redes a serem utilizadas são de diferentes entornos como, por exemplo: as relações sociais, as relações metabólicas, as relações de adjacência dos estômatos de plantas, as relações textuais, as relações de adjacência em séries temporais, entre outros. Tendo em vista a importância dessas aplicações, muitos destes problemas podem ser estudados no contexto de redes complexas. Para isto, no que tange à modelagem de redes, foram propostas soluções de forma individual a cada aplicação.

Cinco problemas específicos são abordados. Para fins de organização, o capítulo é dividido em seções para cada diferente aplicação da metodologia proposta. (i) Uma primeira aplicação (Seção 6.2) é a identificação de padrões estruturais em redes sociais *online*, no qual as redes de usuários de Twitter e Google+ são analisadas; (ii) a segunda aplicação (Seção 6.3) é usada na identificação de organismos de diferentes domínios da vida usando suas reações metabólicas representadas como redes; (iii) na terceira aplicação (Seção 6.4), no contexto de análise de plasticidade fenotípica de plantas, é realizada a classificação de padrões de distribuição em imagens microscópicas de estômatos, cujas espécies foram submetidas a diferentes condições de estresse (iluminação, umidade e CO_2); (iv) a quarta aplicação (Seção 6.5), que diz respeito a identificação de autoria de manuscritos literários, engloba três bases de redes dependendo do tratamento de lematização dos textos usados para gerar as redes. A primeira consiste em textos com lematização parcial, completa e nenhuma; (v) a quinta aplicação (Seção 6.6), aborda uma metodologia para o reconhecimento de padrões de sequências de números pseudo-aleatórios gerados pelo padrão ouro k -mapa logístico descrito no Capítulo 4.

Para maior facilidade de leitura, cada seção deste capítulo apresenta um problema e contexto independentemente das outras. Cada seção segue a mesma metodologia que foi

introduzida no Capítulo 5, ou seja, primeiro uma contextualização do problema, em seguida a configuração do experimento específico (seleção da regra, extração de características e classificação), os respectivos resultados apresentados em forma de tabelas contendo as taxas de acertos e matriz de confusão, visualização dos descritores e sua projeção em duas dimensões, comparação com resultados advindos das medidas estruturais de redes e finalmente discussões dos resultados e/ou observações. Na última parte deste capítulo (Seção 6.8), é feita uma síntese dos principais resultados, das relações encontradas no que tange à seleção das regras, e da comparação entre os resultados obtidos usando o LLNA e as medidas estruturais de redes.

6.1 Configurações dos experimentos

6.1.1 *Benchmark* de redes do mundo real

Para avaliar o método proposto, diferentes modelos de redes que representam problemas do mundo real foram coletadas e/ou geradas dependendo da aplicação em específico. A seguir é dado um resumo das informações sobre esses conjuntos de dados, cuja contextualização será melhor detalhada nas correspondentes seções.

- ***Dataset-social***: este conjunto de dados contém redes da plataforma SNAP (sigla em Inglês de *Stanford Network Analysis Project*). (197) Foram selecionadas aleatoriamente 65 amostras de redes tanto para o Google+ como para o Twitter. Todas as redes sociais representam as relações sociais ou de amizade de usuários anônimos;
- ***Dataset-metabólicas***: este conjunto compreende a base de redes metabólicas. Esta base também foi estudada por Jeong et al. (23), e disponível no repositório (83), a qual contém 43 amostras que fornecem uma descrição das vias metabólicas de três classes de organismos: 6 *archaea*, 32 *bacteria* e 5 *eukaryotes*;
- ***Dataset-estômatos***: este conjunto de dados consiste em três bases de redes correspondentes às espécies: *Tradescantia zebrina*, *Ctenanthe oppenheimiana* e *Callisia repens* que foram submetidas a diferentes condições ambientais: injeção de CO₂ e exposição de luz (natural, 4 horas e 24 horas). Cada rede foi modelada a partir das imagens binárias que representam a distribuição dos estômatos nas folhas, e em seguida foi adotada a estratégia proposta por Backes et al. (17), a qual considera cada ponto (estômato) como um vértice do grafo. O critério para estabelecer as arestas são baseados na distância euclidiana entre pares de estômatos, de acordo a um raio de limiar (δ_{T_l});
- ***Dataset-textuais***: este conjunto de dados corresponde à modelagem de textos literários de autores conhecidos como proposta ao problema de atribuição de autoria. Foi usado um *corpus* que compreende 100 livros em língua inglesa, que foram obtidos

no repositório Project Gutenberg. (198) Foram considerados cinco obras literárias por cada um dos 20 autores distintos, a seguir: *Doyle, Stoker, Darwin, Dickens, Hardy, Wodehouse, Poe, Munro, Melville, Grey, Lang, Davis, James, Bower, Irving, Wells, Alger, Austen, Twain* e *Hawthorne*. A lista completa de livros e os respectivos autores são fornecidos no Apêndice C.

- **Dataset-PRNGs:** este conjunto de dados contém redes espaço-tempo construídas a partir de sequências de números pseudo-aleatórios gerados pelo PRNG baseado no k -mapa logístico proposto no Capítulo 4. Foi usado redes espaço-tempo limiarizadas.

6.1.2 Estratégias para a seleção da regra

Seguindo a metodologia apresentada no Capítulo 5. Dado que o LLNA baseia-se na seleção das melhores regras que otimizem um determinado problema em específico (50), logo cada uma das 262,144 regras da família *Life-Like* devem ser avaliadas usando os conjuntos de dados exclusivamente destinados a este processo. Observe que cada aplicação teve particulares configurações para os vetores de características para a seleção da regra. O LLNA foi evoluído durante $t = 350$ iterações e condições iniciais aleatórias com probabilidade $\sigma = 50\%$. Na maioria dos casos, foi usado apenas o descritor da distribuição da entropia de Shannon $\vec{\mu}_S$, a qual foi dividida em 20 *bins*, ou seja, 20 descritores, e o classificador kNN com $k = 1$. Escolhemos particularmente este classificador porque, em geral, gera melhores resultados, mantendo um bom tempo de processamento. (96)

Além disso, todos os conjuntos de dados utilizados foram divididos em dois subconjuntos mutuamente exclusivos: um para a seleção de regras e um para a validação do método. Esta divisão foi feita com o intuito de gerar um classificador imparcial. (96) A Tabela 13 apresenta um resumo contendo todas as informações sobre as bases de redes usadas neste Capítulo. A primeira coluna, indica o nome da base e os objetos de estudo. A segunda coluna indicada o número de classes e respectivas legendas. A terceira coluna indica o número de redes usadas para a validação, incluindo o número de amostras por classe. A última coluna indica o número de instâncias usadas especificamente para a seleção das regras.

6.1.3 Estratégias para a extração de características

Para a fase de extração de características, aqui foram usadas vários descritores para compor o vetor de características, de forma isolada ou combinada. Para o LLNA, foram utilizadas as distribuições da entropia de Shannon $\vec{\mu}_S$, o comprimento da palavra $\vec{\mu}_W$ e a complexidade de Lempel-Ziv $\vec{\mu}_L$. A distribuição dessas medições foram usadas como vetores de características, respectivamente, bem como a combinação deles $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$. Adicionalmente, os valores médios dessas medições como vetores de características $[\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle]$ também foram considerados. No que tange ao número de descritores (ND), isto depende

Tabela 13 – Resumo de informações das bases de dados de redes apresentadas neste capítulo.

	#Classes / Legendas	Validação		Seleção da regra	
		#Redes × classe		#Redes × classe	
Redes sociais SNAP (197)	2 <i>Twitter, Google+</i>	100	50	30	15
Redes metabólicas WIT (83)	3 <i>Archaea, Bacterium, Eu-</i> <i>karyote</i>	37	4, 30 e 3	6	2, 2 e 2
Redes de estômato <i>Tradescantia zebrina</i>	3 natural, L4h, L24h	12	4	6	2
<i>Callisia repens</i>	2 natural, CO ₂ (↑), CO ₂ (↓)	15	5	—	—
<i>Ctenanthe oppenheimiana</i>	2 L4h, L24h	36	18	—	—
Redes textuais <i>Project Gutenberg</i> (198)	8 <i>Doyle, Stoker, Darwin, Dic-</i> <i>kens, Hardy, Wodehouse, Poe,</i> <i>Munro</i>	40	5	60	5
Redes PRNGs <i>k</i> -mapa logístico	5 <i>k₀, k₁, k₂, k₃, k₄</i>	500	100	150	50

Fonte: Elaborada pela autora.

do número de compartimentos em que as distribuições são divididas. Adotamos arbitrariamente 20 compartimentos (*bins*), uma vez que esse valor não influencia nas taxas de acerto.

6.1.4 Estratégias de treinamento e validação

Para a avaliação do desempenho do método proposto, aqui foi utilizada a estratégia de *k*-validação cruzada. (96) Quando o número de instâncias é maior a 10, foram usadas *k* = 10-validação cruzada, enquanto que para outros casos, como é o *dataset-estômatos* e *dataset-textuais* foi usado *k* = 4 e *k* = 5, respectivamente.

Além disso, o procedimento de validação cruzada foi aplicado 100 vezes. Portanto, o desvio padrão obtido está relacionado à variação de acerto para cada execução deste procedimento, uma vez que a atribuição das instâncias do conjunto de dados a cada rodada é dada aleatoriamente. Em todos os experimentos da seleção da regra foi usado o classificador kNN (k-vizinhos mais próximos) com *k* = 1, já para a avaliação do desempenho foi usado o classificador SVM (máquina de vetores de suporte) usando as configurações e parâmetros por padrão do software WEKA (199), os quais foram avaliados amplamente por Amancio et al. (97) Apenas para o conjunto de redes textuais foram comparados vários classificadores entre eles, kNN, *Naive Bayes* e SVM.

Quando os conjuntos de dados não apresentaram uma distribuição uniforme de instâncias por classes, como é o caso das redes metabólicas, usamos uma estratégia de sub-amostra aleatória. Especificamente, realizamos a classificação sob uma estratégia de reamostragem, com *k* o número mínimo de instâncias numa classe dada, usando 100

configurações aleatórias para cada grupo.

6.1.5 Estratégias de comparação com medidas estruturais de redes

Todos os experimentos apresentados neste capítulo foram comparados usando medidas estruturais de rede. As medidas estruturais correspondem ao grau médio ($\langle k \rangle$), o grau hierárquico médio de nível 2 ($\langle H_{k_2} \rangle$), o grau hierárquico médio do nível 3 ($\langle H_{k_3} \rangle$), o coeficiente de aglomeração médio ($\langle cc \rangle$), o comprimento médio do caminho ($\langle l \rangle$) e o grau de assortatividade (ρ_P), além disso, também considerou-se a combinação destas $[\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \langle l \rangle, \rho_P]$.

É importante salientar que os resultados de todos os métodos foram obtidos utilizando uma mesma configuração experimental (mesmo classificador e esquema de validação) para assim garantir uma comparação do desempenho dos métodos de forma justa.

6.2 Identificação de padrões estruturais em redes sociais online

6.2.1 Contextualização e aquisição de dados

As redes sociais são exemplos de sistemas complexos que foram estudados por muitas décadas usando diferentes abordagens teóricas. Hoje em dia, a interação por meio das redes sociais *online* está em auge, pois, oferece uma grande variedade de formas de interações sociais, através de postagens e compartilhamentos de textos, fotos, áudios, vídeos e quaisquer novas mídias que estiverem por vir. Além disso, estas ferramentas são apoiadas pelos avanços tecnológicos, podendo armazenar uma enorme quantidade de dados, permitindo uma difusão massiva em forma viral.

Tão grande é o impacto do seu uso, que incluso surgiu um novo nicho chamado de *marketing* em mídias sociais. Nela, uma simbiose de empresas, mediadores e consumidores se beneficiam e interagem, extraindo informações dos usuários acerca dos seus gostos particulares, entre outros. Diversos serviços para a construção de redes sociais podem determinar a forma como as pessoas se conectam entre si, cedendo desta forma, estruturas específicas na rede, tal é o caso das plataformas de redes sociais *Twitter* e *Google+*, que de acordo com comentários na mídia (200), parecem evidenciar diferentes formas na qual um usuário pode interagir, ao mesmo tempo que permite às empresas conhecer melhor seus clientes. A forma de usar e os propósitos destes serviços também podem ter características particulares em cada usuário. Algumas delas apresentam ferramentas para compartilhar e agrupar pessoas em círculos topográficos ou comunidades de tópicos específicos, como é o caso do *Google+*, já no caso do *Twitter* trata de um esquema de seguir e ser seguido, de acompanhar bate-papos e reações em tempo real por meio do uso de *hashtags*.

Sem adentrar mais nas questões sociais e de *marketing*, nesta seção apresentamos

um experimento para analisarmos o problema de identificação de padrões estruturais em redes sociais *online*. O objetivo deste experimento é usar o LLNA para identificar a ferramenta de software, *Twitter* e *Google+*, com a qual foi criada uma rede social particular. Para isto, usamos redes do banco de dados SNAP (*Stanford Network Analysis Project*) (197, 201), a qual providencia amostras de redes de usuários anônimos que usam tais serviços de redes sociais, que correspondem às classes desta aplicação. Todas estas amostras de redes sociais, também chamadas de “redes de ego”, representam as relações sociais ou de amizade de um usuário específico (“ego”) que não está representado na rede. Escolhemos aleatoriamente 65 redes tanto para o *Google+* como para o *Twitter*, isto para poder ter uma quantidade de redes mais uniforme para ambos os casos.

6.2.2 Seleção de regras

Para a escolha das melhores regras, a base de redes sociais foi dividida em duas partes: 15 amostras de cada uma para a seleção das melhores regras do LLNA, e as 50 restantes para o treinamento e validação do método. Avaliamos, de forma exaustiva, cada uma das 262,144 regras da família *Life-Like* no conjunto de redes sociais destinadas a seleção da regra. O LLNA foi evoluído durante $t = 350$ iterações para extrair o vetor de características composto apenas pelo descritor da distribuição da entropia de Shannon $\vec{\mu}_S$ contendo 20 descritores, os quais foram submetidos com o classificador kNN.

A Figura 35 descreve a distribuição do histograma das taxas de acerto obtidas para o espaço de regras da família *Life-Like*. Nela foram encontradas mais de 70,000 regras com taxas de acertos, para distinguir diferentes usuários das ferramentas de rede social *Twitter* e *Google+*, são superiores a 80%, das quais escolhemos as 400 melhores regras desse conjunto para o estágio de classificação, detalhado na seguinte seção.

6.2.3 Resultados da validação

Neste experimento foi usado o conjunto de redes sociais especificamente para a validação, como detalhado na Tabela 13. Para realizar a classificação empregamos as 400 melhores regras encontradas durante a seleção das regras, evoluídas durante $t = 350$ iterações.

A relação do desempenho de classificação do LLNA para este conjunto de dados são apresentadas na Tabela 14, a qual apresenta as taxas de acerto médio e desvio padrão obtidas pelas 10 melhores regras, que obtiveram as maiores taxas de acerto, usando diferentes vetores de características e suas combinações. A maior taxa de acerto para distinguir os diferentes usuários das ferramentas de rede social *Twitter* e *Google+* foi de 92 (± 1) % usando a regra B0167-S248, sendo um empate entre a distribuição da complexidade Lempel-Ziv ($\vec{\mu}_L$) e a da entropia de Shannon $\vec{\mu}_S$ como vetores de características. Nesta tabela também observamos que a maioria de regras do LLNA são altas para os diferentes

Figura 35 – Histograma da distribuição das 262,144 regras da família *Life-Like* em função da taxa de acerto atingida usando a base de redes sociais para a seleção de regras. A entropia de Shannon foi usada como vetor de características e classificador kNN.

Fonte: Elaborada pela autora.

vetores de características.

Tabela 14 – Taxas de acerto (%) para a classificação das redes sociais usando o LLNA com diferentes combinações de medidas como vetor de características classificadas por SVM. A última linha apresenta o número de descritores de cada vetor de características.

Regras	$\vec{\mu}_S$	$\vec{\mu}_W$	$\vec{\mu}_L$	$[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$	$[\langle\mu_S\rangle, \langle\mu_W\rangle, \langle\mu_L\rangle]$
B0167-S248	92 (± 1)	88 (± 1)	92 (± 1)	89 (± 1)	83 (± 2)
B05-S148	76 (± 1)	82 (± 2)	79 (± 2)	89 (± 1)	81 (± 1)
B06-S148	79 (± 2)	79 (± 2)	75 (± 2)	78 (± 2)	81 (± 2)
B256-S3467	89 (± 1)	78 (± 2)	87 (± 1)	80 (± 1)	80 (± 1)
B05-S1458	84 (± 1)	80 (± 1)	81 (± 2)	82 (± 1)	80 (± 2)
B1236-S245	82 (± 2)	81 (± 1)	84 (± 1)	86 (± 1)	79 (± 1)
B456-S01237	80 (± 2)	65 (± 2)	78 (± 2)	82 (± 2)	79 (± 1)
B01267-S035678	78 (± 2)	75 (± 2)	69 (± 2)	70 (± 2)	79 (± 2)
B012368-S01245	85 (± 1)	82 (± 1)	76 (± 1)	78 (± 1)	79 (± 2)
B08-S1467	84 (± 1)	78 (± 3)	78 (± 2)	75 (± 1)	79 (± 2)
ND	20	20	20	60	3

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

A Figura 36a ilustra as diferenças em relação à dinâmica espaço-temporal de cada rede social, *Twitter* e *Google+*, nela observa-se o vetor de características da complexidade de Lempel-Ziv ($\vec{\mu}_L$) representada como um gráfico de barras. Cada gráfico representa uma amostra de cada uma das duas classes de redes sociais.

As redes da classe *Twitter* submetidas pela dinâmica da regra B0167-S248 possuem padrões espaço-tempo com valores da complexidade de Lempel-Ziv ($\vec{\mu}_L$) concentrados entre 0.6 e 0.9, enquanto o histograma do *Google+* apresenta os valores de Lempel-Ziv distribuídos pelo histograma, com picos no início. Portanto a separação de ambas as classes é evidente, não somente no gráfico de barras como também na Figura 36b, a qual apresenta a análise de variáveis canônicas usando os mesmos parâmetros e mesmo vetor de

características. Além disso, na Figura 36c é apresentada a matriz de confusão para o melhor caso relatado na última experiência usando a regra B0167-S248 e $\vec{\mu}_L$ como descritor.

Figura 36 – (a) Histograma da distribuição da complexidade de Lempel-Ziv $\vec{\mu}_L$ para duas amostras de cada classe de redes sociais. (b) Projeção bidimensional da análise de variáveis canônicas correspondente ao vetor de características em a). c) Matriz de confusão alcançado pela melhor taxa de acerto do vetor $\vec{\mu}_L$ usando a regra B0167-S248. Na matriz de confusão, cada célula mostra o número de instâncias preditas corretamente.

Fonte: Elaborada pela autora

Quando comparado ao desempenho das medidas estruturais (ver Tabela 15), o LLNA também supera a taxas de acerto obtida ao usar a combinação dessas medidas como vetor de características, $88 (\pm 2)\%$. Portanto, foi obtido um aumento na taxa de classificação de $4(\pm 3)\%$ para o *dataset-social*.

6.2.4 Discussões

Como visto pelos resultados apresentados, as diferenças fornecidas por cada ferramenta de rede social podem influenciar a forma como as pessoas se conectam entre si, resultando em diferenças estruturais entre ambas as redes sociais, embora existam algumas propriedades, como a conexão preferencial de vértices e a presença de *hubs* em ambos. A taxa de acerto máxima obtida com LLNA para esta aplicação foi de $92 (\pm 1)\%$ em contraste com $88 (\pm 2)\%$ usando a combinação de cinco medidas estruturais como vetor de características, porém, em termos de desvio padrão ambos estariam em empate técnico.

Tabela 15 – Comparação da taxa de acerto (%) para a classificação de redes sociais usando medidas estruturais de rede: grau médio ($\langle k \rangle$), o grau hierárquico médio de nível 2 ($\langle H_{k_2} \rangle$), o grau hierárquico médio do nível 3 ($\langle H_{k_3} \rangle$), o coeficiente de aglomeração médio ($\langle cc \rangle$), o comprimento médio do caminho ($\langle l \rangle$) e o grau de assortatividade (ρ_P), e a combinação destas [$\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \langle l \rangle, \rho_P$]. A última linha apresenta os resultados obtidos pelo LLNA usando a regra B0167-S248 e $\vec{\mu}_L$ como vetor de características.

Descritor	Acerto (%)
$\langle k \rangle$	80 (± 1)
$\langle H_{k_2} \rangle$	87 (± 1)
$\langle H_{k_3} \rangle$	84 (± 1)
$\langle cc \rangle$	59 (± 2)
$\langle l \rangle$	50 (± 2)
ρ_P	62 (± 2)
$[\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \langle l \rangle, \rho_P]$	88 (± 2)
LLNA (B0167-S248)	92 (± 1)

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

6.3 Identificação de organismos em redes metabólicas

6.3.1 Contextualização e aquisição de dados

As redes metabólicas descrevem as reações químicas das vias metabólicas que regem as transformações entre compostos químicos através da ação de enzimas, as quais ocorrem nas células. Existem vários modelos para representar tais redes, uma das mais conhecidas é a rede substrato-enzima (rede dirigida). (82) Nesse modelo (ver Figura 37), os substratos (vértices) estão conectados uns aos outros através de arestas, que são as reações metabólicas reais, onde as enzimas são os catalisadores para que tais reações produzam produtos, e eventualmente estes produtos podem se tornar substratos para reações subsequentes. (23)

Figura 37 – Exemplo da modelagem de uma porção de rede metabólica baseado em duas reações enzimáticas, simplificadas, da espécie *Arabidopsis thaliana*. Cada substrato é representado como um vértice, ligado através de relações substrato-enzima dos quais os produtos emergem como novos substratos (vértices). As enzimas catalisadoras (EC) são representadas pelos números acima das arestas (caixas pretas).

Fonte: Adaptada de JEONG et al. (23)

Muitos pesquisadores têm se concentrado no cerne das redes metabólicas desde que Jeong et al. (23) introduziram a obra intitulada “A organização livre de escala das

redes metabólicas”*. De fato, esta propriedade indica que a probabilidade de encontrar um vértice altamente conectado é conservada entre os seres vivos. (23) Isto porque, em geral, a existência de *hubs* é uma das características mais evidentes das redes livre de escala (14) Desde então, explorar a propriedade livre de escala das redes metabólicas e como as redes não biológicas são estudadas, trouxe grande progresso na ciência. (14, 202, 203) Porém, apesar da similaridade destas redes em termos de serem livre de escala, a inerente topologia das redes metabólicas vem sendo entendida como ferramenta alternativa para extrair relações filogenéticas. (204–207) Recentemente, Banerjee (205) explorou as propriedades globais das redes metabólicas para agrupar 43 organismos pertencentes a uma grande extensão de seres vivos em seus respectivos grandes grupos taxonômicos. (205)

Dessa forma, seguindo a linha de pesquisa de Banerjee (205), o objetivo desta aplicação é caracterizar as redes metabólicas de organismos distintos agrupados por classes evolutivas usando o LLNA. Nesta aplicação, foram investigadas três classes distintas de organismos. Para isso, foi utilizado o mesmo conjunto de dados empregado previamente por Jeong et al. (23) e Banerjee (205), o qual se encontra publicamente disponível em. (208) Este conjunto de dados contém 43 redes metabólicas, que fornece uma descrição das vias metabólicas de três domínios de vida: *Archaea*, *Bacterium* e *Eukaryote*. O banco de dados original foi construído com base nas reações metabólicas encontradas na base de dados do WIT. (83) Essas redes metabólicas foram geradas considerando os complexos substrato-produto (ver Figura 37) e as enzimas associadas como representações de vértices e arestas, respectivamente.

O conjunto de dados das redes metabólicas contém 43 amostras que fornecem uma descrição das vias metabólicas de três tipos de organismos: 6 *archaea*, 32 *bacteria* e 5 *eukaryotes* (ver Tabela 16). No que tange ao tamanho das redes, a maior e menor rede contém 819 e 178 metabólitos correspondentes às espécies *Salmonella typhi* e *Mycoplasma pneumoniae*, respectivamente. Além disso, todas estas redes possuem aproximadamente $\gamma = 2.2$ (expoente da lei de potência). (23)

6.3.2 Seleção da regra

Como apresentado na Tabela 16, os dados foram dividido em dois conjuntos: seleção de regras e validação da classificação. O primeiro conjunto contém 2 amostras selecionadas aleatoriamente de cada classe, que foram usadas para encontrar as melhores regras da família *Life-Like* em relação à sua taxa de acerto na distinção entre as classes de organismos; e o segundo conjunto consiste nas redes restantes.

Cada uma das 262,144 regras da família *Life-Like* foram avaliadas exaustivamente no conjunto de redes metabólicas destinadas a seleção da regra. A distribuição do histograma das taxas de acerto obtidas para o espaço de regras da família *Life-Like* é descrita na

* *The large-scale organization of metabolic networks.* (23)

Tabela 16 – Lista dos 43 organismos da base de redes metabólicas divididas em três classes (domínios). As duas primeiras linhas correspondem às redes com as quais foram selecionadas as melhores regras para o problema de identificação de organismos.

