UNIVERSIDADE FEDERAL DO PIAUÍ

CENTRO DE CIÊNCIAS DA NATUREZA

DEPARTAMENTO DE FÍSICA

PROGRAMA DE PÓS-GRADUAÇÃO EM FÍSICA

Uma Abordagem de Técnicas de *Machine Learning* Não Supervisionadas para Transições de Fase Clássicas

Teresina/2022

Danyella Oliveira de Carvalho

Uma Abordagem de Técnicas de *Machine Learning* Não Supervisionadas para Transições de Fase Clássicas

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Física como parte dos requisitos necessários para a obtenção do título de Mestra em Física.

Orientador:

José Pimentel de Lima

Coorientador:
Natanael de Carvalho Costa

Teresina-PI

Março, 2022

FICHA CATALOGRÁFICA
Universidade Federal do Piauí
Sistema de Bibliotecas da UFPI – SIBi/UFPI
Biblioteca Setorial do CCN

C331a Carvalho, Danyella Oliveira de.
Uma abordagem de técnicas de *Machine Learning* não supervisionadas para transições de fase clássicas. / Danyella Oliveira de Carvalho. – 2022.
72 f.

Dissertação (Mestrado) – Universidade Federal do Piauí, Centro de Ciências da Natureza, Pós-Graduação em Física, Teresina, 2022.
"Orientador: Prof. Dr. José Pimentel de Lima".
Coorientador: Prof. Dr. Natanael de Carvalho Costa

1. Transição de Fase. 2. Modelo de Potts. 3. *Machine Learning*. I. Lima, José Pimentel de. II.Título.

CDD 530.13

Bibliotecária: Caryne Maria da Silva Gomes. CRB/3-1461

Dissertação de Mestrado sob o título *"Uma Abordagem de Técnicas de Machine Learning Não Supervisionadas para Transições de Fase Clássicas"* defendida por Danyella Oliveira de Carvalho em 30 de Março de 2022, em Teresina - Piauí, com banca examinadora constituída pelos professores:

Prof. Dr. José Pimentel de Lima (Orientador)
Universidade Federal do Piauí - UFPI

Prof. Dr. Natanael de Carvalho Costa (Coorientador)
Universidade Federal do Rio de Janeiro - UFRJ

Prof. Dr. Tiago Mendes Santos
University of Augsburg

Prof. Dr. Eduardo da Costa Girão
Universidade Federal do Piauí - UFPI

Prof. Dr. Francisco Welington de Sousa Lima (Suplente)
Universidade Federal do Piauí - UFPI

Agradecimentos

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo apoio e financiamento deste projeto.

À Universidade Federal do Piauí (UFPI) e ao Programa de Pós-Graduação em Física, pela estrutura fornecida.

Ao professor José Pimentel de Lima, pelo apoio e solicitude.

Ao professor Natanael de Carvalho Costa, pelos ensinamentos, conselhos e incentivo.

Ao colega Andrea Tirelli, pela colaboração e participação nas resoluções deste trabalho.

Meu sinceros agradecimentos.

Resumo

Machine Learning (ML), ou Aprendizado de Máquina, é um subcampo da Inteligência Artificial, constituído por classes de algoritmos de aprendizagem, que operam de forma tal que se possa extrair informação e "aprender" de uma grande quantidade de dados. Estes algoritmos têm sido incorporados no âmbito das ciências físicas como auxiliares na identificação de padrões, como, por exemplo, no estudo de transições de fase. Tais técnicas mostram-se como uma alternativa na compreensão de problemas complexos. Porém, apesar dos esforços, o limite de aplicação dessas técnicas ainda não é inteiramente claro. Em vista disso, investigamos neste trabalho as transições de fase do modelo de Potts q-estados, nos casos $q = 3$, 4 e 5, utilizando técnicas de ML não supervisionadas, a saber, *Principal Component Analysis* (PCA), *K-Means Clustering* e *Topological Data Analysis* (TDA). As duas primeiras são técnicas de redução de dimensionalidade e clusterização, respectivamente, enquanto a última usa propriedades de invariantes topológicos para separar dados semelhantes. Já a construção da base de dados se dá através de simulações de Monte Carlo, em faixas de temperaturas que contêm as temperaturas críticas do modelo. A análise desse *toy model* nos permite compreender detalhadamente essas técnicas no âmbito de Matéria Condensada e Mecânica Estatística. Utilizando a técnica PCA, foram identificados os pontos críticos com boa precisão em todos os casos ($q = 3$, 4 e 5), através de parâmetros construídos a partir das componentes principais. Além disso, observamos que a formação de *clusters* para temperaturas abaixo da temperatura crítica nos fornece informações sobre as simetrias da Hamiltoniana. Também foram encontrados resultados concordantes para os valores de expoentes críticos, porém com fortes efeitos de tamanho finito. Continuando, usamos a técnica *K-Means* para analisar o comportamento dos *clusters* obtidos através de PCA sob variação de temperatura. Esta análise retornou os pontos críticos, numa aproximação para o limite termodinâmico, com erro relativo inferior a 1%, em comparação com os valores exatos do modelo de Potts. Por fim, com o método TDA, obtivemos os pontos críticos com grande precisão, além de notar que esta técnica possui poucos efeitos de tamanho finito. Em resumo, obtivemos os pontos críticos com boa precisão usando os três métodos, e esperamos que esse estudo detalhado possa esclarecer as características particulares das técnicas aqui expostas, facilitando eventuais utilizações em problemas mais desafiadores no futuro.