Classe <i>Archaea</i>	Classe <i>Bacterium</i>	Classe <i>Eukaryote</i>
MJ <i>Methanococcus jannaschii</i>	AA <i>Aquifex aeolicus</i>	CE <i>Caenorhabditis elegans</i>
AP <i>Aeropyrum pernix</i>	CT <i>Chlamydia trachomatis</i>	AT <i>Arabidopsis thaliana</i>
AG <i>Archaeaoglobus fulgidus</i>	CQ <i>Chlamydia pneumoniae</i>	EN <i>Emericella nidulans</i>
TH <i>Methanobacterium thermoaut.</i>	CY <i>Synechocystis sp.</i>	SC <i>Saccharomyces cerevisiae</i>
PF <i>Pyrococcus furiosus</i>	PG <i>Porphyromonas gingivalis</i>	OS <i>Oryza sativa</i>
PH <i>Pyrococcus horikoshii</i>	MB <i>Mycobacterium bovis</i>	
	ML <i>Mycobacterium leprae</i>	
	MT <i>Mycobacterium tuberculosis</i>	
	BS <i>Bacillus subtilis</i>	
	EF <i>Enterococcus faecalis</i>	
	TM <i>Thermotoga maritima</i>	
	DR <i>Deinococcus radiodurans</i>	
	CA <i>Clostridium acetobutylicum</i>	
	MG <i>Mycoplasma genitalium</i>	
	MP <i>Mycoplasma pneumoniae</i>	
	PN <i>Streptococcus pneumoniae</i>	
	ST <i>Streptococcus pyogenes</i>	
	CL <i>Chlorobium tepidum</i>	
	YP <i>Yersinia pestis</i>	
	AB <i>Actinobacillus actinomycet.</i>	
	HI <i>Haemophilus influenzae</i>	
	PA <i>Pseudomonas aeruginosa</i>	
	RC <i>Rhodobacter capsulatus</i>	
	RP <i>Rickettsia prowazekii</i>	
	NG <i>Neisseria gonorrhoeae</i>	
	NM <i>Neisseria meningitidis</i>	
	CJ <i>Campylobacter jejuni</i>	
	HP <i>Helicobacter pylori</i>	
	EC <i>Escherichia coli</i>	
	TY <i>Salmonella typhi</i>	
	TP <i>Treponema pallidum</i>	
	BB <i>Borrelia burgdorferi</i>	

Fonte: Adaptada de JEONG et al. (23)

Figura 38. Nela foram encontradas mais de 4,000 regras com taxas de acertos superiores a 80%, pelo qual escolhemos as 400 melhores regras desse conjunto para o estágio de classificação que será apresentada na seção seguinte.

6.3.3 Resultados da validação

Neste experimento foi usado o conjunto de redes para validação como detalhado na Tabela 13. Para realizar a classificação foram usadas as 400 melhores regras encontradas durante a seleção das regras. O LLNA foi usado para caracterizar as redes metabólicas. Foram usados os vetores de características $\vec{\mu}_S$, $\vec{\mu}_W$ e $\vec{\mu}_L$ cada um contendo 20 descritores, também foi considerada a combinação destes, $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$, contendo 60 descritores, assim como as médias destes descritores $\langle \mu_S \rangle$, $\langle \mu_W \rangle$, $\langle \mu_L \rangle$, contendo 3 descritores.

A classificação deste conjunto foi feita seguindo a estratégia de sub-amostragem aleatória k -validação cruzada, com $k = 3$ usando 100 configurações aleatórias para cada grupo, isto porque o conjunto de dados não apresenta uma distribuição uniforme das classes.

Figura 38 – Histograma da distribuição das 262,144 regras da família *Life-Like* em função da taxa de acerto atingida para a seleção de regra usando a base de redes metabólicas. A entropia de Shannon foi considerada como vetor de características correspondentes e classificador de kNN.

Fonte: Elaborada pela autora.

A Tabela 17 apresenta a taxas de acerto médio e desvio padrão para a classificação dos diferentes domínios usando o conjunto de redes metabólicas. Na tabela são apresentadas as 10 melhores regras, que obtiveram as maiores taxas de acerto, usando o classificador SVM para várias combinações de vetores de características do LLNA. A maior taxa de acerto obtida foi de 87 (± 13)% usando a regra B05-S13568 e a distribuição da entropia de Shannon $\vec{\mu}_S$ como vetor de características. Nesta tabela observa-se que existe uma ampla variedade nas taxas de acertos com as demais regras, porém a maioria dela está acima de 60%.

Tabela 17 – Taxas de acerto (%) para a classificação de redes metabólicas usando o LLNA com diferentes combinações de medidas como vetor de características classificadas por SVM. A última linha apresenta o número de descritores de cada vetor de características.

Regras	$\vec{\mu}_S$	$\vec{\mu}_W$	$\vec{\mu}_L$	$[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$ [$\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle$]	
B05-S13568	87 (± 13)	61 (± 16)	52 (± 18)	77 (± 15)	63 (± 14)
B0346-S03	62 (± 15)	62 (± 7)	52 (± 16)	68 (± 15)	53 (± 12)
B02468-S7	64 (± 16)	33 (± 0)	61 (± 14)	67 (± 15)	54 (± 16)
B0578-S1356	74 (± 15)	60 (± 11)	57 (± 14)	62 (± 14)	65 (± 14)
B01237-S13	68 (± 14)	33 (± 0)	55 (± 16)	62 (± 16)	39 (± 16)
B01247-S2347	61 (± 16)	56 (± 10)	32 (± 15)	57 (± 18)	35 (± 15)
B3468-S12346	39 (± 16)	49 (± 12)	69 (± 16)	54 (± 15)	28 (± 16)
B2345678-S367	27 (± 15)	67 (± 2)	38 (± 17)	48 (± 17)	49 (± 16)
B23468-S45	40 (± 16)	37 (± 14)	42 (± 14)	44 (± 20)	33 (± 18)
B01678-S013578	40 (± 16)	31 (± 16)	33 (± 16)	37 (± 18)	71 (± 16)
ND	20	20	20	60	3

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

A Figura 39a apresenta o vetor de características da entropia de Shannon ($\vec{\mu}_S$), usando a regra B05-S13568, representada como um gráfico de barras. Cada uma dos gráficos representa uma amostra de cada uma das três classes de organismos. Esta distribuição foi

obtida do padrão espaço-tempo a partir da dinâmica do LLNA sobre as respectivas redes metabólicas. Pode ser observado que há distintas distribuições para cada uma das três classes, por exemplo, a rede da classe *eukaryote* possui um alta frequência de entropias baixas, o que pode ser entendido como a presença de padrões estáveis e/ou oscilantes no respectivo diagrama espaço-tempo. A separação da classe *eukaryote* é evidente também na Figura 39b, a qual apresenta a análise de variáveis canônicas usando os mesmos parâmetros e mesmo vetor de características que as amostras apresentadas na Figura 39a

Figura 39 – (a) Histograma da distribuição da entropia de Shannon, μ_S , para as três amostras de cada categoria de organismos das redes metabólicas. (b) Projeção bidimensional da análise de variáveis canônicas correspondente ao vetor de características em a).

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

Finalmente, na Tabela 18 podemos observar a comparação entre a melhor taxa de acerto obtida com LLNA e as taxas de acerto obtidas com diferentes medidas estruturais de redes. Nesse caso, o LLNA forneceu um aumento na taxa de acerto da classificação de $23 (\pm 23)\%$ quando comparado ao descritor do coeficiente de aglomeração que forneceu a melhor taxa de acerto entre as medições de rede, $64 (\pm 10)\%$.

Tabela 18 – Comparação da taxa de acerto (%) para a classificação das redes metabólicas usando medidas estruturais de rede como vetor de características. A última linha apresenta os resultados obtidos para LLNA usando a regra B05-S13568 e μ_S como vetor de características.

Descritor	Acerto (%)
$\langle k \rangle$	35 (± 14)
$\langle H_{k_2} \rangle$	30 (± 13)
$\langle H_{k_3} \rangle$	30 (± 12)
$\langle cc \rangle$	64 (± 10)
$\langle l \rangle$	48 (± 15)
ρ_P	46 (± 14)
$[\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \langle l \rangle, \rho_P]$	57 (± 16)
LLNA (B05-S13568)	87 (± 13)

Fonte: Adaptada de MIRANDA; MACHICAO; BRUNO (50)

6.3.4 Discussões

Com respeito à análise das redes metabólicas, no estudo original de Jeong et al. (23), os autores mostraram que organismos de classes evolutivas distintas apresentam redes metabólicas com topologia similar, ou seja, com características de redes livre de escala. Porém, neste estudo, mostramos que os algoritmos do LLNA podem dar um passo adiante em termos de análise da topologia de rede, pois, o LLNA é capaz de encontrar sutilezas que podem ser usadas para distinguir redes dentro do mesmo grupo topológico, permitindo a caracterização de sub-categorias de redes. A maior taxa de acerto foi obtida com LLNA foi de $87 (\pm 13)\%$ em contraste com $64 (\pm 10)\%$ usando o coeficiente de aglomeração como vetor de características.

6.4 Análise da plasticidade fenotípica de plantas em redes estomáticas

6.4.1 Contextualização e aquisição de dados

Os estômatos são células que atuam como válvulas localizadas nas folhas e caule de plantas encarregadas da troca gasosa com o ambiente. (209) Dependendo da espécie e das condições ambientais, os estômatos variam em tamanho de aproximadamente 10 a $80 \mu m$ em comprimento, enquanto o número de estômatos por área de folha (densidade estomática) varia entre 5 e 1000 por mm^2 de epiderme.

De acordo com a literatura, a plasticidade fenotípica dos estômatos nas plantas é fortemente relacionada com as variações ambientais. (210) De fato, vários tipos de estresse ambiental como a ingestão alta/baixa de CO_2 e vapor de água (umidade) (211), exposição alta/baixa de luz (natural ou artificial) (212), entre outras habilidade das plantas de adaptarem a capacidade fotossintética quando sob diferentes níveis de qualidade de iluminação (213), fazem com que a epiderme da planta seja altamente plástica, ou seja, a densidade estomática pode ser variável dependendo das condições ambientais.

Compreender onde e como a plasticidade funciona nas plantas é um desafio importante no contexto das mudanças climáticas observadas nos últimos tempos. (214) Portanto, o estudo da geometria dos estômatos pode levar a resposta sobre a forma como as mudanças ambientais afetam as plantas. Assim, no contexto de reconhecimento de padrões, o problema de caracterização de plasticidade fenotípica pode ser contornado sob a perspectiva de redes, no qual os estômatos representem os vértices, e suas relações de densidade estomática são representadas por arestas. Ao contrário de outros trabalhos (215) que visam a identificação de espécies, nos quais a invariância à plasticidade é altamente desejável, com esta base de redes foi objetivado a classificação das condições ambientais de uma espécie em específico.

As bases de dados de imagens de estômatos foram adquiridas pelo grupo de Computação Científica do Instituto de Física de São Carlos (IFSC) a cargo do Prof.

Odemir M. Bruno em parceria com o doutorando Humberto Antunes de Almeida Filho. Esta base contém imagens microscópicas de estômatos de plantas[†] e suas correspondentes imagens binárias (segmentadas), a partir da qual foram acessadas às suas respectivas coordenadas dos centróides das posições espaciais de cada estômato. Esta base corresponde a três espécies de plantas que foram submetidas a uma das duas diferentes condições de estresse: exposição de luz e injeção de CO₂, de forma excludente, detalhadas a seguir:

- A espécie *Ctenanthe oppenheimiana* foi submetida a duas condições de luz artificial de 4 e 24 horas (L4 e L24), respectivamente, durante um período contínuo de 60 dias em estufa com condições controladas. Logo, foram obtidas imagens microscópicas destas correspondentes a um aumento de 10x com 2560×1920 pixels (resolução do microscópio). Há um total de 18 imagens para cada condição. Exemplos destas imagens podem ser observadas na Figura 40a.
- A espécie *Tradescantia zebrina* foi submetida a três condições de iluminação: luz natural (NAT) e luz artificial por 4 e 24 horas (L4 e L24), respectivamente. As plantas foram expostas a estas condições durante 69 dias em estufa com condições controladas. Logo, foram obtidas imagens destas na lupa com um aumento de 5x, contendo 1388×1038 pixels (resolução da lupa). Há um total de 6 imagens para cada condição, as quais foram usadas em (214). Exemplos destas imagens podem ser observadas na Figura 40b;
- A espécie *Callisia repens* foi submetida a estresse de CO₂: fluxo contínuo (CO₂ ↑) e fluxo intermitente (CO₂ ↓), durante um período de 30 dias. As plantas correspondem ao cultivo de brotos divididas em 3 grupos experimentais. O primeiro grupo foi lacrado com válvula injetora de gás CO₂ gasoso sob fluxo contínuo por 6 horas. O segundo grupo experimental teve frascos lacrados durante o mesmo período mas não recebiam fluxo contínuo de CO₂ gasoso. Em ambos os grupos artificiais a umidade relativa foi mantida sob variação de 85 a 90%. Um terceiro grupo de plantas foi coletado na mesma área de crescimento das plantas do primeiro e do segundo grupo experimental mas sob solo, temperatura e umidade ambientes (AMB). Estas imagens correspondem a um aumento de 5x e possuem 2560×1920 pixels (resolução do microscópio). Há um total de 5 imagens para cada uma das 3 condições. Exemplos destas imagens podem ser observadas na Figura 40c;

6.4.2 Modelagem da rede de estômatos

A Figura 40 exemplifica os passos para a construção do modelo de rede estomática, o qual foi proposto por Florindo et al. (214), o qual foi adaptado da proposta de Backes et

[†] As imagens foram obtidas de duas procedências. A espécie *Callisia repens* e *Ctenanthe oppenheimiana* foram coletadas usando um microscópio ICC5 (resolução por padrão de 2560×1920) acoplado a uma câmera Axiocam ERc5s. Já a espécie *Tradescantia zebrina* foi coletada usando uma lupa estéreo Zeiss discovery V.20 acoplado a uma câmera Axiocam ERc1s. (214)

al. (17) para análise de contornos na área de Visão Computacional. Assim, cada estômato corresponde a um vértice da rede, e o critério para estabelecer as arestas é baseado na distância euclidiana entre pares de estômatos dependendo de um limiar T_l .

Cada imagem contém diferentes números de estômatos (vértices) os quais estão no centro de uma circunferência, desta forma o limiar T_l define o raio de conectividade de um estômato com os vizinhos mais próximos a este. Inicialmente, a partir de cada imagem é obtida uma rede de pesos totalmente conectada, contendo as distâncias euclidianas entre cada par de vértices (W). Esta matriz é então transformada em várias redes representadas pela matriz de adjacência A_{T_l} dependendo do limiar T_l , dado pela Equação (6.1), ou seja, se a distância entre pares de estômatos for menor ou igual a esse valor, então uma aresta é estabelecida entre eles.

$$A_{T_l} = \delta_{T_l}(W) = \forall w \in W \begin{cases} a_{ij} = 0 & \text{se } w_{ij} > T_l \\ a_{ij} = 1 & \text{se } w_{ij} \leq T_l \end{cases} \quad (6.1)$$

Já que a caracterização destas redes é realizada através de uma série de δ transformações em que T_l é incrementado em intervalos de T_{inc} , com T_{ini} e T_Q , o valor inicial e final do limiar, respectivamente. (17) De cima para abaixo, a Figura 40 apresenta uma amostra de cada espécie de planta e sua correspondente condição ambiental. A Figura 40a ilustra a transformação $\delta_{T_l}(W)$ seguindo a Equação (6.1). Na parte esquerda da figura, é apresentada a imagem microscópica original de uma folha da espécie *Ctenanthe oppenheimiana*. O círculo amarelo representa os centróides dos estômatos que correspondem à imagem binária já segmentada (não mostrada na figura). No lado direito são mostradas 4 redes resultantes dado um limiar de raio. A cada transformação δ_{T_l} , o limiar T_l é incrementado a pequenos intervalos T_{inc} . No lado direito mostra a rede resultante dado um limiar de raio $T_{ini} = 100\mu m$, até $T_Q = 160\mu m$, com T_{ini} e T_Q , o valor inicial e final do limiar respectivamente. Observe que à medida que T_l aumenta, mais conexões serão estabelecidas entre os centróides dos estômatos e, a densidade da rede será maior.

Considerando que diferentes redes podem ser obtidas, o limiar T_l precisa ser limitado em relação ao tamanho da imagem. Observe que, dado que as coordenadas dos centróides foram dadas em escala real da planta, logo as imagens em pixels podem ser convertidas usando um fator de conversão. Aqui foi usado o fator de escala 0.347 que equivale a $100\mu m$. Observe também que, nas redes das espécies *Tradescantia zebrina* e *Callisia repens*, os limiares são maiores, isto porque as amostras das imagens estão em uma escala menor, logo maior será a densidade estomática.

Nas Figuras 40a-b, pode ser observado que a epiderme das plantas expostas a 24h de iluminação diária possuem densidade estomática maior que as de plantas expostas por 4h. Devido a esse fato, as redes construídas com base nas distâncias limiares (T_l) tornam-se altamente conectadas nas imagens onde a densidade estomática é maior de acordo com o

Figura 40 – Modelagem de rede estomática. Na primeira coluna são mostradas imagens microscópicas de folhas da espécie (a) *Ctenanthe oppenheimiana*, (b) *Tradescantia zebrina* e c) *Calissia repens*, condicionadas à exposição de 4h e 24h de iluminação por dia, injeção em fluxo contínuo e intermitente de CO₂, e em condições naturais, respectivamente. Nas colunas da esquerda para a direita, são mostrados várias redes estomáticas, as quais são obtidas pela transformação de um determinado raio de limiar de T_l (círculo amarelo) os quais delimitam a conectividade da rede (vértices em verde, e arestas em cinza).

Fonte: Elaborada pela autora.

acréscimo do limiar. Devido à alta plasticidade da planta, o número de estômatos (vértices) de plantas expostas a 24h diárias de iluminação é proporcionalmente maior que em plantas expostas a 4h para o mesmo intervalo de limiar. De forma similar, na Figura 40c, que

corresponde à espécie *Callisia repens* é observado quanto maior injeção de CO_2 maior a densidade estomática e portanto maior a conectividade.

6.4.3 Seleção da regra

Para a escolha das melhores regras, a base de imagens da espécie *Tradescantia zebrina* que contém em total 18 imagens foi dividida em 6 por cada condição. Como apresentado na Tabela 13, apenas 2 imagens por cada classe são usadas para a seleção da regra enquanto o restante foi usado para avaliar o desempenho dos classificadores. Observe que mais em frente na Seção 6.4.4, essas mesmas regras são aplicadas não somente para o cenário da espécie *Tradescantia zebrina*, como também nas outras duas espécies, pelo que estas regras parecem satisfazer este contexto específico.

Avaliamos, de forma exaustiva, cada uma das 262,144 regras da família *Life-Like* no conjunto de redes estomáticas destinadas a seleção da regra. As redes deste conjunto de dados foram obtidas apenas usando o limiar $T_l = 110\mu m$. Para caracterizar a dinâmica do LLNA, foi extraído um vetor de características composto pela concatenação dos descritores da entropia de Shannon e as distribuições de Lempel-Ziv $[\vec{\mu}_S, \vec{\mu}_L]$ que consistem em 40 descritores, ou seja, ambas as distribuições são janeladas em 20 compartimentos, as quais foram submetidas com o classificador kNN.

A Figura 41 descreve a distribuição do histograma das taxas de acerto obtidas para o espaço de regras da família *Life-Like*. Nela foram encontradas mais de 5,000 regras com taxas de acertos superiores a 80%, pelo qual escolhemos as 400 melhores regras desse conjunto para o estágio de classificação. Na seguinte seção são apresentados três experimentos com redes estomáticas. Da mesma forma que a fase de seleção da regra, o primeiro experimento também visa a classificação de redes da espécie *Tradescantia zebrina*, *Callisia repens* e *Ctenanthe oppenheimiana*.

6.4.4 Resultados da validação

Foi utilizado o método de LLNA para caracterizar a plasticidade fenotípica das espécies *Tradescantia zebrina*, *Callisia repens* e *Ctenanthe oppenheimiana* quando submetida a condições de estresse. Neste experimento foi feito o uso do conjunto de redes para validação como detalhado na Tabela 13. Para realizar a classificação foram usadas as 400 melhores regras encontradas durante a seleção das regras.

Para esta aplicação nas redes estomáticas seguiu-se a estratégia de concatenação dos descritores obtidos das redes transformadas por cada valor de δT_l , proposta por Backes et al. (17) Portanto, foram usados 16 limiares desde $T_{ini} = 50\mu m$, até $T_Q = 200\mu m$, com incrementos de $T_{inc} = 10\mu m$. Foram usados os vetores de características $\vec{\mu}_S$, $\vec{\mu}_W$ e $\vec{\mu}_L$, os quais contêm a concatenação dos 16 limiares, ou seja, cada uma contendo 20×16 descritores, respectivamente. Também foi avaliada a combinação destes, $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$, contendo 60×16

Figura 41 – Histograma da distribuição das 262,144 regras da família *Life-Like* em função da taxa de acerto atingida. Para a seleção de regra usou-se a base de imagens de estômatos da espécie *Tradescantia zebrina*. A entropia de Shannon e complexidade de Lempel-Ziv foram consideradas como vetores de características correspondentes e classificador de kNN.

Fonte: Elaborada pela autora.

descritores, assim como as médias destes descritores $\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle$, contendo 3×16 descritores.

A Tabela 19 exibe os resultados da 10 melhores regras para a classificação das redes estomáticas das três espécies analisadas usando o classificador SVM para cada vetor de características. A maior taxa de acerto obtida para o conjunto da espécie *Tradescantia zebrina* foi de $90 \pm 6\%$ usando a regra B12345-S04568 e $[\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle]$ como vetor de características. Para a espécie *Callisia repens* a maior taxa de acerto foi de $95 \pm 5\%$ usando a regra B12345-S04568 e $\vec{\mu}_S$ como vetor de características. Já para o conjunto da espécie *Ctenanthe oppenheimiana* foi de $95 \pm 2\%$ usando a regra B57-S378 e $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$ como vetor de características. Nos três casos, observa-se que o desvio padrão para estes conjuntos de dados é maior devido ao pequeno número de instâncias para cada classe. Nesta tabela também é possível observar que há uma maior heterogeneidade em relação ao comportamento das regras para os diferentes vetores de características.

No lado esquerdo da Figura 42 é apresentada a projeção dimensional obtida pela análise de variáveis canônicas para o vetor características que obteve a maior taxa de acerto, para cada uma das espécies dos três conjuntos, e a matriz de confusão correspondente (lado direito). De cima para baixo, os vetores e regras usados foram: $[\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle]$ com regra B12345-S0456 (*Tradescantia zebrina*), $\vec{\mu}_S$ com regra B014-S04567 (*Callisia repens*) e $[\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle]$ com regra B57-S378 (*Ctenanthe oppenheimiana*). Destas figuras, de cima para baixo, fica evidente que existe uma separação clara entre os três modelos de rede que corroboram as altas taxas de acertos. Nela também observa-se que a classe de L24 é linearmente separável dos outros, e que as classes natural e L4 foram confundidas,

Tabela 19 – Taxas de acerto (%) para a classificação da plasticidade fenotípica em plantas de três diferentes espécies usando o LLNA com diferentes combinações de medidas como vetor de características classificadas por SVM. A última linha apresenta o número de descritores de cada vetor de características.

Regras	$\vec{\mu}_S$	$\vec{\mu}_W$	$\vec{\mu}_L$	$[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$	$[\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle]$
<i>Tradescantia zebrina</i>					
B12345-S04568	87 (± 7)	58 (± 11)	76 (± 8)	88 (± 7)	90 (± 6)
B01346-S0257	37 (± 9)	90 (± 6)	65 (± 8)	83 (± 6)	54 (± 9)
B3567-S0468	64 (± 11)	44 (± 10)	88 (± 8)	78 (± 9)	69 (± 14)
B0568-S1578	88 (± 7)	56 (± 10)	47 (± 9)	71 (± 7)	61 (± 10)
B125678-S012345	68 (± 13)	54 (± 9)	87 (± 7)	66 (± 7)	84 (± 9)
B34-S024568	88 (± 7)	40 (± 7)	84 (± 10)	85 (± 7)	68 (± 9)
B35678-S012578	60 (± 9)	53 (± 10)	75 (± 9)	68 (± 10)	86 (± 8)
B02345-S01456	70 (± 9)	84 (± 8)	75 (± 8)	81 (± 9)	65 (± 8)
B056-S0348	66 (± 6)	85 (± 9)	66 (± 10)	67 (± 7)	63 (± 10)
B245-S0234568	77 (± 8)	34 (± 7)	87 (± 7)	78 (± 9)	63 (± 9)
<i>Callisia repens</i>					
B014-S04567	95 (± 5)	24 (± 9)	75 (± 10)	66 (± 10)	71 (± 8)
B016-S023458	67 (± 10)	59 (± 12)	87 (± 9)	75 (± 11)	90 (± 7)
B0238-S04567	80 (± 6)	55 (± 9)	90 (± 7)	81 (± 8)	57 (± 6)
B034-S012456	61 (± 9)	46 (± 7)	58 (± 8)	57 (± 10)	89 (± 11)
B5678-S01256	60 (± 9)	26 (± 9)	43 (± 8)	46 (± 9)	84 (± 8)
B023456-S01678	56 (± 7)	86 (± 6)	51 (± 6)	56 (± 6)	68 (± 8)
B01245-S12456	84 (± 9)	45 (± 9)	71 (± 8)	75 (± 10)	79 (± 7)
B012678-S0123468	83 (± 5)	31 (± 9)	70 (± 9)	68 (± 12)	51 (± 8)
B3567-S025	73 (± 9)	49 (± 8)	83 (± 7)	71 (± 8)	55 (± 7)
B3567-S0256	68 (± 9)	59 (± 8)	75 (± 9)	77 (± 9)	83 (± 6)
<i>Ctenanthe oppenheimiana</i>					
B57-S378	61 (± 5)	87 (± 4)	75 (± 2)	95 (± 2)	77 (± 4)
B0145-S0124568	90 (± 3)	73 (± 4)	91 (± 4)	93 (± 3)	71 (± 4)
B3568-S0256	75 (± 4)	64 (± 5)	79 (± 4)	77 (± 4)	91 (± 2)
B12378-S23457	87 (± 3)	75 (± 5)	69 (± 4)	82 (± 3)	90 (± 2)
B0125678-S12568	88 (± 4)	78 (± 5)	69 (± 3)	73 (± 5)	82 (± 2)
B3578-S0157	85 (± 2)	57 (± 4)	90 (± 4)	82 (± 4)	72 (± 4)
B0124568-S01457	81 (± 3)	84 (± 3)	82 (± 3)	92 (± 2)	77 (± 4)
B01238-S3458	68 (± 4)	88 (± 2)	65 (± 4)	79 (± 2)	77 (± 2)
B14678-S0345678	66 (± 3)	77 (± 4)	66 (± 4)	90 (± 3)	55 (± 3)
B1467-S0345678	62 (± 3)	77 (± 5)	64 (± 4)	91 (± 3)	52 (± 3)
ND	320	320	320	60	48

Fonte: Elaborada pela autora.

tal como indicado pela matriz de confusão, o qual também condiz com a alta similaridade destas imagens de estômatos na Figura 40b.

A Tabela 20 apresenta a comparação entre as taxas de acerto obtidas pelo LLNA e pelas medidas estruturais de redes. Para a espécie *Tradescantia zebrina*, o LLNA proporcionou uma melhora na taxa de classificação de 7 (± 10)% quando comparada com o grau hierárquico que obteve 83 (± 4)%. Já a espécie *Callisia repens* superou altamente em 31(± 9)% quando comparada com a combinação de medidas que obteve 64 (± 7)%. Note que, neste caso particular, quando usado o classificador kNN as taxas de acerto do LLNA também foram superiores aos das medidas estruturais porém numa margem menor de 17 (± 12)% (dados não mostrados). Finalmente, para a espécie *Ctenanthe oppenheimiana*, o LLNA proporcionou uma melhora na taxa de classificação de 15 (± 5)% quando comparada com o grau médio que obteve 80 (± 3)%.

(a) *Tradescantia zebrina*

		Classe Predita		
		NAT	L4	L24
Classe real	NAT	3	1	0
	L4	1	3	0
	L24	0	0	4

(b) *Callisia repens*

		Classe Predita		
		AMB	+ CO2 (↑)	+ CO2 (↓)
Classe real	AMB	5	0	0
	+ CO2 (↑)	0	5	0
	+ CO2 (↓)	0	0	5

(c) *Ctenanthe oppenheimiana*

		Classe Predita	
		L4	L24
Classe real	L4	18	0
	L24	1	17

Figura 42 – Projeção bidimensional obtida a partir da análise de variáveis canônicas e correspondente matriz de confusão alcançado pela melhor taxa de acerto. (a) CVA do vetor $[\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle]$ com a regra B12345-S0456 para a classificação da espécie *Tradescantia zebrina*. (b) CVA do vetor $\vec{\mu}_S$ usando a regra B014-S04567 para a espécie *Callisia repens*. c) CVA do vetor $[\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle]$ com a regra B57-S378 para a espécie *Ctenanthe oppenheimiana*. Na matriz de confusão, cada célula mostra o número de instâncias preditas corretamente.

Fonte: Elaborada pela autora.

6.4.5 Discussões

Nesta seção foi feita uma análise dos padrões de distribuição de estômatos. A construção das redes estomáticas foi baseada na matriz de distâncias entre pares de estômatos correspondentes a imagens microscópicas (Figura 40); então a conexão dos vértices da rede (estômatos) dependem da informação de distância entre e um limiar δ_{T_i} , com o qual a conectividade das redes cresce conforme o limiar aumenta, até tornar-se totalmente conectada. Esta estratégia parece ser muito útil na caracterização de redes estomáticas. Então, a matriz de distância para todos os pares de estômatos em imagens microscópicas providencia uma fonte rica de informação acerca da geometria do estômato

Tabela 20 – Comparação da taxa de acerto (%) para a classificação da plasticidade fenotípica em plantas de diferentes espécies usando medidas estruturais de rede. A última linha apresenta os resultados obtidos para LLNA usando a regra B12345-S04568 e $[\langle\mu_S\rangle, \langle\mu_W\rangle, \langle\mu_L\rangle]$ como vetor de características e classificador SVM.

Acerto (%)	<i>Tradescantia</i>	<i>Callisia</i>	<i>Ctenanthe</i>
$\langle k \rangle$	83 (± 3)	47 (± 10)	80 (± 3)
$\langle H_{k_2} \rangle$	83 (± 4)	47 (± 10)	78 (± 3)
$\langle H_{k_3} \rangle$	81 (± 5)	47 (± 10)	73 (± 4)
$\langle cc \rangle$	67 (± 8)	48 (± 13)	74 (± 3)
$\langle l \rangle$	39 (± 8)	36 (± 8)	72 (± 4)
ρ_P	13 (± 7)	27 (± 10)	72 (± 4)
$[\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \langle l \rangle, \rho_P]$	74 (± 8)	64 (± 7)	78 (± 3)
LLNA	90 (± 6)	95 (± 5)	95 (± 2)
Regra	(B12345-S04568)	(B014-S04567)	(B57-S378)

Fonte: Elaborada pela autora.

da planta, revelando correlações entre o ambiente para o qual a planta é submetida e esta geometria.

A seguir são apresentados dois enfoques, tanto da perspectiva biológica quanto do reconhecimento de padrões. Na primeira perspectiva, no que tange às espécies de plantas submetidas a estresse de luz, foi observado que a densidade de estômatos (Figura 40a-b) é proporcional à exposição de luz, ou seja, a maior intensidade de luz maior a densidade estomática. Além disso, visualmente, as classes de 4h e 24h são mais difíceis de distinguir a olho nú. De forma análoga, no que tange à espécie *Callisia repens*, as condições de estresse por injeção de CO₂ é também proporcional à densidade estomática, sendo que as classes de CO₂ ($\uparrow$) e CO₂ ($\downarrow$) são mais difíceis de serem distinguidas.

Na segunda perspectiva, em geral, o método proposto pode captar as características específicas das três espécies. Para esta aplicação, a taxa de acerto máxima obtida pelo LLNA supera as medidas estruturais de redes, tanto individualmente quanto combinados. Mais especificamente, no que tange à espécie *Tradescantia zebrina* nossos resultados mostraram uma alta taxa de acerto de 90 (± 6)%, enquanto que teve uma sutil dificuldade para discernir as classes natural e 4 horas, como mostrado pela análise canônica e a matriz de confusão (ver Figura 42). No que tange à espécie *Ctenanthe oppenheimiana*, também foi observada uma alta taxa de acerto, e apenas 1 amostra foi mal predita. No que tange à espécie *Callisia repens*, obteve-se uma alta separabilidade das classes no melhor dos casos.

Com respeito às regras do LLNA, foi observado que as 400 melhores regras selecionadas (Seção 6.4.3), usando apenas 2 imagens de *Tradescantia zebrina*, serviram para validar a performance da classificação das outras duas espécies, estas regras parecem satisfazer este contexto específico de redes estomáticas baseada na matriz de distância. Portanto, dados de distância são úteis para dimensionar o efeito do ambiente na rede de estômatos sobre a folha. Entretanto, como será descrito na Seção 6.5, no caso das redes

textuais, uma outra estratégia para a seleção de regras foi usada.

Finalmente, vale a pena reportar que o desenvolvimento de ferramentas computacionais para quantificação da plasticidade de folhas na adaptação ambiental, como as desenvolvidas neste trabalho, deve ser útil tanto em medidas de agricultura quanto em mudanças climáticas. (214) Nesse contexto, nossos resultados demonstraram a viabilidade do método proposto como ferramenta de reconhecimento de padrões.

6.5 Reconhecimento de autoria usando redes textuais

6.5.1 Contextualização e aquisição de dados

A atual produção massiva de dados trouxe inúmeros desafios às áreas de Mineração de dados, Processamento de Linguagem Natural, e Aprendizado de Máquina. Nesse contexto, o reconhecimento de autoria é um problema de interesse prático e técnico considerável que busca atribuir autoria a documentos anônimos ou sob controvérsia. Esta tarefa chamou atenção de pesquisadores principalmente por suas implicações em aplicações reais, como detecção de plágio (216,217), práticas forenses contra crimes cibernéticos (218) e acordos acerca de documentos sob disputa, (219), entre outros.

Muitos métodos têm sido propostos para solucionar o problema da tarefa de atribuição de autoria. (219) Por um lado, as técnicas tradicionais (e.g. modelo n -grams) usam métodos estatísticos baseados em contagem de palavras e medidas relacionadas (219–221), tem provado serem uma simples e efetiva solução em casos particulares, embora são propensos a ataques e manipulações nos dados. (221,222)

Por outro lado, a tarefa de atribuição de autoria tem sido abordada com sucesso utilizando o marco das redes complexas. (85,86,220,223–226) Em um sistema típico de reconhecimento de autoria baseado em rede, textos são modelados como uma rede e a estrutura dessas redes é então usada como propriedade relevante para discriminar diferentes autores. (85) As redes de adjacência de palavras (ou redes de co-ocorrência) (223) são um claro exemplo de modelagem de textos em redes, a qual vem sendo explorada por ter sido útil para extrair padrões estilísticos textuais. (85,227,228) De fato, este modelo foi explorado em inúmeros trabalhos (85,88,89,220), por conta de sua simplicidade e pouca dependência de recursos linguísticos.

Enquanto medidas estruturais de redes são úteis a fim de entender as propriedades topológicas principais de textos, elas podem trazer caracterizações ambíguas, principalmente quando as sutilezas do estilo literário não são mapeadas de forma adequada. Além disso, no que tange às medidas estruturais, não há consenso sobre o tipo de medidas que deva ser usada. (85) Por essa razão, a criação de medidas de redes que sejam informativas, eficientes e não ambíguas para modelos específicos permanecem como um problema aberto na ciência de redes. (85)

Dessa forma, nesta tese foi proposta a caracterização das redes textuais usando o LLNA. Mais especificamente, na Figura 43 é dada uma visão geral da metodologia a ser usada assim como da configuração das experiências. Primeiro, os textos são submetidos a uma estratégia de pré-processamento de texto, para em seguida serem representados por uma rede seguindo o modelo de co-ocorrência. Em seguida, partimos para a metodologia geral do LLNA, ou seja, inicialmente é feita a seleção de melhores regras para dar solução otimizada ao problema de atribuição de autoria. Uma vez obtidas as melhores regras, a validação é feita sob novos dados, os quais são treinados, a partir dos quais medições da dinâmica do LLNA são extraídas para caracterizar o estilo de cada autor.

Figura 43 – Esquema de atribuição de autoria baseada no método LLNA. Os seguintes passos são aplicados: (1) um texto escrito é pré-processado; (2) uma rede é gerada basando-se na extração de palavras-chave do pré-processamento; (3) uma regra selecionada do LLNA evolui sobre a topologia da rede textual (4) propriedades espaço-temporal do LLNA são extraídas e então usadas para a tarefa de atribuição de autoria.

Fonte: Adaptada de MACHICAO et al. (229)

A base de dados de redes textuais foram adquiridos pelo grupo de Computação Científica do IFSC em parceria com o grupo de pesquisa do Instituto de Ciências Matemáticas e de Computação (ICMC), a cargo do Prof. Diego R. Amancio e do doutorando Edilson A Corrêa Jr. As redes foram construídas a partir de corpus inglês de autores literários conhecidos. O corpus compreende 100 livros escritos por 20 diferentes autores que foram obtidos do repositório do *Project Gutenberg* (198), cada um possuindo 5 livros no seu repertório. Esta base foi dividida em duas partes, a primeira parte consiste em 12 autores destinados especificamente; a segunda parte consiste dos 8 autores restantes, listados a seguir:

- Conjunto de autores (classes) para a seleção da regra do LLNA: *Melville, Gray, Lang, Davis, James, Bower, Irving, Wells, Alger, Austen, Twain* e *Hawthorne*.
- Conjunto de autores (classes) para a validação do desempenho da classificação: *Doyle, Stoker, Darwin, Dickens, Hardy, Wodehouse, Poe* e *Munro*.

Esta separação é justificada pois neste estudo, considera-se a tarefa de discriminar entre os 8 autores distintos, dado que este mesmo conjunto de autores e gêneros foram consideradas em trabalhos relacionados (85, 220, 230), fato que facilitará sua comparação direta. Além disso, por meio desta separação, garante-se que conjuntos de dados diferentes

sejam empregados para a seleção da regra e para a aprendizagem e os processos de classificação. Também comparamos a abordagem proposta com as medidas estruturais da rede. (85) Finalmente, na última parte desta aplicação, exploramos os efeitos do processo de lematização nas propriedades das redes.

6.5.2 Modelagem de textos como redes

Nesta seção será apresentada brevemente como os textos são representados como redes seguindo o modelo de co-ocorrência, maiores detalhes podem ser encontrados em Machicao et al. (229) No modelo de rede de co-ocorrência, cada vértice representa uma palavra e os vértices são criados sempre que duas palavras aparecem adjacentes a outra no texto.

Antes da transformação do texto em rede, três passos de pré-processamento são seguidos: (i) o documento é dividido em unidades significativas (palavras-chave); (ii) remoção de palavras que contenham pouco sentido semântico, como artigos, preposições e pontuações; (iii) lematização, a qual é aplicada à fim de mapear as palavras restantes em suas respectivas formas canônicas. Dessa forma, verbos e substantivos são mapeados para sua forma infinitiva e singular, respectivamente. A lematização costuma requerer a identificação de cada parte de discurso para resolver possíveis ambiguidades. Neste trabalho foi usado o método *Average Perceptron part-of-speech Tagger* proposta por Collins (231), que corresponde à tarefa de rotulação de uma palavra em um texto com uma determinada classe gramatical (e.g., substantivo, verbo, adjetivo, advérbio, etc). Um exemplo dos passos de pré-processamento aplicados a um texto e sua correspondente construção de rede é ilustrada na Figura 44.

Figura 44 – Modelagem da rede de co-ocorrência dado o seguinte trecho “Complex networks model several properties of texts. A complex text displays a complex organization”. Neste exemplo foi considerado a lematização de todas as palavras para construir a rede.

Fonte: Adaptada de MACHICAO et al. (229)

Em nossos experimentos, também avaliamos três tipos de estratégias de lematização para gerar as redes textuais, o que levou à geração de três variações distintas de conjuntos de dados, a seguir:

- *Nenhuma*: o conjunto de dados original é mantido, ou seja, não é feita nenhuma lematização;

- *Parcial*: a lematização foi aplicada apenas em substantivos, ou seja, todos os substantivos são mapeados para suas formas singulares;
- *Completa*: A lematização é aplicada em todas as palavras. Portanto, verbos e substantivos são mapeados para suas formas infinitivas e singulares, respectivamente.

Observe que estas três variações também seguem a mesma metodologia detalhada antes para a separação das instâncias para a seleção da regra e para a validação do método.

6.5.3 Seleção da regra

Avaliamos, de forma exaustiva, cada uma das 262,144 regras da família *Life-Like* no conjunto de redes textuais destinadas a seleção da regra. A dinâmica de LLNA foi desenvolvida durante $t = 400$ iterações. Para caracterizar a dinâmica do LLNA, extraímos um vetor de características composto pela concatenação da entropia de Shannon e as distribuições de Lempel-Ziv $[\vec{\mu}_S, \vec{\mu}_L]$, que consistem em 60 descritores, ou seja, ambas as distribuições são janeladas em 30 compartimentos, as quais foram submetidas com o classificador kNN. Esta mesma configuração será seguida nos subseqüentes experimentos.

Figura 45 – Histograma da distribuição das 262,144 regras da família *Life-Like* em função da taxa de acerto atingida. Para a seleção de regra usando as bases de lematização *nenhuma*, *parcial* e *completa*. A concatenação da entropia de Shannon e complexidade de Lempel-Ziv $[\vec{\mu}_S, \vec{\mu}_L]$ foi considerada como vetor de características correspondentes e classificador de kNN.

Fonte: Elaborada pela autora.

A Figura 45 descreve a distribuição do histograma das taxas de acerto obtidas para o espaço de regras da família *Life-Like*. Nela observamos que a maioria das regras produziu classificadores com baixa taxa de acerto. Normalmente, foram encontrados valores inferiores a 40%. Por outro lado, há um pequeno número de regras que alcançaram precisão superior a 60%, correspondendo a aproximadamente 50 regras. No entanto, como a seleção da regra é feita através de um procedimento de otimização (50), portanto, neste estudo, consideramos uma quantidade maior de soluções, que incluíram também aquelas regras que obtiveram taxas de acerto superiores a 55%, que correspondem aproximadamente

a 400 regras. Esta estratégia é justificada porque encontramos melhores taxas de acerto quando testadas nos dados não vistos, como será mostrado mais adiante.

Observe que a seleção das melhores regras é realizada de forma independente em cada um dos três conjuntos de dados: *nenhuma*, *parcial* e *completa* do conjunto de autores destinados a seleção da regra. Além disso, como a seleção da regra é uma fase preliminar, deve-se esperar que entre o conjunto das melhores regras, as melhorias adicionais podem ser alcançadas usando outras medidas de LLNA. (50)

6.5.4 Resultados da validação

Para o problema de reconhecimento da autoria, aplicamos as melhores regras obtidas no passo anterior para identificar os estilos autorais no conjunto de dados de classificação que compreende os 8 autores. Primeiro, compilamos os três conjuntos de dados, *nenhuma*, *parcial* e *completa* usando diferentes medidas extraídas da dinâmica do LLNA: a distribuição de entropia de Shannon $\vec{\mu}_S$ e a distribuição Lempel-Ziv $\vec{\mu}_L$, de forma isolada e combinada $[\vec{\mu}_S, \vec{\mu}_L]$. Observe que a medida de comprimento da palavra, não foi considerada, pois em geral esta não trouxe ganhos à classificação, e sim quando apenas as outras duas foram combinadas.

A Tabela 21 apresenta os melhores resultados obtidos do conjunto de dados para a classificação. As colunas $\vec{\mu}_S$ e $\vec{\mu}_L$ mostram as taxas de acertos obtidas para cada vetor de características distintas. Os resultados ao combinar os descritores das médias $[\langle\mu_S\rangle, \langle\mu_L\rangle]$ são mostrados na última coluna da mesma tabela. Para os conjuntos com lematização *nenhuma*, *parcial* e *completa*, obteve-se a maior taxa de acerto de $69 (\pm 11)\%$, $71 (\pm 13)\%$ e $68 (\pm 17)\%$, usando as regras B124-S257, B2478-S25 e B3567-S03468, respectivamente. Nesses três casos, o vetor de características isolado ($\vec{\mu}_L$) foi que obteve a maior taxa de acerto. Observe que, todos estes experimentos foram avaliados com vários classificadores, e não somente com SVM como descrito nas demais aplicações mostradas neste capítulo. De fato, os melhores resultados foram obtidos com o kNN, pelo que adotamos esse classificador nos subsequentes experimentos.

Para ilustrar a discriminabilidade obtida com o método proposto, na Figura 46a mostramos uma projeção da análise canônica em duas dimensões, no qual o conjunto com lematização *parcial* foi analisado, usando a regra B2478-S25 e o vetor de características $\vec{\mu}_L$. Nesta imagem, observamos que existe uma clara separação entre *Darwin* e os outros autores. Um padrão semelhante ocorre para *Hardy* e *Dickens*, enquanto os outros variam consideravelmente seus estilos de livro para livro, como, por exemplo, *Stoker*. Na Figura 46b fornecemos a matriz de confusão obtida com a melhor classificação. Como esperado, *Darwin* e *Munro* são facilmente discerníveis dos outros autores, ao contrário do *Stoker*, já que três de seus livros foram incorretamente atribuídos a *Dickens*, *Hardy* e *Wodehouse*.

Adicionalmente, outro experimento foi realizado, desta vez considerando um pro-

Tabela 21 – Taxas de acerto (%) obtida usando diferentes medidas ($\vec{\mu}_S$ e $\vec{\mu}_L$) e suas combinações ($[\langle\mu_S\rangle, \langle\mu_L\rangle]$) como vetor de características para classificar 8 autores sob diferentes conjuntos de lematização *nenhuma*, *parcial* e *completa*. Foi usado o classificador kNN. A última linha apresenta o número de descritores de cada vetor de características.

Lematização	Regra	$\vec{\mu}_S$	$\vec{\mu}_L$	$[\langle\mu_S\rangle, \langle\mu_L\rangle]$
<i>Nenhuma</i>	B124-S257	51 (± 16)	69 (± 11)	66 (± 12)
	B1245-S1245	38 (± 15)	62 (± 16)	58 (± 17)
	B245-S457	61 (± 14)	58 (± 17)	48.5 (± 15)
<i>Parcial</i>	B2478-S25	44 (± 14)	71 (± 13)	50 (± 15)
	B026-S14	49 (± 15)	60 (± 15)	64 (± 15)
	B148-S6	63 (± 13)	36 (± 10)	54 (± 16)
<i>Completa</i>	B3567-S03468	40 (± 12)	68 (± 17)	48 (± 20)
	B13568-S13	64 (± 15)	52 (± 15)	35 (± 14)
	B0134568-S0123568	37 (± 17)	55 (± 15)	62 (± 14)
ND		30	30	60

Fonte: Elaborada pela autora

blema de classificação binária, ou seja, calcular a taxa de acerto para classificar os livros que correspondem a um autor A de quaisquer outros livros pertencentes a X autores diferentes. Isto foi analisado para cada um dos autores usando até três diferentes classificadores (kNN, Naive Bayes e SVM). Estes resultados são mostrados na Figura 46c, a qual mostra que *Darwin* é a classe mais separável dentre esse grupo de autores, seguido do *Munro*, e o pior dos casos *Stoker*. Além disso, no que tange aos resultados dos classificadores, observamos que esses resultados mostram que em média as taxas de acerto são independentes do classificador aplicado e que sua variabilidade é apenas sutil.

A Tabela 22 apresenta a comparação entre as taxas de acerto obtidas pelo LLNA (lado direito) e pelas medidas estruturais de redes (lado esquerdo). Cada medida foi calculada diretamente das redes textuais que compõem os três conjuntos de dados (*nenhuma*, *parcial* e *completa*). Note que o desempenho do método clássico baseado em medidas estruturais, em geral, é melhorado quando não utiliza lematização. O melhor resultado foi obtido com o classificador SVM 61 (± 16)%, que é semelhante aos melhores resultados relatados em. (85) Um desempenho semelhante também foi obtido com o classificador *Naive Bayes* 58 (± 15)%.

No lado direito da Tabela 22 são mostrados os resultados obtidos pelo LLNA. As regras B124-S257, B2478-S25, B3567-S03468 que forneceram as taxas de acerto mais altas para os conjuntos com lematização *nenhuma*, *parcial* e *completa* ao usar apenas a distribuição Lempel-Ziv $\vec{\mu}_L$ e classificador kNN.

Quando consideramos todos os conjuntos de dados de lematização em função dos classificadores, a taxa de acerto mais alta do LLNA foi de 71 (± 13)% (lematização *parcial* + kNN), enquanto as medidas estruturais obtiveram 61 (± 16)% (lematização *nenhuma* +

Figura 46 – (a) Análise dos componentes principais realizada para a tarefa de reconhecimento da autoria usando os cinco livros dos 8 autores usados para a classificação com lematização *parcial*. Foi usada a regra B2478-S25 e a distribuição da complexidade de Lempel-Ziv μ_L como vetor de características. (b) Matriz de confusão usando o método kNN alcançado pela melhor taxa de classificação. Cada célula mostra o número de instâncias preditas corretamente, onde os elementos diferentes de zero são indicados. (c) Comparação da precisão obtida pelo método proposto tratando a verificação da autoria como um problema de classificação binária. A taxa de acerto foi calculada como a média e o desvio padrão para a classificação de cinco livros de um autor A contra cinco livros de autores desconhecidos X , usando três classificadores diferentes (kNN, Naive Bayes e SVM).