Palavras-Chave: *machine learning*, transições de fase, modelo de Potts

Abstract

Machine Learning (ML) is a subfield of Artificial Intelligence, consisting of a class of algorithms that are able to extract information – i.e. "learning" – from a large amount of data. These algorithms have been incorporated in the physical sciences as supporting tools to identify patterns, for instance, in the study of phase transitions. Indeed, such techniques are placed as promising methods to understand complex systems. Yet, despite the great efforts, the limits of their applications are not entirely clear. In view of this, in this work we investigate the phase transitions of the q-states Potts model, in the cases q = 3, 4, and 5, using unsupervised ML techniques, namely, the Principal Component Analysis (PCA), the K-Means Clustering and the Topological Data Analysis (TDA). The first two are concerned with dimensionality reduction and clustering techniques, respectively, while the last one uses topological invariant properties to separate data by similarity. The dataset is generated through Monte Carlo simulations, in ranges of temperature that contain the critical temperatures of the model. The analysis of such a toy model allows us to understand the details about the applications of these techniques within the scope of Condensed Matter Physics and Statistical Mechanics. Using the PCA technique, the critical points are identified with good precision for all cases (q = 3, 4 and 5), through parameters constructed from the principal components. In addition, we notice that the formation of clusters at low-temperatures provides information about the symmetries of the Hamiltonian. Also, we investigate the critical exponents, whose results are in rough agreement with the exact ones, but with substantial finite-size effects. Moreover, we use the K-Means technique to analyze the behavior of the clusters obtained through PCA, under temperature variation. This analysis returned the critical points, in an approximation to the thermodynamic limit, with a relative error of less than 1%, compared to the exact values of the Potts model. Finally, with the TDA method, we obtain the critical points with very good precision, besides the fact that this technique has a weak dependency on the system size. In summary, the critical points are obtained with good precision using these three methods, and we expect that this detailed study may shed light on the features of these techniques when dealing with phase transitions, therefore paving the way to their usage in more challenging problems.

Keywords: machine learning, phase transitions, Potts model

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

1.1 *Machine Learning*

Machine Learning (ML), ou Aprendizado de Máquina, é um subcampo de estudo da Inteligência Artificial [1]. O termo, cunhado em 1959 por Arthur L. Samuel ($1901 - 1990$), cientista da computação americano, surgiu com a seguinte definição: "a habilidade dos computadores de aprender sem serem explicitamente programados". Na prática, ML pode ser entendido como classes de algoritmos que usam a própria "experiência" para melhorar seu desempenho ou fazer predições. Aqui, experiência significa aprender de dados prévios, conhecidos e categorizados, ou aprender de padrões. Tais algoritmos são baseados em cálculos matemáticos e estatísticos, em conceitos de probabilidade, métrica e topologia, transformações lineares e não lineares, dentre outros conceitos.

Essas classes de algoritmos podem ser divididas majoritariamente em *supervisionados* e *não supervisionados*. Os algoritmos supervisionados se dividem em duas grandes categorias: os algoritmos de classificação e os de regressão. Os de classificação são aqueles que operam baseados em um *label*, que são categorias às quais devem pertencer os dados de entrada. Por exemplo: uma rede neural treinada com milhares de imagens de pessoas com diferentes características e gêneros distintos deve ser capaz de classificar uma nova imagem como pertencente à categoria "Feminino" ou "Masculino"; esse tipo de algoritmo é amplamente usado em mídias sociais, como $Facebook$ ou $Instagram$. Um classificador *Naive Bayes* pode decidir se um dado *e-mail* é ou não *spam*: baseado numa coleção de *e-mails* anteriores que já foram ou não sinalizados como não desejados, os padrões são encontrados e, na presença de um novo *e-mail*, o algoritmo tomará uma decisão baseado em uma certa probabilidade. Já uma técnica de regressão, por outro lado, resulta em valores contínuos, como preço, salário ou temperatura, por exemplo. No entanto, essa é uma visão bastante simplista dos algoritmos, e nem sempre eles estarão limitados a tarefas restritas. Por exemplo: as Árvores de Decisão, ou as Máquinas de Vetores de suporte que pertencem à classe do classificadores, também podem ser utilizadas para regressão;

algoritmos de redução de dimensionalidade podem servir para a etapa de pré-processamento em um problema de classificação [2] e assim por diante. Com tantas possibilidades, as aplicações também são inúmeras: reconhecimento de imagens, análise de textos [3], recomendações de serviços *online*, previsões climáticas [4], classificação de imagens em diagnósticos médicos [5, 6], Processamento de Linguagem Natural [7] (que é um dos principais assuntos relacionados a *Deep Learning*), carros autônomos [8], entre outros exemplos. As próximas seções trazem uma visão geral a respeito das técnicas mais importantes de *machine learning*, para contextualização do leitor.