Fonte: Elaborada pela autora.

SVM), o que corresponderia a um empate técnico de ambos os métodos.

Porém, ao comparar os resultados por conjuntos de dados por lematização, o desempenho obtido com os descritores de LLNA para o classificador kNN supera as medidas estruturais em $17 (\pm 13)\%$, $22 (\pm 14)\%$ e $24 (\pm 33)\%$ para lematização *nenhuma*, *parcial* e *completa*, respectivamente.

6.5.5 Discussões

Nesta seção foi abordado o problema de atribuição de autoria usando o LLNA. Para avaliarmos o método foram adquiridas redes textuais correspondentes a autores literários do século XX usados como *benchmark* da área. De acordo com os resultados, o LLNA obteve um ganho sutil quando comparado às medidas estruturais de redes. De fato, a melhor taxa de acerto encontrada usando a melhor configuração de parâmetros mostra inequivocamente

Tabela 22 – Comparação da taxa de acerto (%) para a identificação de autoria usando a concatenação de medidas estruturais de rede $[\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \langle l \rangle, \rho_P]$ e os resultados obtidos com o LLNA usando a combinação de melhor regra e vetor de características $\vec{\mu}_L$.

Medidas estruturais de rede				Método proposto (LLNA)		
	<i>Nenhuma</i>	<i>Parcial</i>	<i>Completa</i>	<i>Nenhuma</i> (B124-S257)	<i>Parcial</i> (B2478-S25)	<i>Completa</i> (B3567-S03468)
NB	58 (±15)	56 (±15)	51 (±15)	49 (±13)	55 (±13)	48 (±16)
kNN	52 (±15)	49 (±15)	44 (±16)	69 (±11)	71 (±13)	68 (±17)
SVM	61 (±16)	49 (±14)	50 (±15)	44 (±19)	56 (±13)	51 (±20)

Fonte: Elaborada pela autora

que os descritores extraídos pelo LLNA podem capturar as particularidades dos estilos autorais, permitindo assim a discriminação dos autores em textos desconhecidos.

Apesar da taxa de acerto obtida, esta ainda é relativamente pobre quando comparada aos métodos estatísticos tradicionais. Vale lembrar que o problema de atribuição de autoria não é uma tarefa trivial, isto porque dos diversos métodos existentes, os métodos estatísticos clássicos têm provado serem efetivos, porém ainda limitados (221, 222), razão pela qual os métodos baseados em extração de características a partir de representação de redes têm surgido como alternativas complementares aos métodos tradicionais. Como mostrado em alguns estudos (89, 224), as características adquiridas por modelos de redes são distintas das tradicionais.

Assim, nesta tese, o LLNA é apresentado não para superar aos métodos tradicionais, e sim de ser uma ferramenta auxiliar que traz informação complementar, que não poderia ter sido obtida apenas das propriedades estatísticas dos textos e sim quando levadas num contexto de redes e quando a dinâmica de autômatos celulares é explorada nelas. Principalmente, quando métodos estatísticos tradicionais são combinados com aqueles baseados em propriedades de rede, dessa forma, é possível obter um melhoramento dos resultados. Note que medidas baseadas em rede providenciam uma visão complementar de documentos não estruturados, uma propriedade que pode ser explorada mais a fundo em abordagens híbridas.

Finalmente, no que tange à lematização, tipicamente, em técnicas baseadas em redes, este pré-processamento é importante para agrupar diferentes formas de escrita no mesmo vértice. Na tarefa de atribuição da autoria, a lematização pode levar a uma grande perda de informação, dificultando a identificação precisa das escolhas de escrita específicas dos autores. (219) Do mesmo modo, em tarefas semânticas, como a desambiguação do sentido das palavras, a lematização das palavras desempenha um papel importante na performance. (232) Embora Toman et al. (233) argumente que este esse pré-processamento não afeta a performance geral dos sistemas de classificação de texto. No entanto, uma contribuição desta tese, foi providenciar uma análise sistemática do efeito de lematização

em métodos de reconhecimento de autoria baseados em redes. Peculiarmente, os melhores resultados foram alcançados com processo de lematização *parcial*, sugerindo que este procedimento é mais adequado que simplesmente fazer uma lematização em todas as palavras quando redes textuais são usadas como modelo subjacente para esta tarefa, ou nenhuma.

6.6 Classificação de sequências pseudo-aleatórias em redes espaço-tempo

6.6.1 Contextualização

Apesar de nos últimos anos o conceito de redes ter sido aplicado em diferentes problemas, não foi encontrado na literatura, até agora, trabalhos que investigam o uso de redes na análise de pseudo-aleatoriedade de PRNGs. Esta abordagem representa uma linha de pesquisa e paradigma com novos desafios e limitações, sobretudo porque é intuitivo pensar que padrões e aleatoriedade são termos opostos. Observe que a qualidade pseudo-aleatória de um PRNG não poderia ser avaliada se antes não existisse um padrão ouro com o qual classificá-la. Uma vez que foi desenvolvido o k -mapa logístico como um PRNG parametrizado por k (ver Capítulo 4), foi aberta a possibilidade de explorar tais sequências num contexto de reconhecimento de padrões.

Assim, no que tange à teoria de redes, esta abordagem traz consigo um marco de trabalho para comparar a estrutura topológica das redes que modelem um processo pseudo-aleatório. Este tipo de análise é fundamental para inferir, por exemplo, a qualidade/quantidade de informação (padrões) nas métricas que captam características que vão além dos testes estatísticos de pseudo-aleatoriedade (e.x.: DIEHARD (181) e NIST (171)). A modelagem de dados advindos de processos aleatórios como redes complexas ainda é uma linha de pesquisa que apresenta grande potencial e diversos desafios. Estes processos aleatórios, entenda-se como geradores de números pseudo-aleatórios, também podem ser levados no contexto de séries temporais.

Na literatura, a modelagem destas sequências em redes vem começando a ser mais exploradas como, por exemplo, as redes de recorrência (15), as quais são usadas para representar séries temporais geradas por sistemas não lineares. Uma outra abordagem é a transformação das sequências de dígitos de números irracionais e transcendentais em uma rede (d, r) , na qual cada d dígitos corresponde a um vértice e sua conectividade é determinada pela adjacência dependendo de um raio r , (90, 91) porém, o estudo de séries temporais no contexto de reconhecimento de padrões ainda é pouco explorada, ou seja, poder classificar sequências advindas de um PRNG em específico. Assim, foram rotuladas as sequências geradas por k_0, k_1, k_2, k_3, k_4 para tentar separar as sequências geradas por cada gerador específico.

É importante salientar que, quando as sequências advêm de um sistema (gerador) aleatório, é esperado obter taxas de acerto próximas de 50%, uma vez que não é possível

extrair nenhum tipo de informação do sistema. Por outro lado, taxas superiores a 50% sugerem que as sequência apresentam padrões, ou seja, segue a seguinte relação:

se a taxa de acerto $\gg 50\%$ então a qualidade do PRNG é fraca, e
se a taxa de acerto $\approx 50\%$ então a qualidade do PRNG é forte.

Portanto, no contexto de PRNGs, quanto maior a taxa de acerto, o método de extração de características pode ser utilizado como um índice da qualidade da pseudo-aleatoriedade.

Aqui é explorado o LLNA como uma ferramenta para classificar sequências de números pseudo-aleatórios, mais concretamente para analisar a pseudo-aleatoriedade do PRNG baseado no k -mapa logístico. Com base neste PRNG, várias sequências pseudo-aleatórias podem ser geradas a partir do parâmetro k . Como foi descrito no Capítulo 4, existem padrões mais evidentes para valores pequenos de k , enquanto vai tornando-se menos trivial para maiores valores de k . Já que a pseudo-aleatoriedade vai incrementando a cada acréscimo de k , justificamos poder classificar pares de geradores (classificação binária), por exemplo, k_0 vs k_1 , k_1 vs k_2 , e assim por diante.

6.6.2 Modelagem de séries temporais como redes espaço-tempo limiarizada

A etapa de modelagem é um passo importante para a obtenção de uma boa representação dos dados evitando perda de informação. Aqui é apresentado um modelo de redes, considerando que o espaço-tempo de uma órbitas gerada por um sistema dinâmico é a tesselação da rede. Esse modelo, seguindo os critérios de raio limiar, pode ser transformado em várias redes. Observe que, traçando um paralelo com as redes estomáticas dadas na Seção 6.4, a rede espaço-tempo também segue a estratégia proposta por Backes et al (17) para análise de contornos.

A Figura 47 exemplifica os passos para a construção do modelo de rede espaço-tempo limiarizada[‡]. Dado um diagrama espaço-tempo composto por uma órbita $\mathcal{O}^k(\mu, x_0) = \{x_0^k, x_1^k, \dots, x_t^k\}$ gerada pelo k -mapa logístico (Equação (4.2)) com condição inicial x_0 , como ilustrado na Figura 47a. Para construir uma rede associada a esta sequência, dizemos que cada ponto x_0^k representa um vértice, e a adjacência com os pontos vizinhos representam as arestas, em que o número de vértices (N) é dado pelo número de iterações que o sistema evolui.

A partir de um diagrama espaço-tempo várias redes podem ser extraídas usando um parâmetro de corte (limiar) T_l . Inicialmente, é obtida uma rede de pesos totalmente conectada, contendo as distâncias euclidianas entre cada par de vértices (W). Esta matriz é então transformada em várias redes representadas pela matriz de adjacência A_{T_l} dependendo

[‡] O modelo de rede espaço-tempo foi proposto na qualificação da mestranda Marcela Lopes Alves pertencente ao grupo de Computação Científica do Instituto de Física de São Carlos (IFSC) a cargo do Prof. Odemir M. Bruno.

Figura 47 – (a) Evolução temporal de uma órbita gerada pelo mapa logístico original com condição inicial aleatória, para os parâmetro $\mu = 3.999$ evoluída durante 150 iterações e desconsideradas as primeiras 100 (transiente). Os números ao lado de cada vértice representam a seqüência temporal na qual cada ponto foi gerado. O círculo representa os possíveis vizinhos adjacentes ao ponto central. (b) Construção da rede espaço-tempo limiarizada, de esquerda para direita, por $T_l = \{0.08, 0.1, 0.12\}$ para a mesma órbita gerada pelo k -mapa logístico com k_0 , k_1 e k_2 (de cima para baixo) correspondente à órbita original em a). Foram consideradas $t = 150$ iterações, pelo que cada rede contém $N = 150$ vértices. O tempo total de evolução foi normalizado entre $[0, 1]$ usando a escala de $\Delta t = 0.01$.

Fonte: Elaborada pela autora.

do limiar T_l , dado pela Equação (6.2) (usada também na Seção 6.4), ou seja se a distância entre pares de pontos for menor ou igual a esse valor, então uma aresta é estabelecida entre eles.

$$A_{T_l} = \delta_{T_l}(W) = \forall w \in W \begin{cases} a_{ij} = 0 & \text{se } w_{ij} > T_l \\ a_{ij} = 1 & \text{se } w_{ij} \leq T_l \end{cases} \quad (6.2)$$

Observe que, o limiar T_l define o raio de conectividade de um ponto com os vizinhos mais próximos a este. Por exemplo, se T_l for pequeno, o número de arestas também será pequeno e a rede será composta de muitos componentes conectados sem interseção. Caso contrário, se T_l for grande, a rede se tornará completamente conetada. Para valores intermediários de T_l , outras propriedades podem começar a aparecer, ou vice-versa. Note também que, já que o tempo é discreto, então este deve ser escalado usando uma constante Δt , de tal forma que o limiar T_l seja proporcional ao momento de calcular a distância

euclidiana entre os pontos da órbita. Na Figura 47 pode ser observado a construção de várias redes limiarizadas geradas pelo PRNG baseado k -mapa logístico com $\mu = 3.9999$ (ver Seção 4.4).

6.6.3 Seleção da regra

Foram construídas duas bases de redes diferentes para a seleção de regras e validação da classificação. Em todos os experimentos foi adotado o PRNG baseado no k -mapa logístico com condições iniciais aleatórias e parâmetro de controle $\mu = 3.9999$. Para o primeiro conjunto foram geradas 50 redes para cada uma das classes k_0 , k_1 e k_2 as quais foram usadas para encontrar as melhores regras da família *Life-Like*. Cada uma destas redes foi gerada usando $N = 100$ vértices, ou seja o k -mapa logístico foi iterado por 100 iterações; e o segundo conjunto consiste de 100 redes para as classes $k_0, k_1, \dots, k_4$, sendo em total 500 redes para o processo de validação da classificação. Cada uma destas redes foi gerada usando $N = 500$ vértices.

Cada uma das 262,144 regras da família *Life-Like* foram avaliadas exaustivamente no conjunto de redes destinadas à seleção da regra. As redes deste conjunto de dados foram obtidas apenas usando o limiar $T_l = 0.12$. Para caracterizar a dinâmica do LLNA, foi extraído um vetor de características para cada rede composto pela distribuição da entropia de Shannon $\vec{\mu}_S$ que consistem em 20 descritores, as quais foram submetidas com o classificador kNN.

A distribuição do histograma das taxas de acerto obtidas para o espaço de regras da família *Life-Like* é descrita na Figura 48. Nela foram encontradas mais de 3000 regras com taxas de acertos superiores a 50%, pelo qual escolhemos um subconjunto menor deste para o estágio de classificação que será apresentada na seção seguinte.

6.6.4 Resultados da validação

Para a classificação de redes espaço-tempo seguiu-se a estratégia de concatenação dos descritores obtidos das redes transformadas por cada valor de δ_{T_l} , proposta por Backes et al. (17) Aqui foram considerados 5 limiares, começando por $T_{ini} = 0.08$ até $T_Q = 0.16$, com incrementos de $T_{inc} = 0.02$. Além disso, foram extraídos os vetores de características $\vec{\mu}_S$, $\vec{\mu}_W$ e $\vec{\mu}_L$, os quais contem a concatenação dos 5 limiares, ou seja, cada uma contendo 20×5 descritores, respectivamente. Também foi avaliada a combinação destes, $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$, contendo 60×5 descritores, assim como as médias destes descritores $\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle$, contendo 3×5 descritores.

A Tabela 23 exhibe os resultados da 3 melhores regras para a classificação das redes do *dataset-PRNGs* usando o classificador kNN para cada vetor de características. A maior taxa de acerto obtida para o par de classes k_0 vs k_1 foi de $96.38 \pm 0.57\%$ usando a regra B0234678-S03457 e $\vec{\mu}_L$ como vetor de características. Já para o par de classes k_1

Figura 48 – Histograma da distribuição das 262,144 regras da família *Life-Like* em função da taxa de acerto atingida para a seleção de regra usando a base de redes espaço-tempo do PRNG *k*-mapa logístico, limiarizadas por $T_l = 0.12$. A entropia de Shannon ($\vec{\mu}_S$) foi considerada como vetor de características correspondentes e classificador de kNN.

Fonte: Elaborada pela autora.

vs k_2 a maior taxa de acerto foi de $62.71 \pm 1.62\%$ usando a regra B046-S3468 e $\vec{\mu}_S$ como vetor de características. Para os pares k_2 vs k_3 e k_3 vs k_4 a maior taxa de acerto foi de $54.45 (\pm 1.65)\%$ e $54.64 (\pm 1.59)\%$ respectivamente, e usando as regras B046-S3468 e B03578-S01468 com seus respectivos vetores de características $\vec{\mu}_S$ e $[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$. Nesses quatro casos, como esperado, observa-se a queda da taxa de acerto, correspondente à mesma dificuldade para separar estas classes.

Tabela 23 – Taxa de acerto (%) da classificação binária de sequências geradas pelo PRNG *k*-mapa logístico usando o LLNA com diferentes combinações de medidas como vetor de características classificadas por kNN. A última linha apresenta o número de descritores de cada vetor de características.

Classes	Regra	$\vec{\mu}_S$	$\vec{\mu}_W$	$\vec{\mu}_L$	$[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$	$[\langle \mu_S \rangle, \langle \mu_W \rangle, \langle \mu_L \rangle]$
k_0 vs k_1	B0234678-S03457	83.58 (± 1.17)	85.52 (± 1.25)	96.38 (± 0.57)	94.06 (± 0.74)	92.21 (± 1.16)
	B03578-S01468	79.35 (± 1.34)	65.37 (± 1.31)	94.82 (± 0.75)	84.66 (± 1.05)	88.94 (± 0.99)
	B046-S3468	95.55 (± 0.83)	70.63 (± 1.34)	93.62 (± 0.76)	91.55 (± 0.91)	94.97 (± 0.83)
k_1 vs k_2	B0234678-S03457	60.47 (± 1.67)	56.6 (± 1.3)	62.7 (± 1.49)	65.21 (± 1.75)	53.44 (± 1.7)
	B03578-S01468	58.95 (± 2.02)	46.85 (± 2.17)	62.44 (± 1.67)	55.22 (± 1.8)	60.5 (± 1.28)
	B046-S3468	60.24 (± 1.69)	55.7 (± 1.64)	62.71 (± 1.62)	58.02 (± 1.77)	60.64 (± 1.64)
k_2 vs k_3	B0234678-S03457	49.44 (± 1.73)	40.35 (± 2.02)	47.66 (± 1.55)	50.93 (± 1.98)	47.99 (± 1.58)
	B03578-S01468	47.03 (± 1.58)	52.86 (± 1.77)	52.19 (± 2.01)	48.58 (± 2)	52.26 (± 1.73)
	B046-S3468	54.45 (± 1.65)	49.23 (± 1.68)	51.78 (± 1.85)	46.49 (± 1.87)	48.61 (± 1.5)
k_3 vs k_4	B0234678-S03457	51.25 (± 2)	55.26 (± 1.98)	53.73 (± 1.57)	53.74 (± 1.52)	50.65 (± 1.53)
	B03578-S01468	47.89 (± 1.95)	50.84 (± 1.95)	50.96 (± 2.22)	54.64 (± 1.59)	41.46 (± 1.69)
	B046-S3468	48.16 (± 1.9)	48.51 (± 1.84)	51.99 (± 1.5)	52.25 (± 1.79)	45.89 (± 1.71)
	ND	100	100	100	300	15

Fonte: Elaborada pela autora.

O lado esquerdo da Figura 49 apresenta o vetor de características correspondente à distribuição da complexidade de Lempel-Ziv ($\vec{\mu}_L$), usando a regra B0234678-S03457, representada como um gráfico de curvas. No lado direito da Figura 49 é apresentada a

projeção bidimensional obtida pela análise de t-SNE para o vetor características que obteve a maior taxa de acerto, para cada um dos pares de classes. Nestas figuras, de cima para baixo, fica evidente que existe uma separação clara apenas para a classificação de k_0 vs k_1 , porém não com maior sucesso para os demais pares de classes, pois a taxa de acerto vai decrescendo, como esperado. Assim o descritor da complexidade de Lempel-Ziv $\vec{\mu}_L$ consegue separar as classes k_0 , k_1 e k_2 das demais.

Figura 49 – (Distribuição da complexidade de Lempel-Ziv $\vec{\mu}_L$ (coluna à esquerda) usando a regra B0234678-S03457 para as quatro combinações de pares de classes do PRNG k -mapa logístico (a) k_0 vs k_1 , (b) k_1 vs k_2 , (c) k_2 vs k_3 , e (d) k_3 vs k_4). Projeção bidimensional usando a técnica de visualização t-SNE correspondente ao vetor de características, respectivamente (coluna à direita).

Fonte: Elaborada pela autora.

A Tabela 24 apresenta a comparação entre as taxas de acerto obtidas pelo LLNA e pelas medidas estruturais de redes. Como pode ser observado, ao classificar estes mesmos pares de classes usando medidas estruturais foi obtido performance similares ao do LLNA, pois estas não conseguem separá-las além de k_2 vs k_3 . Em geral, podemos dizer que ambos métodos obtiveram taxas de acerto bastante similares, com exceção do par k_1 vs k_2 , no

qual a combinação de medidas estruturais obteve maior taxa de acerto que o LLNA, com uma margem de 14%.

Tabela 24 – Comparação da taxa de acerto (%) para a classificação das redes de espaço-tempo do PRNG k -mapa logístico usando medidas estruturais de rede como vetor de características, usando classificador kNN. A última linha apresenta os resultados obtidos com o LLNA.

Descritor	k_0 vs k_1	k_1 vs k_2	k_2 vs k_3	k_3 vs k_4
$\langle k \rangle$	78.14 (± 1.21)	68.46 (± 1.52)	48.65 (± 1.8)	47.28 (± 1.91)
$\langle H_{k_2} \rangle$	82.96 (± 1.14)	69.09 (± 1.24)	52.73 (± 1.81)	45.45 (± 1.89)
$\langle H_{k_3} \rangle$	79.72 (± 1.17)	65.23 (± 1.33)	50.59 (± 1.68)	50.27 (± 1.56)
$\langle cc \rangle$	66.95 (± 1.23)	56.46 (± 1.39)	51.72 (± 1.76)	51.35 (± 1.63)
ρ_P	95.61 (± 0.62)	55.34 (± 1.67)	51.95 (± 2.23)	55.3 (± 1.86)
$[\langle k \rangle, \langle H_{k_2} \rangle, \langle H_{k_3} \rangle, \langle cc \rangle, \rho_P]$	97.29 (± 0.62)	76.57 (± 1.38)	51.9 (± 1.82)	51.9 (± 1.82)
LLNA	96.38 (± 0.57)	62.71 (± 1.62)	54.45 (± 1.65)	54.64 (± 1.59)
Descritor	$\vec{\mu}_L$	$\vec{\mu}_L$	$\vec{\mu}_S$	$[\vec{\mu}_S, \vec{\mu}_W, \vec{\mu}_L]$
Regra	(B0234678-S03457)	(B046-S3468)	(B046-S3468)	(B03578-S01468)

Fonte: Elaborada pela autora.

6.6.5 Discussões

Nesta seção foram desenvolvidos mecanismos para avaliar a pseudo-aleatoriedade de um PRNG no contexto do reconhecimento de padrões. Assim, primeiro os dados de sequências pseudo-aleatórias foram modelados em redes, seguindo o critério de vizinhança de acordo com uma transformação δ_{T_i} . Em seguida, o LLNA foi usado para extrair características destas redes. Com base no modelo de redes espaço-tempo, o LLNA permitiu distinguir apenas até o PRNG k_2 , o que não teria maior significância, pois, o PRNG k -mapa logístico consegue passar todos os testes de aleatoriedade estatísticos quando $k \geq 4$, como reportado na Tabela 3 do Capítulo 4.

Esta queda da taxa de acerto pode se dever a vários fatores: (i) com respeito aos valores dos limiares δ_{T_i} ainda precisam ser explorados a fim de conseguir relacioná-los com as taxas de acerto, por exemplo usar diferentes combinações; (ii) o tamanho das redes apenas consideraram $N = 500$ pontos das órbitas, o qual significaria que não foi o tamanho necessário para realizar uma boa caracterização das redes; (iii) o modelo de rede espaço-tempo em si, pode ainda precisar de novas condições de vizinhança (adjacência), as quais também devem ser melhor exploradas. Assim, sendo advertidos destes fatores, uma solução alternativa ,advinda tipicamente do processamento de linguagem natural, foi proposta e é detalhada na Seção 6.7.

6.7 Classificação de sequências pseudo-aleatórias em redes de transição

6.7.1 Contextualização

Uma forma de buscar compreender as sequências de números aleatórios (séries temporais) seria considerar que a sequência em que os números ocorrem possuem uma propriedade markoviana. Esta propriedade refere-se à perda de memória de um processo

estocástico. Já que o mapa logístico trata-se de uma função da forma $x_{x+1} = f(x_t)$, então em princípio, o estado futuro x_{t+1} guardaria certa relação com o estado presente x_t . Baixo este princípio, é possível analisarmos os padrões de transições (rede de transição) e tentar classificar a qualidade pseudo-aleatória do padrão ouro k -mapa logístico.

6.7.2 Modelagem de redes de transição

A técnica de *n-grams*, tipicamente usada no processamento de linguagem natural (219) é uma forma interessante de analisar as sequências de palavras (letras, dígitos). Assim, o espaço de fase do k -mapa logístico $[0, 1]$ pode ser dividido em S compartimentos (estados). Por exemplo, dada a órbita $0.5289, 0.98249, 0.02394, 0.79242, \dots$ dividida em $S = 10$, seria transformada nas sequências de *n-grams* $5, 9, 0, 7, \dots$ para $n = 1$, ou em $59, 07, \dots$ para $n = 2$, respectivamente.

Uma matriz estocástica descreve uma cadeia de Markov em um espaço de estados finito S , em que a probabilidade de mover-se desde i para j em um passo é dado por $\Pr(j|i) = P_{i,j}$, então a matriz estocástica correspondente é dada por:

$$P = \begin{pmatrix} P_{0,0} & P_{0,1} & \dots & P_{0,j} & \dots & P_{0,S-1} \\ P_{1,0} & P_{1,1} & \dots & P_{1,j} & \dots & P_{1,S-1} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ P_{i,0} & P_{i,1} & \dots & P_{i,j} & \dots & P_{i,S-1} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ P_{S-1,0} & P_{S-1,1} & \dots & P_{S-1,j} & \dots & P_{S-1,S-1} \end{pmatrix}. \quad (6.3)$$

Desta matriz, se sabe que o total de probabilidade de transição de um estado i para todos os outros estados deve ser 1, ou seja, cumpre-se que

$$\sum_{j=1}^S P_{i,j} = 1. \quad (6.4)$$

Na Figura 50a é dado um exemplo da construção da rede de transição criada a partir de uma sequência de números (vértices) gerados pelo PRNG k -mapa logístico, assim é possível estabelecer relações de adjacência, no qual pares de *n-grams*, sem sobreposição, conetam-se entre si (arestas)

Neste exemplo, é obtida uma rede de transição usando 1-gram. Por exemplo, 5 coneta-se com 9, em seguida, $9 \rightarrow 0$, logo, $0 \rightarrow 2$, e assim por diante. Observe que a rede possui pesos, que correspondem ao número de vezes em que os pares de *n-grams*, sem sobreposição, aconteceram. Esta rede de transição é associada a uma matriz de transição da cadeia de Markov. Apenas para ilustração, foi considerado uma sequência contendo 10^4 números, desta é possível obter uma matriz de probabilidades (Figura 50b), a mesma que foi ilustrada numa escala de cores (Figura 50c).

Figura 50 – (a) Construção da rede de transição do PRNG k -mapa logístico, em que cada número (círculo) da seqüência representa um vértice na rede e as arestas representam o número de vezes em que esse par de vértices aconteceu na seqüência. (b) Ilustração de uma matriz de probabilidades de transição de tamanho $S \times S$, com $S = 10$, no qual as linhas e colunas correspondem a todas os pares de estados preenchidos pelas frequências (normalizadas) com que aparecem na seqüência. (c) Ilustração colorida da matriz correspondente a (b).

Fonte: Elaborada pela autora.

Para os subsequentes experimentos foram geradas 5 bases de seqüências pseudo-aleatórias. Cada uma das bases contendo 100 seqüências aleatórias (amostras) de 10^2 , 2.5×10^3 , 10^4 , 2.5×10^5 , e 10^6 pontos gerados com condições iniciais aleatórias pelo PRNG $k_0, k_1, \dots, k_{10}$ -mapa logístico. Na Figura 51a é ilustrada a matriz de probabilidades de transição usando 10^6 números gerados pelo k -mapa logístico representada por cubos. Desta figura, podemos observar que o padrão inicial de k_0 condiz com o padrão U invariante do mapa logístico, e conforme k aumenta tal padrão vai se espalhando até chegar numa distribuição mais uniforme (Figura 51b-c).

A matriz de probabilidades de transição foi utilizada diretamente como vetor de características $\vec{\mathcal{F}}$, ou seja, considerando que trata-se de uma matriz de $S \times S$, com $S = 10$, possui, portanto, 100 descritores. Para isso, a matriz da Equação (6.3) é convertida em vetor da seguinte forma

$$\vec{\mathcal{F}} = \{P_{0,0}, \dots, P_{0,S-1}, \dots, P_{i,0}, \dots, P_{i,S-1}, P_{S-1,0}, \dots, P_{S-1,S-1}\}. \quad (6.5)$$

No lado esquerdo da Figura 52 apresenta o vetor de características correspondente à matriz de probabilidades de transição descrito na Equação (6.5), representada como um gráfico de curvas para os PRNGs $k_0, k_1, \dots, k_6$. No lado direito da Figura 52 é apresentada a projeção bidimensional obtida pela análise de t-SNE para o mesmo vetor de características. Nesta figura fica evidente que existe uma separação clara para todas as classes até k_6 . Observe que, a partir de $k > 6$ esta separação já não é mais possível (não mostrado).

Figura 51 – (a) Matriz de probabilidades de transição correspondente a uma amostra do PRNG k_0 , apresentando o padrão característico em forma de U. (b) Matriz de probabilidades de transição representada por cubos para $k_0, k_1 \dots k_9$, da esquerda para a direita e de cima para baixo. A terceira dimensão representa cada uma das 100 amostras geradas pelo PRNG k -mapa logístico. (c) Gráfico de barras da matriz de probabilidades de transição calculada a partir da média das 100 amostras e respectivo desvio padrão para k_0, k_1 e k_9 .

Fonte: Elaborada pela autora.

Figura 52 – (a) Distribuição do vector de características correspondente à matriz de probabilidade de transição para os PRNGs $k_0, k_1, \dots, k_6$ e (b) projeção bidimensional usando a técnica de visualização t-SNE correspondente a a).

Fonte: Elaborada pela autora.

6.7.3 Resultados da validação

Para a avaliação do desempenho do descritor proposto, aqui foi utilizada a estratégia de 10-validação cruzada (96) e o classificador kNN com $k = 1$. Os cinco conjuntos de dados foram avaliados, os resultados são apresentados na Figura 53. Nela, cada matriz apresenta

a taxa de acerto média para a classificação binária entre classes k_i vs k_{i+1} . Note que o desvio padrão de cada taxa de acerto é bastante pequena na escala de 0.1%, pelo que não foi mostrado nas tabelas para melhor visualização.

A correta interpretação destes resultados é guiada pela barra de cores, a qual indica os maiores e menores taxas de acerto. No primeiro conjunto de dados (Figura 53a) foi utilizado sequências contendo 100 amostras de 1-grams, com o qual apenas foi possível distinguir entre k_0 e k_1 com taxas de acerto superiores de 90%, enquanto que o resto das classes obteve taxas próximas de 50%. Entretanto, conforme o tamanho da sequência aumenta, estas taxas de acerto também vão incrementando-se, por exemplo, na Figura 53b usando 2500 n -grams já é possível distinguir até k_3 vs k_4 com 74.06%. Surpreendentemente, quando foi explorado sequências de 10^6 , consegue separar as classes de k_7 vs k_8 com taxa de acerto de 78.56%.

6.7.4 Discussões

Nesta seção foi proposto o uso de técnicas advindas do processamento de linguagem natural, mais especificamente, considerando as sequências de números pseudo-aleatórias do PRNG k -mapa logístico como cadeias de Markov. Considerando que os sistemas dinâmicos uni-dimensionais são da forma $x_{t+1} = f(x_t)$, logo, esta premissa parece encaixar-se na tarefa de distinção entre as classes k . Pelos resultados obtidos usando a matriz de probabilidades de transição obtivemos altas taxas de acerto, inclusive até separar k_7 das demais usando apenas sequências contendo 10^6 números. Além disso, se pode perceber que, conforme o tamanho das sequências com as quais são treinadas, sobem as taxas de acerto, conseguindo separar ainda mais classes. Com isto, surge uma dúvida: é possível separar mais classes dependentes do parâmetro k ?

6.8 Resumo dos resultados e discussões

Neste capítulo foi demonstrado o grande alcance multidisciplinar no qual o método proposto pode ser aplicado. Fazendo uma análise geral dos resultados obtidos, o LLNA conseguiu melhorar as taxa de acerto com relação das medidas estruturais de redes, em quase todos os problemas aqui apresentados, como é evidenciado pela Figura 54 que apresenta um resumo com todas as comparações.

Especificamente, foi advertido que nos problemas de identificação de organismos vivos, o LLNA alcançou os melhores desempenhos, da mesma forma que no contexto da análise de plasticidade fenotípica de plantas por meio das redes estomáticas, em quase todas as espécies que foram submetidas às condições de estresse ambiental. Também destacou-se no contexto de identificação de autoria usando redes textuais, considerado um problema de grande desafio na área de Processamento de Linguagem Natural.

(a) # dígitos= 10^2

	k ₁	k ₂	k ₃	k ₄	k ₅	k ₆	k ₇	k ₈	k ₉	k ₁₀
k ₀	100	100	100	100	100	100	100	100	100	100
k ₁		93.79	93.26	95.25	96.1	94.73	96.34	95.39	95.12	96.01
k ₂			59.53	62.35	62.99	64.01	69.67	64.51	67.12	64.14
k ₃				52.18	56.75	48.55	54.16	48.64	54.67	56.2
k ₄					49.7	48.83	49.17	51.23	50.36	50.04
k ₅						51.01	54.76	46.11	45.09	48.28
k ₆							50.99	46.93	42.64	47.77
k ₇								48.63	48.48	47.26
k ₈									51.13	50.12
k ₉										48.33

(b) # dígitos= 2.5×10^3

	k ₁	k ₂	k ₃	k ₄	k ₅	k ₆	k ₇	k ₈	k ₉	k ₁₀
k ₀	100	100	100	100	100	100	100	100	100	100
k ₁		100	100	100	100	100	100	100	100	100
k ₂			100	100	100	100	100	100	100	100
k ₃				74.06	80.64	85.7	88.21	88.83	90.17	90.83
k ₄					49.51	55.33	51.52	58.19	58.03	60.7
k ₅						50.64	51.31	50.91	52.91	53.9
k ₆							46.58	51.53	55.09	46.32
k ₇								46.15	44.24	46.63
k ₈									56.12	45.16
k ₉										45.7

(c) # dígitos= 10^4

	k ₁	k ₂	k ₃	k ₄	k ₅	k ₆	k ₇	k ₈	k ₉	k ₁₀
k ₀	100	100	100	100	100	100	100	100	100	100
k ₁		100	100	100	100	100	100	100	100	100
k ₂			100	100	100	100	100	100	100	100
k ₃				98.14	98.57	99.98	99.99	100	100	100
k ₄					62.92	81.23	86.61	87.01	85.02	92.65
k ₅						50.57	58.05	55.58	62.17	65.6
k ₆							50.65	55.83	53.73	54.2
k ₇								50.74	48.21	54.36
k ₈									50.78	49.75
k ₉										50.99

(d) # dígitos= 2.5×10^5

	k ₁	k ₂	k ₃	k ₄	k ₅	k ₆	k ₇	k ₈	k ₉	k ₁₀
k ₀	100	100	100	100	100	100	100	100	100	100
k ₁		100	100	100	100	100	100	100	100	100
k ₂			100	100	100	100	100	100	100	100
k ₃				100	100	100	100	100	100	100
k ₄					100	100	100	100	100	100
k ₅						97.1	100	99.96	100	100
k ₆							74.87	90.78	96.32	98.62
k ₇								63.08	75.16	81.07
k ₈									55.92	57.69
k ₉										50.89

(e) # dígitos= 10^6

	k ₁	k ₂	k ₃	k ₄	k ₅	k ₆	k ₇	k ₈	k ₉	k ₁₀
k ₀	100	100	100	100	100	100	100	100	100	100
k ₁		100	100	100	100	100	100	100	100	100
k ₂			100	100	100	100	100	100	100	100
k ₃				100	100	100	100	100	100	100
k ₄					100	100	100	100	100	100
k ₅						100	100	100	100	100
k ₆							97.7	100	100	100
k ₇								78.56	92.38	97.27
k ₈									60.17	76.5
k ₉										61.65

Figura 53 – Resultados obtidos para a classificação dos pares de classes k_i vs k_{i+1} -mapa logístico usando como vetor de características à matriz de probabilidade de transição. As taxas de acerto correspondem à classificação usando kNN com $k = 1$ para cinco conjuntos com tamanhos de sequências diferentes geradas pelo PRNG k -mapa logístico.

Fonte: Elaborada pela autora.

No que tange ao estágio de seleção das regras, a família de regras *Life-like*, providencia um amplo espectro de regras contendo possíveis soluções a um determinado problema. Fazendo uma comparação paralela, podemos dizer que a seleção da regra é um estágio de “farejamento” pelas melhores regras que possam satisfazer um cenário específico, em alusão com os cães farejadores que precisam de um objeto alvo para cheirar, e a partir daí realizar uma busca. Porém a busca por todo o espaço de regras por uma solução ótima para um problema específico é ainda uma limitação do método, pelo qual outras estratégias merecem ser estudadas, como por exemplo, algoritmos genéticos.

A Figura 55 descreve a distribuição do histograma das taxas de acerto obtidas para o espaço de regras da família *Life-Like* para os diferentes conjuntos de redes usados neste capítulo. Nela observa-se que os problemas com menores números de regras com taxas

Figura 54 – Resumo da comparação dos resultados de classificação do método proposto e as medidas estruturais de redes para as diferentes bases de dados utilizadas neste capítulo. Os gráficos de barras apresentam a taxa de acerto média e desvio padrão (barra de erros) usando as melhores combinações de vetores de características do LLNA e as medidas estruturais de forma isolada ou combinada.

Fonte: Elaborada pela autora.

de acerto acima de 80% são o problema de redes textuais e das redes espaço-tempo do k -mapa logístico.

Os resultados apresentados neste trabalho são corroborados pelo procedimento de otimização de regra, que foi feito para cada conjunto de dados em questão, objetivando solucionar um problema específico de reconhecimento de padrões. A inserção de novas

Figura 55 – Distribuição do espaço de regras da família *Life-Like* para as seis bases de redes estudadas.

Fonte: Elaborada pela autora.

instâncias num conjunto de dados, provavelmente requisita um novo processo de treinamento. Selecionando um conjunto de regras discriminatórias que servisse para coordenar a dinâmica do autômato, observamos que a variação dos estados binários de vértices é mais discriminatória que uma abordagem de medidas estruturais de rede.

A melhor regra para um único conjunto de dados pode não ter uma performance tão boa em um segundo conjunto. Desta maneira, não há generalizações sobre a otimização do processo de encontrar a melhor regra do LLNA. Esta não é uma limitação do método, embora esta questão reflete características intrínsecas de cada fonte de dados. Consequentemente, a seleção das melhores regras tem que ser feita para cada conjunto.

Na Tabela 25 é apresentada a interseção das melhores regras encontradas para os diferentes contextos dentro de um conjunto de 10,000 regras. Observa-se que são relativamente poucas as regras em comum, e não é possível deduzir alguma relação entre os problemas.

Tabela 25 – Número de regras em comum dentro das 10,000 melhores selecionadas para a classificação das seis bases de redes.

	Metabólicas	Social	Textuais	Estômatos	k-mapa logístico
Sintéticas	309	314	244	376	398
Metabólicas		337	337	639	403
Social			262	388	391
Textuais				225	383
Estômatos					427

Fonte: Elaborada pela autora.

Devemos enfatizar que a aplicabilidade dos métodos será em dados que podem ser mais abrangentes desde séries temporais até sequências geradas por um PRNG. À vista disto, a principal motivação é a análise de sequências de números pseudo-aleatórios com o fim de encontrar padrões e/ou encontrar as formas de distinguir dentre várias classes de PRNGs. A abstração de tais sequências de dados, primeiro representadas como séries temporais e logo modeladas como redes complexas, nos abre a possibilidade de usar os

métodos de reconhecimento de padrões em redes para poder estudar PRNGs com o fim de “medir a pseudo-aleatoriedade” em tais sistemas. Assim, no primeiro contexto, o método traz contribuições na detecção de padrões. Já no segundo cenário, o mesmo método poderá ser usado como um teste de aleatoriedade.

7 CONCLUSÕES

Nesta tese foram apresentados padrões e pseudo-aleatoriedade como uma nova abordagem que relaciona ambos os conceitos, que tradicionalmente são vistos como opostos. Uma vez que padrões podem abranger arranjos desde os mais simples ou complexos até mesmo aleatórios. Este trabalho demonstrou que padrões e aleatoriedade se entrelaçam.

O desenvolvimento deste doutorado se assemelhou ao diagrama de bifurcação do mapa logístico, no sentido que, esta tese, se bifurcou primeiramente em dois ramos: aleatoriedade e padrões. O desenrolar da pesquisa nestes dois assuntos gerou novas bifurcações que por final teceram uma única malha.

As propriedades pseudo-aleatórias das profundezas dos sistemas caóticos foram a base para o estudo para relacionar padrões e pseudo-aleatoriedade. Inicialmente foram considerados os k dígitos menos significativos como fontes de pseudo-aleatoriedade. Ao longo desta tese, foi descoberto que as propriedades de pseudo-aleatoriedade de um mapa caótico podem ser melhoradas à medida que k é aumentado. Foi evidenciada uma transição rápida de uma aleatoriedade “fraca” para “forte” conforme k tende ao infinito o que permite manipular parametricamente a pseudo-aleatoriedade. Esta conjectura torna-se cada mais evidente, pois as diferentes estratégias aqui analisadas (testes estatísticos, todos os meios de análise de dinâmica de sistemas caóticos, análise do expoente de Lyapunov e análise espectral) corroboraram que k está relacionado às qualidades pseudo-aleatórias.

Na segunda parte da tese foi explorada a vida artificial advinda da dinâmica dos autômatos celulares usando as redes (tesselação) como habitat. Regidas por leis de vida e morte, os autômatos produziram padrões espaço-temporais que permitiram caracterizar as propriedades intrínsecas destas redes. Após validados os parâmetros da rede-autômato, este foi usado para caracterizar tanto redes sintéticas como redes reais, mostrando assim sua capacidade genérica e multidisciplinar. Cinco diferentes tarefas de reconhecimento de padrões foram avaliadas: (i) na identificação de padrões estruturais em redes sociais, (ii) na identificação de organismos de diferentes domínios da vida usando redes metabólicas; (iii) na análise de plasticidade fenotípica de plantas; e (iv) na identificação de autoria de manuscritos literários. Em todos estes casos, o método proposto superou as medidas estruturais de redes.

Esta mesma abordagem foi utilizada para analisar as sequências de números pseudo-aleatórios gerados pelo padrão ouro do k -mapa logístico no contexto de reconhecimento de padrões. O que gerou uma nova abordagem para investigar sequências pseudo-aleatórias como séries temporais no contexto de reconhecimento de padrões.

7.1 Contribuições e publicações

Continuando com a visão da árvore de bifurcações desta tese, cada um desses galhos rendeu várias contribuições.

- No que tange à teoria do caos, foi corroborado que, apesar da simplicidade de alguns sistemas caóticos (e.g. mapa logístico e o mapa da tenda), é possível obter ótimas fontes de pseudo-aleatoriedade. Isto pela forte relação com diferentes propriedades dos sistemas caóticos como a sensibilidade às condições iniciais, ergodicidade, transitividade, instabilidade e dinâmica determinística;
- Além disso, temos contribuído com uma nova forma de analisar as profundezas dos espaços de fases dos sistemas caóticos. Isto foi possível graças à análise dos dígitos menos significativos das órbitas dos mapas caóticos, os quais, tipicamente são ignorados e inclusive menosprezados na literatura. Nesta tese foi reforçado que a sensibilidade às condições iniciais é um dos principais pilares do caos, pois, traz consigo que os dígitos menos significativos são os mais vulneráveis a mudanças drásticas por conseguinte ótima fonte de pseudo-aleatoriedade com o qual foi desenvolvida mais uma contribuição que é o PRNG baseado no k -mapa logístico e o k -mapa da tenda, trabalho que rendeu artigo publicado na *Chaos*. (133);
- Uma outra contribuição desta tese foi a produção de um padrão ouro de pseudo-aleatoriedade parametrizado por k . O padrão ouro pode ser uma importante ferramenta de pesquisa que pode auxiliar o desenvolvimento de três diferentes áreas: o reconhecimento de padrões, a criptografia, e a criptoanálise. Uma vez que, poderá gerar conjuntos virtualmente infinitos com aleatoriedade com base teórica conhecida;
- No reconhecimento de padrões, o uso deste conjunto de dados pseudo-aleatórios trará importantes avanços, se usado como *benchmark* para avaliação do poder dos métodos. Uma vez que fornece um conjunto de dados com base teórica e, ao mesmo tempo, muito desafiador;
- Esta mesma lógica traz contribuições para a criptoanálise desde que permite a criar novos métodos para avaliar ou testar a pseudo-aleatoriedade;
- Uma outra contribuição desta tese são as bases matemáticas de forma generalizada, que permitem estudar a análise das profundezas do caos para qualquer sistema caótico;
- É outra contribuição desta tese, nesse sentido, que as ferramentas da teoria do caos, como, por exemplo, o expoente de Lyapunov serviram para analisar padrões em sistemas criptográficos, trabalho que rendeu artigo publicado na *Communications in Nonlinear Science and Numerical Simulation* (5);

- Outra contribuição tange às limitações dos testes de aleatoriedade. Nesta tese foi identificada a falta de uma nova métrica para medir a pseudo-aleatoriedade, de tal forma que seja possível explorar novos métodos que consigam medir as qualidades pseudo-aleatórias de forma mais precisa;
- Com relação ao segundo ramo da bifurcação, esta tese contribuiu com uma definição formalizada da combinação das redes e autômatos celulares sob a perspectiva de sistemas dinâmicos. A partir da dinâmica adicionada à rede-autômato foi possível extrair padrões espaço-tempo, os quais são trazidos de um entorno adimensional para uma forma de visualização em duas dimensões;
- A partir do desenvolvimento dos padrões espaço-tempo, uma outra contribuição é usar diferentes medidas para caracterizar as propriedades intrínsecas das redes. Os resultados demonstraram que o método é invariante ao tamanho das redes tanto quanto é tolerante ao ruído;
- Uma outra contribuição foi introduzir o espaço de regras da família *Life-Like* advindas de autômatos celulares bidimensionais no contexto de redes, para o qual foi usado um critério de densidade da vizinhança;
- Esta tese também trouxe contribuição com respeito à multidisciplinaridade dos problemas de reconhecimento de padrões estudados sob a perspectiva de redes complexas. Cada um desses problemas é uma contribuição por si só. Trabalho que rendeu artigo publicado na *Scientific Reports* (50);
 - Na identificação de padrões estruturais em redes sociais *online*;
 - Na identificação de organismos de diferentes domínios da vida usando suas reações metabólicas representadas como redes num contexto de reconhecimento de padrões;
 - Na classificação de padrões de distribuição em imagens microscópicas de estômatos, cujas espécies foram submetidas a diferentes condições de estresse;
 - Também temos contribuído no problema de atribuição de autoria por meio da análise dos padrões espaço-tempo extraídos das redes textuais, como alternativa complementar (ou hibridização) aos métodos tradicionais de processamento de linguagem natural;
 - Metodologia para o reconhecimento de padrões de sequências pseudo-aleatórias gerados pelo padrão ouro k -mapa logístico.
- Além da multidisciplinaridade, os problemas tratados estão inseridos no contexto de *big-data* e ciências dos dados. Uma vez que apresentam informações de diferentes naturezas e tamanhos tratados com a mesma abordagem.

A Tabela 26 que apresenta a síntese das principais contribuições desta tese.

Tabela 26 – Síntese das principais contribuições desta tese.

Abordagem	Contribuições
Aleatoriedade nas profundezas do caos	☐ Foi corroborado com a literatura de que os sistemas caóticos, inclusive os mais simples como é o mapa logístico ou o mapa da tenda, podem ser ótimos PRNGs. ☐ Proposta de uma análise das profundezas usando os dígitos menos significativos das órbitas dos mapas caóticos. ☐ Proposta de um PRNG forte usando o caos do mapa logístico e do mapa da tenda. ☐ Padrão ouro: criação de um gerador de sequências pseudo-aleatórias com complexidade parametrizada por k , que é muito importante para analisar e propor novos métodos de reconhecimento de padrões e inteligência artificial.
Aplicações na Criptografia/Criptoanálise	☐ Teste de aleatoriedade baseado no <i>deep-zoom</i> deram as bases matemáticas para avaliar a pseudo-aleatoriedade. ☐ Criptoanálise usando caos. ☐ Possibilidade de criar Criptografia forte usando caos. ☐ Levantamento das principais questões sobre as limitações dos testes de aleatoriedade estatísticos.
Vida artificial nas redes e Reconhecimento de padrões	☐ Análise de padrões espaço-temporais em redes. ☐ Realização de autômato celular <i>Life-Like</i> em redes. ☐ Proposta de modelo <i>Life-Like</i> de rede-autômato. ☐ Uso de padrão espaço-tempo para extração de características para o reconhecimento de padrões. ☐ Método invariante ao tamanho da rede N e ao ruído e transformações da rede. ☐ Diversas aplicações do LLNA no contexto de problemas do mundo real usando redes sociais, redes metabólicas, redes textuais, redes estomáticas, redes espaço-tempo, redes de transição. ☐ Proposta de método para <i>big-data</i> e ciência dos dados.

Fonte: Elaborada pela autora.

Esta tese gerou as seguintes publicações:

• Publicações:

- ✓ MACHICAO, J.; RIBAS, L.; SCABINI, L.; BRUNO, O. M. Cellular automata rule characterization and classification using texture descriptors. **Physica A** v. 497, p. 109—117, 2018;
- ✓ MACHICAO, J.; BRUNO, O. M. Improving the pseudo-randomness properties of chaotic maps using deep-zoom. **Chaos: an interdisciplinary journal of nonlinear science** v. 27 p. 053116, 2017;

- ✓ MACHICAO, J.; BRUNO, O. M. A cryptographic hash function based on chaotic network automata. **Journal of Physics: conference series** v. 936, p. 012058, 2017;
- ✓ ALMEIDA, H. A.; MACHICAO, J.; BRUNO, O. M. Geometry from stomata networks at leaves of the *Ctenanthe oppenheimiana*. **Journal of Physics: conference series** v. 936, p. 012085, 2017;
- ✓ ALMEIDA, H. A.; MACHICAO, J.; BRUNO, O. M. Geometric plasticity at leaves from *Ctenanthe oppenheimiana* probed by measure of distances between stomata. **Journal of Physics: conference series** v. 936, p. 012094, 2017;
- ✓ MIRANDA, G. H. B.; MACHICAO, J.; BRUNO, O. M. Exploring spatio-temporal dynamics of cellular automata for pattern recognition in networks. **Scientific Reports** v. 6 n. 37329, 2016;
- ✓ MIRANDA, G. H. B.; MACHICAO, J.; BRUNO, O. M. Network Analysis Using Spatio-Temporal Patterns. **Journal of Physics: conference series** v. 738 p. 012011, 2016;
- ✓ MACHICAO, J.; BAETENS, J. M.; MARCO, A. G.; DE BAETS, B.; BRUNO, O. M. A dynamical system approach to the discrimination of the modes of operation of cryptographic systems. **Communications in Nonlinear Science and Numerical Simulation** v. 29 n. 1–3, p. 102–115, 2015;

• **Submetidos:**

- ✓ MACHICAO, J.; ALMEIDA, H. A.; LAHR, D. J. G.; BUCKERIDGE, M.; BRUNO, O. M. Topological assessment of metabolic networks reveals evolutionary information. **Scientific Reports**;
- ✓ ALMEIDA, H. A.; MACHICAO, J.; BRUNO, O. M. A hierarchical model of metabolic machinery based on the *k*core decomposition of plant metabolic networks. **Plos One**;
- ✓ MACHICAO, J.; CORRÊA E. Jr.; MIRANDA, G.H.B.; AMANCIO, D.; BRUNO, O. M. Authorship attribution based on Life-Like network automata. **Plos One**;

Publicado em ([229](#))

- ✓ MIRANDA, G. H. B.; MACHICAO, J.; BRUNO, O. M. An optimized shape descriptor based on structural properties of networks. **Digital Signal Processing**.

Publicado em ([234](#))

• **Em fase de redação:**

- ✓ MACHICAO, J.; ALVES, M.; BAPTISTA, M.; BRUNO, O. M. Cryptographic method based on the chaotic properties of the *k*-logistic map;
- ✓ MACHICAO, J.; BRUNO, O. M. Dynamical analysis of a deformation on unimodal maps;
- ✓ MACHICAO, J.; BRUNO, O. M. The logistic map and the number of the beast;

✓ MACHICAO, J.; ALVES, M.; BRUNO, O. M Analyzing the Life-Like network automata rule space based on parametric pseudo-randomness;

7.2 Trabalhos futuros

Muito ainda há para ser explorado no paradigma de padrões e pseudo-aleatoriedade nos sistemas complexos. Algumas oportunidades de pesquisa diretamente oriundas deste doutorado são apresentados na sequência.

- Um dos trabalhos mais próximos é o estudo de rede-autômato totalística e outer-totalística, e outros tipos/famílias de regras. Além disso explorar as propriedades de cada regra;
- Estudar novas medidas que possam ser extraídas do padrão espaço-tempo;
- Calcular o expoente de Lyapunov nas rede-autômato;
- Aprofundar no estudo das redes espaço-tempo;
- Novos métodos para aperfeiçoar o estágio de seleção das regras;
- Explorar novos mapas iterativos;
- Estudar que acontece quando $\mu = 3.99999$ ou o mais tendente a 4;
- Investigar o que acontece quando o parâmetros k tende a infinito;
- Encontrar formas de analisar analiticamente o expoente de Lyapunov do k -mapa logístico;
- Estudar outras formas de modelar as redes de sequências pseudo-aleatórias, explorar novos limiares, ou outras formas além das de cadeias de Markov;
- Outra pergunta que buscamos responder diz respeito as consequências, implicações e aplicações dos padrões encontrados;
- Essa combinação de aleatoriedade e regularidade tem motivado pesquisadores a procurarem padrões em sequências de dados que eventualmente possam lançar alguma luz sobre sua natureza final.

REFERÊNCIAS

- 1 GLEICK, J. **Chaos: making a new science**. New York: Penguin, 1988. (A Penguin Book: Science).
- 2 CRUTCHFIELD, J. Between order and chaos. **Nature Physics**, v. 8, n. 1, p. 17–24, 2012.
- 3 HAYLES, N. **How we became posthuman: virtual bodies in cybernetics, literature, and informatics**. Chicago, Ill: University of Chicago Press, 1999. 364 p.
- 4 MACHICAO, J. **Autômatos celulares caóticos aplicados na criptografia e criptoanálise**. 2013. 121 p. Dissertação (Mestrado em Ciências) — Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2013.
- 5 MACHICAO, J. et al. A dynamical systems approach to the discrimination of the modes of operation of cryptographic systems. **Communications in Nonlinear Science and Numerical Simulation**, v. 29, n. 1-3, p. 102–115, 2015.
- 6 BOCCARA, N. **Modeling complex systems**. Berlin: Springer Verlag, 2004. (Graduate texts in contemporary physics).
- 7 DEVANEY, R. **An introduction to chaotic dynamical systems**. Boulder, CO, USA: Addison-Wesley, 1989. 50 p. (Advanced Book Program).
- 8 FROYLAND, J. **Introduction to chaos and coherence**. New York: CRC Press, 1992.
- 9 THEODORIDIS, S.; KOUTROUMBAS, K. **Pattern recognition**. 4th. ed. New York: Academic Press, 2008.
- 10 GIRVAN, M.; NEWMAN, M. Community structure in social and biological network. **Proceedings of the National Academy of Sciences of the United States of America**, v. 99, n. 12, p. 7821–7826, 2002.
- 11 NEWMAN, M. E. The structure and function of complex networks. **Society for Industrial and Applied Mathematics Review**, v. 45, n. 2, p. 167–256, 2003.
- 12 RAIN, J.-C.; AL. et. The protein–protein interaction map of helicobacter pylori. **Nature**, v. 409, n. 6817, p. 211–215, 2001.
- 13 VALVERDE, S.; CANCHO, R. F.; SOLE, R. V. Scale-free networks from optimal design. **Europhysics Letters**, v. 60, n. 4, p. 512, 2002.
- 14 COSTA, L. d. F.; RODRIGUES, F. A.; TRAVIESO, G.; BOAS, P. V. Characterization of complex networks: a survey of measurements. **Advances In Physics**, v. 56, n. 1, p. 167, 2007.
- 15 DONNER, R. V. et al. Recurrence networks — a novel paradigm for nonlinear time series analysis. **New Journal of Physics**, v. 12, n. 3, p. 033025, 2010.

- 16 SILVA, T.; ZHAO, L. Network construction techniques. In: _____. **Machine learning in complex networks**. Berlin: Springer International Publishing, 2016. p. 93–132.
- 17 BACKES, A. R.; CASANOVA, D.; BRUNO, O. M. A complex network-based approach for boundary shape analysis. **Pattern Recognition**, v. 42, n. 1, p. 54–67, 2009.
- 18 _____. Texture analysis and classification: a complex network-based approach. **Information Sciences**, v. 219, p. 168–180, 2013.
- 19 GONÇALVES, W. N.; MACHADO, B.; BRUNO, O. M. A complex network approach for dynamic texture recognition. **Neurocomputing**, v. 153, n. 0, p. 211–220, 2015.
- 20 MAY, R. M. Simple mathematical models with very complicated dynamics. **Nature**, v. 261, n. 5560, p. 459–467, 1976.
- 21 WOLFRAM, S. Statistical mechanics of cellular automata. **Reviews of Modern Physics**, v. 55, n. 3, p. 601–644, 1983.
- 22 BARABÁSI, A.-L.; PÓSFAL, M. **Network science**. Cambridge: Cambridge University Press, 2016.
- 23 JEONG, H.; TOMBOR, B. R.; OLTVAI, Z. N.; BARABÁSI, A. L. The large-scale organization of metabolic networks. **Nature**, v. 407, n. 6804, p. 651–654, 2000.
- 24 RODRIGUES, F. A. **Caracterização, classificação e análise de redes complexas**. 2007. 110 p. Tese (Doutorado) — Instituto de Física de São Carlos, Universidade de São Paulo, 2007.
- 25 NEWMAN, M. **Networks: an introduction**. Oxford: OUP Oxford, 2010.
- 26 GROS, C. **Complex and adaptive dynamical systems: a primer**. Berlin: Springer Verlag, 2008.
- 27 LORENZ, E. N. Deterministic nonperiodic flow. **Journal of the Atmospheric Sciences**, v. 20, n. 2, p. 130–141, 1963.
- 28 LI, T.-Y.; YORKE, J. A. Period three implies chaos. **American Mathematical Monthly**, v. 82, n. 10, p. 985–992, 1975.
- 29 MANDELBROT, B. **The fractal geometry of nature**. New York: Henry Holt and Company, 1982.
- 30 SMITH, P. **Explaining chaos**. Cambridge, UK: Cambridge University Press, 1998.
- 31 ZUCHOWSKI, L. C. **A philosophical analysis of chaos theory**. Salzburg, Austria: Palgrave Macmillan, 2017. (New Directions in the Philosophy of Science).
- 32 YUGUDA, K. On elements of deterministic chaos and cross links in non-linear dynamical systems. **IOSR Journal of Mathematics**, v. 11, n. 6, p. 13–16, 2015.
- 33 PEITGEN, H.; JÜRGENS, H.; SAUPE, D. **Chaos and fractals: new frontiers of science**. New York: Springer, 2004.

- 34 BANKS, J. et al. On devaney's definition of chaos. **American Mathematical Monthly**, v. 99, n. 4, p. 332–334, 1992.
- 35 ECKMANN, J.-P.; RUELLE, D. Ergodic theory of chaos and strange attractors. **Review of Modern Physics**, v. 57, n. 3, part 1, p. 617–656, 1985.
- 36 OTT, E. **Chaos in dynamical systems**. Cambridge, UK: Cambridge University Press, 2002.
- 37 BAETENS, J. M.; BAETS, B. D. Phenomenological study of irregular cellular automata based on Lyapunov exponents and Jacobians. **Chaos**, v. 20, n. 3, p. 033112, 2010.
- 38 WOLF, A.; SWIFT, J. B.; SWINNEY, H. L.; VASTANO, J. A. Determining lyapunov exponents from a time series. **Physica D: nonlinear phenomena**, v. 16, n. 3, p. 285–317, 1985.
- 39 ECKMANN, J.; WITTEWER, P. A complete proof of the feigenbaum conjectures. **Journal of Statistical Physics**, v. 46, n. 3, p. 455–475, 1987.
- 40 KING, C. C. Fractal and chaotic dynamics in nervous systems. **Progress in Neurobiology**, v. 36, n. 4, p. 279–308, 1991.
- 41 HAO, B.-L. **Elementary symbolic dynamics and chaos in dissipative systems**. Singapore: World Scientific, 1989.
- 42 WOLFRAM, S. Twenty problems in the theory of cellular automata. **Physica Scripta**, T9, p. 170–183, 1985.
- 43 _____. Universality and complexity in cellular automata. **Physica D**, v. 10, n. 1-2, p. 1–35, 1984.
- 44 GARDNER, M. The fantastic combinations of John Conway's new solitaire game "life". **Scientific American**, v. 223, p. 120–123, 1970.
- 45 WOLFRAM, S. Cellular automata. **Los Alamos Science**, v. 9, p. 2–27, 1983.
- 46 _____. **Cellular automata and complexity: collected papers**. Reading, MA: Addison-Wesley, 1994.
- 47 _____. **A new kind of science**. Champaign, Illinois, USA: Wolfram Media, Inc, 2002.
- 48 BAETENS, J. M.; BAETS, B. D. On the topological sensitivity of cellular automata. **Chaos: an interdisciplinary journal of nonlinear science**, v. 21, n. 2, p. 023108, 2011.
- 49 MACHICAO, J.; MARCO, A.; BRUNO, O. M. Chaotic encryption method based on life-like cellular automata. **Expert Systems with Applications**, v. 39, p. 12626–12635, 2012.
- 50 MIRANDA, G. H. B.; MACHICAO, J.; BRUNO, O. M. Exploring spatio-temporal dynamics of cellular automata for pattern recognition in networks. **Scientific Reports**, v. 6, n. 37329, 2016.

- 51 WATTS, D. J. **Small worlds: the dynamics of networks between order and randomness**. Princeton: Princeton University Press, 1999. (Princeton studies in complexity).
- 52 MARR, C.; HÜTT, M. Outer-totalistic cellular automata on graphs. **Physics Letters A**, v. 373, n. 5, p. 546–549, 2009.
- 53 SMITH, D. et al. Network automata: coupling structure and function in dynamic networks. **Advances in Complex Systems**, v. 14, n. 3, p. 317–339, 2011.
- 54 ZHOU, H.; LIPOWSKY, R. Dynamic pattern evolution on scale-free networks. **Proceedings of the National Academy of Sciences of the United States of America**, v. 102, n. 29, p. 10052–10057, 2005.
- 55 WU, A.-C.; XU, X.-J.; WANG, Y.-H. Excitable greenberg-hastings cellular automaton model on scale-free networks. **Physical Review E**, v. 75, n. 3, p. 032901, 2007.
- 56 BAETENS, J.; BAETS, B. D. Cellular automata on irregular tessellations. **Dynamical Systems**, v. 27, n. 4, p. 411–430, 2012.
- 57 ALBERT, R.; BARABÁSI, A.-L. Statistical mechanics of complex networks. **Reviews of Modern Physics**, v. 74, n. 1, p. 47–97, 2002.
- 58 COSTA, L. d. F.; AL. et. Analyzing and modeling real-world phenomena with complex networks: a survey of applications. **Advances in Physics**, v. 60, n. 3, p. 329–412, 2011.
- 59 WATTS, D. J.; STROGATZ, S. H. Collective dynamics of “small-world” networks. **Nature**, v. 393, n. 6684, p. 440–442, 1998.
- 60 NEWMAN, M. E. J. Assortative mixing in networks. **Physical Review Letters**, v. 89, n. 20, p. 208701, 2002.
- 61 BANERJEE, A.; JOST, J. Spectral plot properties: towards a qualitative classification of networks. **Networks and Heterogeneous Media**, v. 3, n. 2, p. 395–411, 2008.
- 62 KLEINBERG, J. Authoritative sources in a hyperlinked environment. **Journal of the ACM**, New York, v. 46, n. 5, p. 604–632, 1999.
- 63 ERDÖS, P.; RÉNYI, A. On random graphs. **Publicationes Mathematicae Debrecen**, v. 6, p. 290–297, 1959.
- 64 NEWMAN, M. E. J.; PARK, J. Why social networks are different from other types of networks. **Physical Review E**, v. 68, p. 036122, 2003.
- 65 BARABÁSI, A.-L.; ALBERT, R. Emergence of scaling in random networks. **Science**, v. 286, n. 5439, p. 509–512, 1999.
- 66 BARABÁSI, A.-L.; BONABEAU, E. Scale-free networks. **Scientific American**, v. 288, n. 60–69, 2003.
- 67 CLAUSET, A.; SHALIZI, C. R.; NEWMAN, M. E. J. Power-law distributions in empirical data. **Society for Industrial and Applied Mathematics Review**, v. 51, n. 4, p. 661–703, 2009.

- 68 DOROGOVTSSEV, S. N.; MENDES, J. F. Evolution of networks. **Advances in Physics**, v. 51, n. 4, p. 1079–1187, 2002.
- 69 WAXMAN, B. M. Routing of multipoint connections. **IEEE Journal on Selected Areas in Communications**, v. 6, n. 9, p. 1617–1622, 1988.
- 70 RAD, A. A.; BENYOUCEF, M. Towards detecting influential users in social networks. In: _____. INTERNATIONAL CONFERENCE, 5th. MCETECH, 2011, Les Diablerets. **Proceedings...** Berlin: Springer Verlag, 2011. p. 227–240.
- 71 LILJEROS, F. et al. The web of human sexual contacts. **Nature**, v. 411, n. 6840, p. 907–908, 2001.
- 72 TRAVERS, J.; MILGRAM, S. An experimental study of the small world problem. **Sociometry**, v. 32, n. 4, p. 425–443, 1969.
- 73 HUSS, M.; HOLME, P. Currency and commodity metabolites: their identification and relation to the modularity of metabolic networks. **IET Systems Biology**, v. 15, p. 280–285, 2007.
- 74 MARTÍNEZ, V.; BERZAL, F.; CUBERO, J.-C. A survey of link prediction in complex networks. **ACM Computing Surveys**, v. 49, n. 4, p. 1–33, 2016.
- 75 WENG, L.; MENCZER, F.; MENCZER, F.; AHN, Y.-Y. Virality prediction and community structure in social networks. **Scientific Reports**, v. 3, p. 2522, 2013.
- 76 DAVIDSON, M. W.; HAIM, D. A.; RADIN, J. M. Using networks to combine “big data” and traditional surveillance to improve influenza predictions. **Scientific Reports**, v. 5, p. 8154, 2015.
- 77 COSTA, L. d. F. P. B.; SILVA, F.; RODRIGUES, F. A pattern recognition approach to complex networks. **Journal of Statistical Mechanics: theory and experiment**, v. 2010, n. 11, p. P11015, 2010.
- 78 CASANOVA, D. **Redes complexas em visão computacional com aplicações em bioinformática**. 2013. Tese (Doutorado em Física Aplicada) — Instituto de Física de São Carlos, Universidade de São Paulo, 2013.
- 79 GONÇALVES, W. N.; SILVA, N. R. da; COSTA, L. d. F.; BRUNO, O. M. Texture recognition based on diffusion in networks. **Information Sciences**, v. 364, p. 51–71, 2016.
- 80 BACKES, A. R.; BRUNO, O. M. Shape classification using complex network and multi-scale fractal dimension. **Pattern Recognition Letters**, v. 31, n. 1, p. 44–51, 2010.
- 81 BACKES, A. R.; CASANOVA, D.; BRUNO, O. M. Contour polygonal approximation using the shortest path in networks. **International Journal of Modern Physics**, v. 25, n. 2, 2014.
- 82 ZHAO, J. et al. Complex networks theory for analyzing metabolic networks. **Chinese Science Bulletin**, v. 51, n. 13, p. 1529–1537, 2006.
- 83 OVERBEEK, R.; AL. et. WIT: integrated system for high-throughput genome sequence analysis and metabolic reconstruction. **Nucleic Acids Research**, v. 28, n. 1, p. 123–125, 2000.

- 84 BANERJEE, A. Structural distance and evolutionary relationship of networks. **Biosystems**, v. 107, n. 3, p. 186–196, 2012.
- 85 AMANCIO, D. R.; ALTMANN, E. G.; OLIVEIRA JUNIOR, O. N.; COSTA, L. d. F. Comparing intermittency and network measurements of words and their dependence on authorship. **New Journal of Physics**, v. 13, n. 12, p. 123024, 2011.
- 86 MEHRI, A.; DAROONEH, A. H.; SHARIATI, A. The complex networks approach for authorship attribution of books. **Physica A: statistical mechanics and its applications**, v. 391, n. 7, p. 2429–2437, 2012.
- 87 AKIMUSHKIN, C.; AMANCIO, D. R.; OLIVEIRA JUNIOR, O. N. Text authorship identified using the dynamics of word co-occurrence networks. **Plos One**, v. 12, n. 1, p. 1–15, 2017.
- 88 AMANCIO, D. R.; SILVA, F. N.; COSTA, L. d. F. Concentric network symmetry grasps authors' styles in word adjacency networks. **Europhysics Letters**, v. 110, n. 6, p. 68001, 2015.
- 89 AMANCIO, D. R. A complex network approach to stylometry. **Plos One**, v. 10, n. 8, p. e0136076, 2015.
- 90 LUI, K. W.; SO, H.; CHEN, G. The π sequence as a complex network. **Physica A: statistical mechanics and its applications**, v. 387, n. 22, p. 5653–5661, 2008.
- 91 CHEN, C.; LU, J. an; WU, X. Complex networks constructed from irrational number sequences. **Physica A: statistical mechanics and its applications**, v. 389, n. 13, p. 2654–2662, 2010.
- 92 DUDA, R.; HART, P.; STORK, D. **Pattern classification**. New York: Wiley, 2001. (Pattern Classification and Scene Analysis: Pattern Classification).
- 93 ALDERSON, D.; DOYLE, J. C.; LI, L.; WILLINGER, W. Towards a theory of scale-free graphs: definition, properties, and implications. **Internet Mathematics**, v. 2, n. 4, p. 431–523, 2005.
- 94 GONÇALVES, W. N.; MARTINEZ, A. S.; BRUNO, O. M. Complex network classification using partially self-avoiding deterministic walks. **Chaos: an interdisciplinary journal of nonlinear science**, v. 22, n. 3, p. 033139, 2012.
- 95 RIBAS, L. C. **Análise de texturas dinâmicas baseada em sistemas complexos**. 2017. Dissertação (Mestrado em Ciências de Computação e Matemática Computacional) — Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo, 2017.
- 96 BISHOP, C. M. **Pattern recognition and machine learning (information science and statistics)**. Secaucus, NJ, USA: Springer Verlag, 2006.
- 97 AMANCIO, D. R. et al. A systematic comparison of supervised classifiers. **Plos One**, v. 9, n. 4, p. e94137, 04 2014.
- 98 VAPNIK, V. N. **The nature of statistical learning theory**. New York, NY, USA: Springer Verlag, 1995.

- 99 HEARST, M. A. et al. Support vector machines. **IEEE Intelligent Systems and their Applications**, v. 13, n. 4, p. 18–28, 1998.
- 100 JOHNSON, R.; WICHERN, D. **Applied multivariate statistical analysis**. Upper Saddle River, NJ, USA: Pearson Prentice Hall, 1998. (Applied Multivariate Statistical Analysis).
- 101 MAATEN, L. van der; HINTON, G. Visualizing high-dimensional data using t-SNE. **Machine Learning Research**, v. 9, p. 2579–2605, 2008.
- 102 MAATEN, L. van der. **t-SNE**. 2017. Disponível em: <<https://lvdmaaten.github.io/tsne/>>. Acesso em: 8.10.2017.
- 103 WOLFRAM, S. Cryptography with cellular automata. In: ADVANCES IN CRYPTOLOGY, CRYPTO'85, 1985, Santa Barbara, CA. **Proceedings...** Santa Barbara, CA, USA: Springer Verlag, 1986. (Lecture notes in computer sciences, v. 218), p. 429–432.
- 104 MATTHEWS, R. On the derivation of a “chaotic” encryption algorithm. **Cryptologia**, v. 13, n. 1, p. 29–42, 1989.
- 105 WHEELER, D. D. Problems with chaotic cryptosystems. **Cryptologia**, v. 13, n. 3, p. 243–250, 1989.
- 106 WHEELER, D. D.; MATTHEWS, R. A. J. Supercomputer investigations of a chaotic encryption algorithm. **Cryptologia**, v. 15, n. 2, p. 140–152, 1991.
- 107 HABUTSU, T.; NISHIO, Y.; SASASE, I.; MORI, S. A secret key cryptosystem by iterating a chaotic map. In: ADVANCES IN CRYPTOLOGY, EUROCRYPT'91, 1991, Brighton, UK. **Proceedings...** Brighton, UK: Springer Verlag, 1991. (Lecture Notes in Computer Science, v. 547), p. 127–140.
- 108 BIHAM, E. Cryptanalysis of the chaotic-map cryptosystem suggested at eurocrypt'91. In: ANNUAL INTERNATIONAL CONFERENCE ON THEORY AND APPLICATION OF CRYPTOGRAPHIC TECHNIQUES, 10th, 1991, Berlin, **Proceedings...** New York: Springer Verlag, 1991. p. 532–534.
- 109 BAPTISTA, M. Cryptography with chaos. **Physics Letters A**, v. 240, p. 50–54, 1998.
- 110 ÁLVAREZ, E. et al. New approach to chaotic encryption. **Physics Letters A**, v. 263, p. 373–375, 1999.
- 111 ÁLVAREZ, G.; MONTROYA, F.; ROMERA, M.; PASTOR, G. Cryptanalysis of a chaotic encryption system. **Physics Letters A**, v. 276, p. 191–196, 2000.
- 112 LI, S.; MOU, X.; CAI, Y. Improving security of a chaotic encryption approach. **Physics Letters A**, v. 290, n. 3, p. 127–133, 2001. ISSN 0375-9601.
- 113 WONG, W.-K.; LEE, L.-P.; WONG, K.-W. A modified chaotic cryptographic method. In: _____. JOINT WORKING CONFERENCE ON COMMUNICATIONS AND MULTIMEDIA SECURITY ISSUES OF THE NEW CENTURY, 11 th, CMS'01, 2001, Darmstadt, Germany. **Proceedings...** Boston, MA: Springer, 2001. p. 123–126.

- 114 WONG, K. A fast chaotic cryptographic scheme with dynamic look-up table. **Physics Letters A**, v. 298, n. 4, p. 238–242, 2002.
- 115 ÁLVAREZ, G.; MONTOYA, F.; ROMERA, M.; PASTOR, G. Cryptanalysis of an ergodic chaotic cipher. **Physics Letters A**, v. 311, n. 2–3, p. 172–179, 2003.
- 116 ÁLVAREZ, G.; MONTOYA, F.; ROMERA, M.; PASTOR, G. Keystream cryptanalysis of a chaotic cryptographic method. **Computer Physics Communications**, v. 156, n. 2, p. 205–207, 2004.
- 117 _____. Cryptanalysis of dynamic look-up table based chaotic cryptosystems. **Physics Letters A**, v. 326, n. 3, p. 211–218, 2004.
- 118 ADDABBO, T. et al. The digital tent map: performance analysis and optimized design as a low-complexity source of pseudorandom bits. **IEEE Transactions on Instrumentation and Measurement**, v. 55, n. 5, p. 1451–1458, 2006.
- 119 PATIDAR, V.; SUD, K. K.; PAREEK, N. K. A pseudo random bit generator based on chaotic logistic map and its statistical testing. **Informatica**, v. 33, p. 441–452, 2009.
- 120 KANSO, A.; SMAOUI, N. Logistic chaotic maps for binary numbers generations. **Chaos, Solitons & Fractals**, v. 40, n. 5, p. 2557–2568, 2009.
- 121 CECEN, S.; DEMIRER, R. M.; BAYRAK, C. A new hybrid nonlinear congruential number generator based on higher functional power of logistic maps. **Chaos, Solitons & Fractals**, v. 42, n. 2, p. 847–853, 2009.
- 122 PAREEK, N. K.; PATIDAR, V.; SUD, K. K. A random bit generator using chaotic maps. **International Journal of Network Security**, v. 10, p. 32–38, 2010.
- 123 MARCO, A.; MARTINEZ, A. S.; BRUNO, O. M. Fast, parallel and secure cryptography algorithm using Lorenz attractor. **International Journal of Modern Physics C**, v. 21, p. 12626–12635, 2010.
- 124 NIAN-SHENG, L. Pseudo-randomness and complexity of binary sequences generated by the chaotic system. **Communications in Nonlinear Science and Numerical Simulation**, v. 16, n. 2, p. 761–768, 2011.
- 125 DABAL, P.; PELKA, R. A chaos-based pseudo-random bit generator implemented in FPGA device. In: INTERNATIONAL SYMPOSIUM OF DESIGN AND DIAGNOSTICS OF ELECTRONIC CIRCUITS SYSTEMS, 14th, (DDECS). **Proceedings...** Cottbus, Germany: IEEE, 2011. p. 151–154.
- 126 BEHNIA, S.; AKHAVAN, A.; AKHSHANI, A.; SAMSUDIN, A. A novel dynamic model of pseudo random number generator. **Journal of Computational and Applied Mathematics**, v. 235, n. 12, p. 3455–3463, 2011.
- 127 ADDABBO, T.; FORT, A.; ROCCHI, S.; VIGNOLI, V. Digitized chaos for pseudo-random number generation in cryptography. In: KOCAREV, L.; LIAN, S. (Ed.). **Chaos-based cryptography. Studies in computational intelligence**. Berlin: Springer Nature, 2011. v. 354, p. 67–97.

- 128 LUCA, A.; ILYAS, A.; VLAD, A. Generating random binary sequences using tent map. In: INTERNATIONAL SYMPOSIUM ON SIGNALS, CIRCUITS AND SYSTEMS, 10 th, ISSCS, 2001, Lasi, Romania. **Proceedings...** New York: IEEE, 2011. p. 1–4.
- 129 LI, C. Y. et al. Period extension and randomness enhancement using high-throughput reseeding-mixing prng. **IEEE Transactions on Very Large Scale Integration (VLSI) Systems**, v. 20, n. 2, p. 385–389, 2012.
- 130 PERSOHN, K.; POVINELLI, R. Analyzing logistic map pseudorandom number generators for periodicity induced by finite precision floating-point representation. **Chaos, Solitons & Fractals**, v. 45, p. 238–245, 2012.
- 131 MIN, L.; CHEN, T.; ZANG, H. Analysis of fips 140-2 test and chaos-based pseudorandom number generator. **Chaotic Modeling and Simulation**, v. 2, n. 1, p. 273–280, 2013.
- 132 FRANÇOIS, M.; GROSGES, T.; BARCHIESI, D.; ERRA, R. Pseudo-random number generator based on mixing of three chaotic maps. **Communications in Nonlinear Science and Numerical Simulation**, v. 19, n. 4, p. 887–895, 2014.
- 133 MACHICAO, J.; BRUNO, O. M. Improving the pseudo-randomness properties of chaotic maps using deep-zoom. **Chaos: an interdisciplinary journal of nonlinear science**, v. 27, p. 053116, 2017.
- 134 SHANNON, C. E. **A mathematical theory of communication**. New York, NY, USA, 1948. v. 27, 623-423 p.
- 135 LEMPEL, A.; ZIV, J. On the complexity of finite sequences. **IEEE Transactions on Information Theory**, v. 22, n. 1, p. 75–81, 1976.
- 136 FIGURSKA, M.; STAŃCZYK, M.; KULESZA, K. Humans cannot consciously generate random numbers sequences: polemic study. **Medical Hypotheses**, v. 70, n. 1, p. 182–185, 2008.
- 137 HAWKING, S. W. **Does God play dice?** 1999. Disponível em: <<http://www.hawking.org.uk/does-god-play-dice.html>>. Acesso em: 01.10.2017.
- 138 KNUTH, D. E. **The art of computer programming: seminumerical algorithms**. 3rd. ed. Boston, MA, USA: Addison Wesley Longman Publishing Co., Inc., 1997. v. 2.
- 139 PRESS, W. H.; FLANNERY, B. P.; TEUKOLSKY, S. A.; VETTERLING, W. T. **Numerical recipes in C: the art of scientific computing**. 2nd. ed. Cambridge: Cambridge University Press, 1992. 269–277 p. (Fortran numerical recipes).
- 140 SOLOMONOFF, R. A formal theory of inductive inference. Part I. **Information and Control**, v. 7, n. 1, p. 1–22, 1964.
- 141 _____. A formal theory of inductive inference. Part II. **Information and Control**, v. 7, n. 2, p. 224–254, 1964.
- 142 KOLMOGOROV, A. N. Three approaches for defining the concept of information quantity. **Problems of Information Transmission**, v. 1, n. 1, p. 1–7, 1965.

- 143 CHAITIN, G. **Exploring Randomness**. London: Springer, 2012. (Discrete Mathematics and Theoretical Computer Science).
- 144 KENNY, C. **Random number generators: an evaluation and comparison of random.org and some commonly used generators**. Dublin, 2005. Disponível em: <https://www.random.org/analysis/Analysis2005.pdf>.
- 145 METROPOLIS, N.; ULAM, S. M. The Monte Carlo method. **Journal of the American Statistical Association**, v. 44, n. 247, p. 335–341, 1949.
- 146 BRATLEY, P.; FOX, B.; SCHRAGE, L. **A guide to simulation**. New York: Springer Verlag, 1987.
- 147 HUANG, A. **Hacking the Xbox: an introduction to reverse engineering**. San Francisco: No Starch Press, 2003.
- 148 LUBY, M.; LUBY, M. **Pseudorandomness and cryptographic applications**. Princeton: Princeton University Press, 1996. (Princeton computer science notes).
- 149 HELLEKALEK, P. Good random number generators are (not so) easy to find. **Mathematics and Computers in Simulation**, v. 46, n. 5–6, p. 485–505, 1998.
- 150 STRZALKO, J. et al. **Dynamics of gambling: origins of randomness in mechanical systems**. Berlin: Springer Verlag, 2009. (Lecture Notes in Physics).
- 151 SILVERMAN, M.; STRANGE, W.; SILVERMAN, C. R.; LIPSCOMBE, T. Tests of alpha-, beta-, and electron capture decays for randomness. **Physics Letters A**, v. 262, n. 4–5, p. 265–273, 1999.
- 152 WALKER, J. **HotBits: genuine random numbers, generated by radioactive decay**. 2006. Disponível em: www.fourmilab.ch/hotbits. Acesso em: 7.10.2016.
- 153 BUCCI, M. et al. A high-speed oscillator-based truly random number source for cryptographic applications on a smart card IC. **IEEE Transactions on Computers**, v. 52, p. 403–409, 2003.
- 154 HOLMAN, W. T.; CONNELLY, J. A.; DOWLATABADI, A. B. An integrated analog/digital random noise source. **IEEE Transactions on Circuits and Systems I: fundamental theory and applications**, v. 44, n. 6, p. 521–528, 1997.
- 155 HAAHR, M. **Random.org: introduction to randomness and random numbers**. 1999. Disponível em: <https://cran.r-project.org/web/packages/random/vignettes/random-essay.pdf>. Acesso em: 10.10.2016.
- 156 SILICON GRAPHICS. Landon Curt Nole and Robert G. Mende and Sanjeev Sisodiya. **Method for seeding a pseudo-random number generator with a cryptographic hash of a digitization of a chaotic system**. 1998. US 5732138A.
- 157 REN, M. et al. Quantum random-number generator based on a photon-number-resolving detector. **Physical Review A**, v. 83, p. 023820, 2011.
- 158 XU, F. et al. Ultrafast quantum random number generation based on quantum phase fluctuations. **Optics Express**, v. 20, n. 11, p. 12366–12377, 2012.

- 159 DIACONIS, P.; HOLMES, S.; MONTGOMERY, R. Dynamical bias in the coin toss. **SIAM Reviews**, v. 49, n. 2, p. 211–235, 2007.
- 160 GOLDREICH, O. **A primer on pseudorandom generators**. Providence, RI, USA: American Mathematical Society, 2010. (University Lecture Series).
- 161 EICHENAUER, J.; LEHN, J. A non-linear congruential pseudo random number generator. **Statistische Hefte**, v. 27, n. 1, p. 315–326, 1986.
- 162 PEINADO, A.; FÚSTER-SABATER, A. Generation of pseudorandom binary sequences by means of linear feedback shift registers (lfsrs) with dynamic feedback. **Mathematical and Computer Modelling**, v. 57, n. 11-12, p. 2596–2604, 2013.
- 163 MATSUMOTO, M.; NISHIMURA, T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. **ACM Transactions on Modeling and Computer Simulation**, v. 8, p. 3–30, 1998.
- 164 Java™ PLATFORM STANDARD ED.7. **Documentação da classe java.util.Random**. 2016. Disponível em: <<https://docs.oracle.com/javase/7/docs/api/java/util/Random.html>>. Acesso em: 17.10.2017.
- 165 MARSAGLIA, G. Xorshift rngs. **Journal of Statistical Software**, v. 8, n. 14, p. 1–6, 2003.
- 166 MICROSOFT. **Descrição da função RAND no Excel 2007 e no Excel 2003**. 2007. Disponível em: <<https://support.microsoft.com/pt-br/help/828795/description-of-the-rand-function-in-excel>>. Acesso em: 17.10.2017.
- 167 PHP DOCUMENTATION. **Documentação da função lcg_value do (PHP 4, PHP 5, PHP 7)**. 2017. Disponível em: <<http://php.net/manual/en/function.lcg-value.php>>. Acesso em: 17.10.2017.
- 168 MARSAGLIA, G. Random numbers fall mainly in the planes. **Proceedings of the National Academy of Sciences**, v. 61, n. 1, p. 25–28, 1968. Disponível em: <<http://www.pnas.org/content/61/1/25.short>>.
- 169 MARKOWSKY, G. The sad history of random bits. **Journal of Cyber Security and Mobility**, v. 3, p. 1–24, 2014.
- 170 PASHLEY, P.; AMODEO, A. Generating random numbers. In: PETERSON, P.; BAKER, E.; MCGAW, B. (Ed.). **International Encyclopedia of Education**. 3. ed. Oxford: Elsevier, 2010. p. 184–189.
- 171 RUKHIN, A. et al. **A statistical test suite for random number generator for cryptographic applications**. Gaithersburg, MD, USA: NIST, 2001. Special publication 800-22.
- 172 ÁLVAREZ, G.; LI, S. Some basic cryptographic requirements for chaos-based cryptosystems. **International Journal of Bifurcation and Chaos**, v. 16, n. 8, p. 2129–2151, 2006.
- 173 ARROYO, D. **Framework for the analysis and design of encryption strategies based on discrete-time chaotic dynamical systems**. 2009. 14-23 p. Tese (Doutorado) — Universidad Politécnica de Madrid, Madrid, 2009.

- 174 MIN, L.; HU, K.; ZHANG, L.; ZHANG, Y. Study on pseudorandomness of some pseudorandom number generators with application. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL INTELLIGENCE AND SECURITY, 9th, CIS. **Proceedings...** Leshan, Sichuan Province, China: New York: IEEE, 2013. p. 569–574.
- 175 HU, H.; LIU, L.; DING, N. Pseudorandom sequence generator based on the chen chaotic system. **Computer Physics Communications**, v. 184, n. 3, p. 765–768, 2013.
- 176 ÖZTÜRK, İ.; KILIÇ, R. A novel method for producing pseudo random numbers from differential equation-based chaotic systems. **Nonlinear Dynamics**, v. 80, n. 3, p. 1147–1157, 2015.
- 177 TOMASSINI, M.; SIPPER, M.; PERRENOUD, M. On the generation of high-quality random numbers by two-dimensional cellular automata. **IEEE Transactions on Computers**, v. 49, n. 10, p. 1146–1151, 2000.
- 178 SPENCER, J. Pseudorandom bit generators from enhanced cellular automata. **Cellular Automata**, v. 10, n. 3–4, p. 295–317, 2015.
- 179 WOLFRAM. **Random Number Generation**. 2017. Disponível em: <http://reference.wolfram.com/language/tutorial/RandomNumberGeneration.html>. Acesso em: 17.10.2017.
- 180 HU, H.; DENG, Y.; LIU, L. Counteracting the dynamical degradation of digital chaos via hybrid control. **Communications in Nonlinear Science and Numerical Simulation**, v. 19, n. 6, p. 1970 – 1984, 2014.
- 181 MARSAGLIA, G. **The Marsaglia random number CDROM, with the DIEHARD battery of tests of randomness**. Florida, USA: The National Science Foundation, 1998. Disponível em: <http://www.stat.fsu.edu/pub/diehard>. Acesso em: 20.04.2013.
- 182 WALKER, J. **A pseudorandom number sequence test program**. 1998. Disponível em: <http://www.fourmilab.ch/random/>. Acesso em: 20.04.2013.
- 183 L'ECUYER, P.; SIMARD, R. Testu01: a c library for empirical testing of random number generators. **ACM Transactions on Mathematical Software**, v. 33, n. 4, p. 22:1–22:40, 2007.
- 184 BROWN, R. G. **DIEHARDER**. 2017. Disponível em: <http://webhome.phy.duke.edu/~rgb/General/dieharder.php>. Acesso em: 20.10.2017.
- 185 OTEO, J.; ROS, J. Double precision errors in the logistic map: statistical study and dynamical interpretation. **Physical Review E**, v. 76, n. 3, p. 036214, 2007.
- 186 MARIE, P.; TURCHETTI, G.; VAIENTI, S.; ZANLUNGO, F. Error distribution in randomly perturbed orbits. **Chaos: an interdisciplinary journal of nonlinear science**, v. 19, n. 4, p. 043118, 2009.
- 187 SHI, P. A relation on round-off error, attractor size and its dynamics in driven or coupled logistic map system. **Chaos: an interdisciplinary journal of nonlinear science**, v. 18, n. 1, p. 013122, 2008.

- 188 LI, S.; CHEN, G.; MOU, X. On the dynamical degradation of digital piecewise linear chaotic maps. **International Journal of Bifurcation and Chaos**, v. 15, n. 10, p. 3119–3151, 2005.
- 189 ARAKI, S.; MIYAZAKI, T.; UEHARA, S.; KAKIZAKI, K. A study on precision of pseudorandom number generators using the logistic map. In: INTERNATIONAL SYMPOSIUM ON INFORMATION THEORY AND ITS APPLICATIONS, ISITA, 2012, **Proceedings...** New York: IEEE, 2012. p. 740–744.
- 190 APFLOAT arbitrary precision library. 2014. High performance arbitrary precision arithmetic package for C++ and Java. Disponível em: <<http://www.apfloat.org/>>. Acesso em: 29.06.2017.
- 191 LEE, P.-H.; CHEN, Y.; PEI, S.-C.; CHEN, Y.-Y. Evidence of the correlation between positive lyapunov exponents and good chaotic random number sequences. **Computer Physics Communications**, v. 160, p. 187–203, 2004.
- 192 ALLIGOOD, K.; SAUER, T.; YORKE, J. **Chaos: an introduction to dynamical systems**. New York: Springer, 2012. (Textbooks in Mathematical Sciences).
- 193 COLLET, P.; ECKMANN, J. **Iterated maps on the interval as dynamical systems**. New York, NY, USA: Birkhäuser Boston, 2009. (Modern Birkhäuser Classics).
- 194 LUKE, S. **Implementação de Mersenne Twister em Java**. 2004. Disponível em: <<https://cs.gmu.edu/~sean/research/>>. Acesso em: 14.11.2017.
- 195 SHANNON, C. A mathematical theory of communication. **Bell System Technical Journal**, v. 27, n. 3, p. 379–423, 1948.
- 196 CSARDI, G.; NEPUSZ, T. The igraph software package for complex network research. **International Journal Complex Systems**, v. 1695, n. 5, p. 1–9, 2006.
- 197 LESKOVEC, J.; KREVL, A. **SNAP datasets: stanford large network dataset collection**. 2014. Disponível em: <<http://snap.stanford.edu/data>>. Acesso em: 05.09.2017.
- 198 Project Gutenberg (n. d). **Free ebooks by project Gutenberg**. 2017. Disponível em: <www.gutenberg.org>. Acesso em: 05.04.2017.
- 199 IH, W.; E, F.; MA, H. **Data mining: practical machine learning tools and techniques**. Burlington: Morgan Kaufmann, 2011.
- 200 JOHNSON, S. **Twitter vs. Google Plus: why social media marketing needs both**. 2013. Disponível em: <https://www.weidert.com/whole_brain_marketing_blog/bid/113084/twitter-vs-google-plus-why-social-media-marketing-needs-both>. Acesso em: 04.11.2017.
- 201 MCAULEY, J. J.; LESKOVEC, J. **Learning to discover social circles in ego networks**. In: NIPS. [S.l.: s.n.], 2012. v. 2012, p. 548–56.
- 202 BORNHOLDT, S.; SCHUSTER, H. **Handbook of graphs and networks**. New York: Wiley Online Library, 2003. v. 2.
- 203 DOROGOVTSSEV, S.; MENDES, J. **Evolution of networks: from biological nets to the internet and WWW**. Oxford: Oxford University Press, 2013.

- 204 HEYMANS, M.; SINGH, A. K. Deriving phylogenetic trees from the similarity analysis of metabolic pathways. **Bioinformatics**, v. 19, 2003.
- 205 BANERJEE, A. Structural distance and evolutionary relationship of networks. **Biosystems**, v. 107, n. 3, p. 186–196, 2012.
- 206 CHAE, L.; LEE, I.; SHIN, J.; RHEE, S. Y. Towards understanding how molecular networks evolve in plants. **Current opinion in plant biology**, v. 15, n. 2, p. 177–184, 2012.
- 207 CHAE, L.; KIM, T.; NILO-POYANCO, R.; RHEE, S. Y. Genomic signatures of specialized metabolism in plants. **Science**, v. 344, n. 6183, p. 510–513, 2014.
- 208 METABOLIC Dataset. CCNR/ICeNSA: Interdisciplinary Center for Network Sciences & Applications. 2000. Disponível em: <<http://www3.nd.edu/~networks/resources.htm>>. Acesso em: 05.09.2016.
- 209 COWAN, I. R.; TROUGHTON, J. H. The relative role of stomata in transpiration and assimilation. **Planta**, v. 97, n. 4, p. 325–336, 1971.
- 210 HETHERINGTON, A. M.; WOODWARD, F. I. The role of stomata in sensing and driving environmental change. **Nature**, v. 424, p. 901–908, 2003.
- 211 BEERLING, D.; WOODWARD, F. Changes in land plant function over the phanerozoic: reconstructions based on the fossil record. **Botanical Journal of the Linnean Society**, v. 124, n. 2, p. 137–153, 1997.
- 212 CAI, Z.; QI, X.; CAO, K. Response of stomatal characteristics and its plasticity to different light intensities in leaves of seven tropical woody seedlings. **Journal of Applied Ecology**, v. 15, n. 2, p. 201–204, 2004.
- 213 MURCHIE, E. H.; HORTON, P. Acclimation of photosynthesis to irradiance and spectral quality in british plant species: chlorophyll content, photosynthetic capacity and habitat preference. **Plant, Cell & Environment**, v. 20, n. 4, p. 438–448, 1997.
- 214 FLORINDO, J. B.; LANDINI, G.; FILHO, H. A.; BRUNO, O. M. Analysis of stomata distribution patterns for quantification of the foliar plasticity of tradescantia zebrina. **Journal of Physics Conference Series**, v. 633, p. 012113, 2015.
- 215 SILVA, N. R. da et al. Leaf epidermis images for robust identification of plants. **Scientific Reports**, v. 6, p. 25994, 2016.
- 216 FRANCO-SALVADOR, M.; ROSSO, P.; MONTES-Y-GÓMEZ, M. A systematic study of knowledge graph analysis for cross-language plagiarism detection. **Information Processing & Management**, v. 52, n. 4, p. 550–570, 2016.
- 217 LABBÉ, C.; LABBÉ, D. Duplicate and fake publications in the scientific literature: how many scigen papers in computer science?. **Scientometrics**, v. 94, n. 1, p. 379–396, 2013.
- 218 VACCA, J. R. **Computer forensics: computer crime scene investigation** (networking series). Rockland, MA, USA: Charles River Media, Inc., 2005.

- 219 STAMATATOS, E. A survey of modern authorship attribution methods. **Journal of the American Society for Information Science and Technology**, v. 60, n. 3, p. 538–556, 2009.
- 220 AMANCIO, D. R. Authorship recognition via fluctuation analysis of network topology and word intermittency. **Journal of Statistical Mechanics: theory and experiment**, v. 2015, n. 3, p. P03005, 2015.
- 221 BRENNAN, M.; AFROZ, S.; GREENSTADT, R. Adversarial stylometry: circumventing authorship recognition to preserve privacy and anonymity. **ACM Transactions on Information System Security**, v. 15, n. 3, p. 12:1–12:22, 2012.
- 222 BRENNAN, M. R.; GREENSTADT, R. **Practical attacks against authorship recognition techniques**. 2009. Disponível em: <https://www.cs.drexel.edu/~greenie/brennan_paper.pdf>. Acesso em: 16.11.2017.
- 223 AMANCIO, D. R. et al. Probing the statistical properties of unknown texts: application to the Voynich manuscript. **Plos One**, v. 8, n. 7, p. e67310, 07 2013.
- 224 SEGARRA, S.; EISEN, M.; RIBEIRO, A. Authorship attribution through function word adjacency networks. **IEEE Transactions on Signal Processing**, v. 63, n. 20, p. 5464–5478, 2015.
- 225 AMANCIO, D. R.; OLIVEIRA JUNIOR, O.; COSTA, L. da F. Structure–semantics interplay in complex networks and its effects on the predictability of similarity in texts. **Physica A**, v. 391, n. 18, p. 4406–4419, 2012.
- 226 ARUN, R.; SURESH, V.; MADHAVAN, C. E. V. Stopword graphs and authorship attribution in text corpora. In: IEEE INTERNATIONAL CONFERENCE ON SEMANTIC COMPUTING, ICSC’09, IEEE, 2009, Berkeley, CA. **Proceedings...** Washington: IEEE, 2009. p. 192–196.
- 227 SOLÉ, R. V.; COROMINAS-MURTRA, B.; VALVERDE, S.; STEELS, L. Language networks: their structure, function, and evolution. **Complexity**, v. 15, n. 6, p. 20–26, 2010.
- 228 AMANCIO, D. R.; NUNES, M. G. V.; OLIVEIRA JUNIOR, O. N.; COSTA, L. da F. Using complex networks concepts to assess approaches for citations in scientific papers. **Scientometrics**, v. 91, n. 3, p. 827–842, 2012.
- 229 MACHICAO, J. et al. **Authorship attribution based on Life-Like network automata**. 2016. Disponível em: <<http://arxiv.org/abs/1610.06498>>. Acesso em: 17.11.2017.
- 230 EBRAHIMPOUR, M. et al. Automated authorship attribution using advanced signal classification techniques. **Plos One**, v. 8, n. 2, p. e54998, 2013.
- 231 COLLINS, M. Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In: CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING ON ACL-2, EMNLP’02, 2002, Stroudsburg, **Proceedings...** Stroudsburg, PA, USA: Association for Computational Linguistics, 2002. v. 10, p. 1–8.

- 232 NAVIGLI, R. Word sense disambiguation: a survey. **ACM Computing Surveys**, v. 41, n. 2, p. 10:1–10:69, 2009.
- 233 TOMAN, M.; TESAR, R.; JEZEK, K. Influence of word normalization on text classification. **Proceedings of InSciT**, v. 4, p. 354–358, 2006.
- 234 MIRANDA, G. H. B.; MACHICAO, J.; BRUNO, O. M. **An optimized shape descriptor based on structural properties of networks**. 2017. Disponível em: <https://arxiv.org/abs/1711.05104>. Acesso em: 17.11.2017.

Apêndices

APÊNDICE A – TESTES DE ALEATORIEDADE

A.1 DIEHARD

Tabela 27 – Descrição da bateria de testes de aleatoriedade DIEHARD.

Testes Diehard	Descrição
<i>BirthdaySpacings</i>	Escolhe m pontos aleatórios (aniversários) em um intervalo de n -bits (anos), por defeito usar $m = 2^9$ e $n = 2^{24}$. Compara a distribuição do espaçamento entre os valores para uma distribuição de Poisson.
<i>OverlappingPermutations</i>	Analisa sequências de cinco números consecutivos. Já que existem $5!$ possíveis combinações, estas devem ser normalmente distribuídas.
<i>Ranks31x31 matrices</i>	Matrizes binárias aleatórias de 31×31 são criadas, cujas linhas correspondem aos 31 bits mais à esquerda de cada inteiro de 32-bits. Depois, calcula-se o grau de cada linha (<i>ranking</i>). Logo, determina as frequências dos k -ésimos colocados
<i>Ranks32x32 matrices</i>	Matrizes 32×32 (ver teste anterior).
<i>Ranks6x8 matrices</i>	Matrizes 6×8 , segura 6 números de 32-bits e escolhe 8 bits. (ver teste anterior).
<i>Monkey20bitsWords</i>	Conta o número de palavras de 20 bits sobrepostas. Conta o número de palavras sucessivas de 20 caracteres sobrepostas. O número de palavras faltando deve ser normalmente distribuído.
<i>OPSO</i>	Forma palavras sobrepostas de 2 letras de um alfabeto de 1024 letras. As palavras faltantes devem ser próximas de uma distribuição normal.
<i>OQSO</i>	Forma Palavras sobrepostas de 4 letras de um alfabeto de 32 letras (ver OPSO)
<i>DNA</i>	Forma Palavras sobrepostas de 10 letras de um alfabeto de 4 letras: C,G,A,T, (ver OPSO).
<i>Count1sStream</i>	Usa as probabilidades do número de 1 em cada byte. Determina quantas vezes as letras de 8 bits e as palavras de 5 letras aparecem em um fluxo de bytes.
<i>Count1sSpecific</i>	Usa as probabilidades do número de 1 em bytes específicos (ver teste anterior).
<i>ParkingLot</i>	Aleatoriamente, estacione carros em um quadrado de 100×100 , e se ocorrer um “acidente”, não estacione o carro. Continue assim, até 12000 carros e compare quantas conseguiram ser estacionadas com sucesso.
<i>MinimumDistance</i>	Mede a distância quadrática entre 8000 pontos aleatórios em um espaço de $10^4 \times 10^4$. Essas distâncias devem ser distribuídas exponencialmente. O p -valor é dado pela variação do valor esperado.
<i>RandomSpheres</i>	Escolha 4000 pontos aleatórios colocados em um cubo de $10^3 \times 10^3$. Em cada ponto centre uma esfera grande o suficiente para tocar apenas o cubo mais próximo. O raio mínimo é encontrado e este teste é repetido 20 vezes. Esses raios devem ser exponencialmente distribuídos. O p -valor será determinado com base em quão longe os resultados são esperados.
<i>Squeeze</i>	Compara o número de iterações para reduzir um ponto flutuante em $[0, 1]$, começando por $k = 2^31$ até chegar a $k = 1$. Repita isso 10^5 vezes. O número de valores necessários deve seguir uma certa distribuição.
<i>OverlappingSums</i>	Adicione 100 valores consecutivos em $[0, 1]$, incrementando um elemento de cada vez. As somas devem ser normalmente distribuídas.
<i>Runs (up/down)</i>	Itere 10^4 valores, e conte o número de “corridas” consecutivas, onde os elementos estão aumentando em valor ou diminuindo em valor consecutivamente. Uma corrida é a sequência ininterrupta de bits idênticos. As contagens devem seguir uma certa distribuição.
<i>Craps (wins/throws/game)</i>	Reproduz o jogo de <i>craps</i> (apostas) 2×10^5 vezes, contando o número de vitórias, perdas e o número de jogadas por jogo. Cada contagem deve seguir uma distribuição normal.

Fonte: Adaptada de MARSAGLIA (181).

A.2 NIST

A bateria de testes NIST (171) possui 15 testes detalhadas na Tabela 28.

Tabela 28 – Descrição da bateria de testes de aleatoriedade NIST.

NIST testes	Descrição
<i>Frequency</i>	Frequência de 1s e 0s.
<i>BlockFrequency</i> ($m = 128$)	Frequência de 1s e 0s dentro de um bloco de tamanho m .
<i>CumulativeSums</i>	A soma cumulativa dos dígitos ajustados (bit 0=-1, bit 1=+1) na sequência deve ser próximo de zero.
<i>Runs</i>	Grande (pequeno) número de corridas indica que a oscilação é muito rápida (muito lenta). Oscilações são as mudanças entre corridas de 0's e corridas de 1's.
<i>LongestRun</i>	Analisa o tamanho das corridas mais longas de 1's contidas dentro de um bloco de tamanho m .
<i>Rank</i>	Verifica a dependência linear entre subsequências da sequência original.
<i>FFT</i>	Procura padrões repetitivos usando por meio das alturas dos picos na Transformação Discreta de Fourier em uma sequência.
<i>Non-overlappingTemplate</i>	Conta a frequência de padrões pré-estabelecidos (por exemplo: "000000001", "000011101") em uma janela de m -bits. Há um total de 148 padrões de 9-bits. Se o padrão não for encontrado, a janela é deslizada em um bit, caso contrário continua da posição passada, evitando sobrepor janelas.
<i>OverlappingTemplate</i>	Similar ao teste anterior, exceto que permite sobreposição ao deslizar as janelas de busca.
<i>Universal</i>	Determina se a sequência é comprimível. Conta o número de bits entre padrões (subsequências de bits).
<i>ApproximatedEntropy</i> ($m = 10$)	A distribuição conjunta da frequência de todos os padrões possíveis de m -bits e $m + 1$ -bits sobrepostos em toda a sequência deve ser uniforme.
<i>RandomExcursions</i>	Calcula a distribuição do número de ciclos com k visitas exatamente em uma caminhada aleatória de soma cumulativa (bit 0=-1, bit 1=+1) para cada um dos estados: -4, -3, -2, -1 e +1, +2, +3, +4. Um ciclo é o número de passos desde que começa até que retornam à origem.
<i>RandomExcursions</i> (variante)	Similar ao teste anterior. Porém, calcula o número total de vezes que um estado particular é visitado na caminhada aleatória.
<i>Serial</i> ($m = 16$)	Similar ao teste <i>ApproximatedEntropy</i> . A frequência de todos os padrões possíveis de m -bits com sobreposições deve ser uniforme.
<i>LinearComplexity</i> ($M = 500$)	= Definida pelo comprimento do registro de deslocamento de retorno linear (LFSR) mais curto que possa gerar uma sequência. Foi observado que, este teste é o mais custoso computacionalmente, comparado aos demais testes.

Fonte: Adaptada de NIST (171); KENNY (144, p. 17).

APÊNDICE B – K -MAPA DA TENDA

B.1 k -mapa da tenda

A abordagem das profundezas do caos com base na análise dos k -dígitos de precisão também pode ser estendida a outros mapas caóticos no intervalo da unidade. Como já foi introduzido na Seção 2.3.2, o mapa da tenda dada pela Equação (2.6) é um sistema dinâmico unidimensional no espaço de fase $x_t \in [0, 1]$ e cujo parâmetro $\delta \in [0, 2]$. Portanto, ampliamos as definições dadas na Seção 4.2. Assim, dada uma órbita do mapa de tenda original $\mathcal{O}(\delta, x_0) = \{x_0, \dots, x_t\}$ e a órbita resultante $\mathcal{O}^k(\delta, x_0) = \{x_0^k, \dots, x_t^k\}$, podemos generalizar a Equação (4.2).

De forma semelhante às experiências anteriores, aqui também analisamos as propriedades dinâmicas globais do k -mapa da tenda para fins dos PRNGs, pelo que limitamos os experimentos ao regime caótico do mapa k -mapa da tenda usando o parâmetro de controle $\delta = 2$. Embora, esta seção não pretenda abranger todas as ferramentas de visualização como foi feito com o k -mapa logístico, são visualizadas as mais importantes.

B.1.1 Diagrama de bifurcação do k -mapa da tenda

A Figura 56 mostra o diagrama de bifurcação e corresponde esqueleto para o k -mapa da tenda. Observamos um fenômeno semelhante a relatado no k -mapa logístico (Figura 17). Portanto, à medida que os k -dígitos são aumentados, as órbitas começam a ser torcidas rapidamente, as bifurcações onde acontece periodicidade diminuem e as regiões estáveis permanecem. Além disso, a bifurcação de esqueleto sobreposta também mostra um comportamento de padrão de zigzague que descreve um padrão mais irregular.

B.1.2 Diagrama de Poincaré do k -mapa da tenda

Os diagramas de fase do k -mapa da tenda são mostrados na Figura 57 (de cima para baixo) correspondente ao k_0 a k_5 . Percebemos que o k -mapa da tenda se comporta de forma semelhante ao mapa logístico (ver Figura 22) em termos de melhora da aleatoriedade quando k é aumentado (133). Aqui, o mapa da tenda original (k_0), do qual recebeu seu nome, parece zigzaguear (k_1), até se tornar mais difuso. No entanto, alguns padrões difusos ainda permanecem no diagrama de fases tridimensional de k_4 e o espaço de fase começa a ser mais aleatorizado a partir de $k \geq 5$.

B.2 PRNG baseado no k -mapa da tenda

Também usamos o k -mapa da tenda para propósitos de um PRNG. Semelhante ao mapa de k -mapa logístico. Para isso, foi restringido o espaço de fase ao intervalo $]0, 1[$

Figura 56 – Diagrama de bifurcação do k_0 , k_1 , k_2 e k_3 -mapa da tenda. O eixo horizontal mostra $\mu \in [1, 2]$ (passos de 0.001), e o eixo vertical representa os valores de longo prazo do mapa de k , a partir da mesma condição inicial. Calculamos 10^5 iterações e descartamos os 200 primeiros valores do tempo transiente. As curvas sobrepostas $P_{n+1}(\delta)$ correspondem ao esqueleto da bifurcação (linhas escuras) em relação ao mapa da tenda, em que $P_{n+1}(\delta) = f(P_n, \delta)$ com $x_c = 0.5$.

Fonte: MACHICAO; BRUNO. (133)

e o parâmetro de controle para $\delta \rightarrow 2$. Assim, para gerar números, discretizam-se os valores x_t^k do k -mapa da tenda em inteiros de 32 bits conforme definido na Equação (4.4). Configuramos a mesma configuração para os experimentos usando os testes de suite DIEHARD (181) e NIST (171). Tabela 29 e Tabela 30 relataram o número de arquivos que passaram os testes de pseudo-aleatoriedade para ambos os conjuntos, respectivamente.

Em relação a essas tabelas, observamos uma falha parcial no teste de DIEHARD a partir do k_0 a k_5 -mapa da tenda. Mais precisamente, as matrizes de sub-testes *Ranks6x8*, *OQSO*, *CountIsStream* e *CountIs*. Não obstante $k \geq 6$, a maioria das 100 amostras foram aprovadas com sucesso. Em contraste com o k -mapa logístico, o k -mapa da tenda passou com sucesso o teste NIST para todos os parâmetros k .

Figura 57 – Diagramas de fase para k_0 , k_1 , k_2 , k_3 , k_4 e k_5 usando $\delta = 2$. Os diagramas de fases bidimensionais e tridimensionais são mostrados da esquerda para a direita, respectivamente. Os eixos horizontal e vertical mostram o espaço de fase de x_t em função de x_{t+1} e x_{t+2} . Cada órbita contém 10^4 pontos com condição inicial aleatória. Os primeiros 200 foram descartados (período transiente).

Fonte: MACHICAO; BRUNO. (133)

Tabela 29 – Resultados dos testes de aleatoriedade DIEHARD para 100 arquivos gerados pelos diferentes k -mapa da tenda com sementes aleatórias. Cada coluna indica o número de sequências (arquivos) que passaram. Testes que falharam são mostrados em cinza. Todos os testes passaram usando o intervalo $0.0001 < p\text{-valor} < 0.9999$.

Testes DIEHARD	k_0	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9
<i>BirthdaySpacings</i>	100	100	100	100	100	100	100	99	100	100
<i>OverlappingPermutations</i>	99	97	100	99	96	98	97	98	100	97
<i>Ranks31x31 matrices</i>	100	100	100	100	100	100	100	100	100	100
<i>Ranks32x32 matrices</i>	100	100	100	100	100	100	100	100	100	100
<i>Ranks6x8 matrices</i>	0	0	0	100	100	100	100	100	100	100
<i>Monkey20bitsWords</i>	100	100	100	100	100	100	100	100	100	100
<i>OPSO</i>	100	100	100	100	100	100	100	100	100	100
<i>OQSO</i>	0	0	0	100	100	100	100	100	100	100
<i>DNA</i>	100	100	100	100	100	100	100	100	100	100
<i>Count1sStream</i>	0	0	0	0	100	100	100	100	100	100
<i>Count1sSpecific</i>	0	0	0	0	0	0	100	100	100	100
<i>ParkingLot</i>	100	100	100	100	100	100	100	100	100	100
<i>MinimumDistance</i>	81	95	94	100	98	98	97	99	98	99
<i>RandomSpheres</i>	100	100	100	100	100	100	100	100	100	100
<i>Squeeze</i>	100	100	100	100	100	100	100	100	100	100
<i>OverlappingSums</i>	100	100	100	100	100	100	100	100	100	100
<i>Runs (up)</i>	100	100	100	100	100	100	100	100	100	100
<i>Runs (down)</i>	100	100	100	100	100	100	100	100	100	100
<i>Craps (wins)</i>	100	100	100	100	100	100	100	100	100	100
<i>Craps (throws/game)</i>	100	100	100	100	100	100	100	100	100	100

Fonte: Adaptada de MACHICAO; BRUNO. (133)

Tabela 30 – Resultados dos testes de aleatoriedade NIST para 100 arquivos gerados pelos diferentes k -mapa da tenda, com sementes aleatórias. Cada coluna indica o número de sequências (arquivos) que passaram. Os testes falhados são mostrados em cinza. Todos os testes passaram para o nível de significância $\alpha = 0.01$.

Testes NIST	k_0	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8	k_9
<i>Frequency</i>	98	99	99	99	99	99	98	99	99	99
<i>BlockFrequency</i> ($m = 128$)	100	99	99	100	99	100	99	100	99	99
<i>CumulativeSums</i>										
<i>Forward sums</i>	98	99	99	98	99	99	98	99	99	99
<i>Reverse sums</i>	98	99	99	99	99	99	99	99	99	99
<i>Runs</i>	79	92	98	99	99	99	99	99	99	99
<i>LongestRun</i>	97	99	99	99	99	99	99	99	99	99
<i>Rank</i>	99	99	99	99	99	99	99	99	100	99
<i>FFT</i>	99	99	99	99	99	99	99	99	98	99
<i>Non-overlappingTemplate</i>										
000000001	75	97	99	99	99	99	99	99	98	99
000000011	89	97	99	100	99	99	98	99	99	99
000000101	92	98	99	99	100	99	100	98	99	99
000000111	91	99	99	99	99	99	99	99	99	99
000001001	92	99	99	99	99	99	100	99	99	99
000001011	93	99	99	99	99	99	99	99	99	99
000001101	95	99	99	99	99	100	99	99	99	99
000001111	90	98	99	99	99	99	99	99	99	99
000010001	93	100	99	99	99	98	99	99	99	99
000010011	93	99	98	99	99	99	99	99	99	99
000010101	94	99	99	99	99	99	99	99	99	100
000010111	93	99	99	99	99	99	99	99	99	99
000011001	93	99	99	99	99	99	100	100	99	99
000011011	94	98	99	99	99	99	99	99	99	99
000011101	93	98	99	99	99	99	99	99	99	99
<i>OverlappingTemplate</i>	62	91	99	99	99	99	99	99	99	99
<i>Universal</i>	97	98	99	99	99	99	99	99	98	99
<i>ApproxEntropy</i> ($m = 10$)	82	99	99	99	99	99	99	99	99	99
<i>RandomExcursions</i>										
$x = -4$	98	99	99	98	99	100	98	98	99	99
$x = -3$	100	98	99	98	99	99	99	99	99	99
$x = -2$	98	99	99	99	99	99	99	99	100	99
$x = -1$	99	99	99	99	98	99	99	99	99	99
$x = 1$	99	99	99	99	99	98	100	99	99	99
$x = 2$	100	98	99	99	99	100	99	99	98	99
$x = 3$	99	99	99	99	99	100	99	99	99	99
$x = 4$	98	99	98	99	99	99	99	99	99	99
<i>RandExcursVar</i>										
$x = -9$	99	100	99	99	100	99	99	99	99	98
$x = -8$	99	99	99	100	100	99	100	99	99	98
$x = -7$	99	99	99	100	99	99	99	100	99	98
$x = -6$	99	99	100	100	100	99	99	99	98	99
$x = -5$	99	99	99	100	100	99	99	99	99	98
$x = -4$	99	99	99	99	99	99	99	99	100	99
$x = -3$	99	99	100	99	100	99	99	100	99	99
$x = -2$	99	99	100	100	99	99	100	99	99	99
$x = -1$	99	99	99	99	99	99	99	99	98	99
$x = 1$	99	99	98	99	99	99	98	100	98	99
$x = 2$	98	99	98	99	99	99	98	99	98	99
$x = 3$	98	99	99	99	99	100	98	99	99	99
$x = 4$	99	99	98	99	98	99	99	99	99	99
$x = 5$	98	99	98	99	98	99	98	99	99	99
$x = 6$	98	99	99	99	99	99	99	99	99	99
$x = 7$	99	100	100	99	99	99	98	99	99	99
$x = 8$	99	99	99	99	99	99	99	99	99	99
$x = 9$	99	99	99	99	99	99	98	99	99	99
<i>Serial</i> ($m = 16$)										
<i>Serial 1</i>	99	99	99	99	99	99	99	99	99	99
<i>Serial 2</i>	99	99	99	99	100	99	99	99	99	99
<i>LinearComplexity</i> ($M = 500$)	99	99	99	99	99	100	99	99	98	99

Fonte: Adaptada de MACHICAO; BRUNO. (133)

APÊNDICE C – LISTA DE LIVROS E AUTORES DA BASE GUTENBERG

C.1 Lista de livros e autores usados para a validação

- Author 1 - Arthur Conan Doyle (1859-1930)
 - Book 1: Uncle Bernac - A Memory of the Empire
 - Book 2: The Tragedy of the Korosko
 - Book 3: The Valley of Fear
 - Book 4: The War in South Africa
 - Book 5: Through the Magic Door
- Author 2 - Bram Stoker (1847-1912)
 - Book 1: Dracula's Guest
 - Book 2: The Jewel Of Seven Stars
 - Book 3: The Lady of the Shroud
 - Book 4: Lair of the White Worm
 - Book 5: The Man
- Author 3 - Charles Darwin (1809-1882)
 - Book 1: Coral Reefs
 - Book 2: The Expression of the Emotions in Man and Animals
 - Book 3: Geological Observations on South America
 - Book 4: The Different Forms of Flowers on Plants of the Same Species
 - Book 5: Volcanic Islands
- Author 4 - Charles Dickens (1812-1870)
 - Book 1: American Notes
 - Book 2: A Tale of Two Cities
 - Book 3: Hard Times
 - Book 4: Great Expectations
 - Book 5: Barnaby Rudge: A Tale of the Riots of Eighty
- Author 5 - Thomas Hardy (1840-1928)
 - Book 1: A Changed Man; and Other Tales
 - Book 2: A Pair of Blue Eyes
 - Book 3: Far from the Madding Crowd
 - Book 4: Jude the Obscure
 - Book 5: The Hand of Ethelberta
- Author 6 - Pelham Grenville Wodehouse (1881-1975)
 - Book 1: My Man Jeeves
 - Book 2: Tales of St. Austin's
 - Book 3: The Adventures of Sally

- Book 4: The Clicking of Cuthbert
- Book 5: The Man with Two Left Feet
- Author 7 - Alan Poe (1809-1849)
 - Book 1: The Works of Edgar Allan Poe (Volume 1)
 - Book 2: The Works of Edgar Allan Poe (Volume 2)
 - Book 3: The Works of Edgar Allan Poe (Volume 3)
 - Book 4: The Works of Edgar Allan Poe (Volume 4)
 - Book 5: The Works of Edgar Allan Poe (Volume 5)
- Author 8 - Hector Hugh Munro (Saki) (1870-1916)
 - Book 1: Beasts and Super Beasts
 - Book 2: The Chronicles of Clovis
 - Book 3: The Toys of Peace
 - Book 4: When William Came
 - Book 5: The Unbearable Bassington

C.2 Lista de autores usados para a seleção da regra

- Author 9 - Herman Melville (1819-1891)
 - Book 1: Moby Dick; Or, The Whale
 - Book 2: The Piazza Tales
 - Book 3: The Confidence-Man: His Masquerade
 - Book 4: Typee: A Romance of the South Seas
 - Book 5: White Jacket; Or, The World on a Man-of-War
- Author 10 - Zane Grey (1872-1939)
 - Book 1: The Lone Star Ranger: A Romance of the Border
 - Book 2: Riders of the Purple Sage
 - Book 3: The Mysterious Rider
 - Book 4: To the Last Man
 - Book 5: The Call of the Canyon
- Author 11 - Andrew Lang (1844-1912)
 - Book 1: The Blue Fairy Book
 - Book 2: The Arabian Nights Entertainments
 - Book 3: The Yellow Fairy Book
 - Book 4: The Violet Fairy Book
 - Book 5: The Pink Fairy Book
- Author 12 - Richard Harding Davis (1864-1916)
 - Book 1: Real Soldiers of Fortune

- Book 2: Notes of a War Correspondent
- Book 3: Cinderella, and Other Stories
- Book 4: The Congo and Coasts of Africa
- Book 5: Soldiers of Fortune
- Author 13 - Henry James (1843-1916)
 - Book 1: The Turn of the Screw
 - Book 2: The Portrait of a Lady - Volume 1
 - Book 3: The Ambassadors
 - Book 4: The Real Thing and Other Tales
 - Book 5: The American
- Author 14 - B. M. Bower (1871-1940)
 - Book 1: Cabin Fever
 - Book 2: The Trail of the White Mule
 - Book 3: Lonesome Land
 - Book 4: The Lookout Man
 - Book 5: The Long Shadow
- Author 15 - Washington Irving (1783-1859)
 - Book 1: The Sketch-Book of Geoffrey Crayon
 - Book 2: Knickerbocker's History of New York, Complete
 - Book 3: Chronicle of the Conquest of Granada, from the mss. of Fray Antonio Agapida
 - Book 4: The Alhambra
 - Book 5: Tales of a Traveller
- Author 16 - Herbert George Wells (1866-1946)
 - Book 1: The War of the Worlds
 - Book 2: A Short History of the World
 - Book 3: The First Men in the Moon
 - Book 4: Tales of Space and Time
 - Book 5: The World Set Free
- Author 17 - Horatio Alger (1832-1899)
 - Book 1: Ragged Dick, Or, Street Life in New York with the Boot-Blacks
 - Book 2: Adrift in New York: Tom and Florence Braving the World
 - Book 3: Brave and Bold; Or, The Fortunes of Robert Rushton
 - Book 4: Fame and Fortune; or, The Progress of Richard Hunter
 - Book 5: The Errand Boy; Or, How Phil Brent Won Success

- Author 18 - Jane Austen (1775-1817)
 - Book 1: Pride and Prejudice
 - Book 2: Emma
 - Book 3: Sense and Sensibility
 - Book 4: Persuasion
 - Book 5: Mansfield Park
- Author 19 - Mark Twain (1835-1910)
 - Book 1: Adventures of Huckleberry Finn
 - Book 2: The Adventures of Tom Sawyer
 - Book 3: The Prince and the Pauper
 - Book 4: A Connecticut Yankee in King Arthur's Court
 - Book 5: Roughing It
- Author 20 - Nathaniel Hawthorne (1804-1864)
 - Book 1: The Scarlet Letter
 - Book 2: Mosses from an Old Manse, and Other Stories
 - Book 3: The House of the Seven Gables
 - Book 4: Twice Told Tales
 - Book 5: The Blithedale Romance