1.1.1 Modelos Supervisionados

Regressão Linear

Imagine um conjunto de dados usado para prever o preço de casas numa determinada região. Esses dados devem conter suas principais características, como tamanho do imóvel, quantidade de cômodos, localização, etc., além do preço de cada um dos imóveis (que serão os *targets*), enquanto as características compõem o escopo das variáveis independentes. Dessa forma, é possível fazer uma aproximação para o valor de um determinado imóvel baseado em suas características, comparando com as informações já conhecidas de vendas anteriores. A fórmula geral da regressão linear é do tipo

$$\hat{y} = a_1 x_1 + a_2 x_2 + \ldots a_n x_n + b \tag{1.1}$$

em que x_n denota a $n-$ésima variável. Para o caso de uma única variável, essa fórmula é simplificada para

$$\hat{y} = a_1 x_1 + b, \tag{1.2}$$

que é facilmente identificada como a função de uma reta cujo coeficiente de inclinação é a_1 e b o valor que intercepta o eixo y. A Figura 1.1 ilustra a reta de regressão gerada pelo algoritmo $LinearRegression$ da biblioteca scikit-learn (https://scikit-learn.org), calculada para dados sintéticos gerados aleatoriamente. Os parâmetros a_i e b são aprendidos pelo modelo e serão escolhidos aqueles cujos valores minimizem as distâncias $\sum_{i=1}^{n}(y - \hat{y})^2$, onde y é o valor real e $\hat{y}$ o valor predito, e a reta seja a mais ajustada possível aos dados.

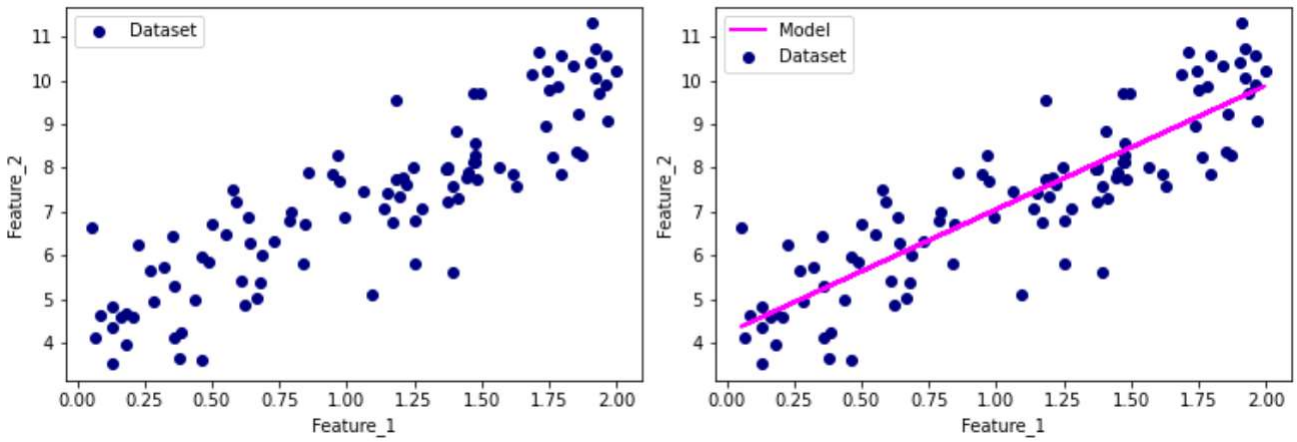

Figura 1.1: Amostras geradas randomicamente no espaço de duas variáveis (esquerda) e a reta de regressão associada (direita).

Árvores de Decisão

As árvores de decisão são modelos supervisionados bastante intuitivos e podem ser usados tanto para tarefas de classificação quanto de regressão, embora a primeira categoria seja a mais comum. Nesse modelo, cada nó representa uma tomada de decisão do tipo *if* / *else* que exige um retorno do tipo *True* / *False* [9]. Quanto mais profunda a árvore, isto é, quanto mais ramificações, maior é a complexidade do algoritmo. No exemplo da Figura 1.2, a árvore é construída para decidir qual é o animal (Falcão, Pinguim, Golfinho ou Urso) baseado em suas características. A primeira pergunta é "Possui penas?". Se há um retorno *False*, então há duas possibilidades: Golfinho ou Urso. A decisão é levada, então, ao segundo nó: "Possui nadadeiras?". Se a resposta for *False* novamente, então o animal em questão pertence à classe "Urso". Ou seja, a cada condição verificada, a pergunta segue em fluxo para um nó mais profundo da árvore, até não haver mais nós, que é o estado de "folhas puras": quando o resultado pertence a uma classe única. Qualquer novo dado sujeito à decisão da árvore será categorizado dentro de uma das quatro espécies usadas para treinamento, pois o algoritmo não é capaz de criar novas categorias.

Esse modelo de classificador é bastante sensível aos dados de treinamento, podendo um erro se propagar rapidamente pelos galhos. Para contornar esse problema, muitas vezes são usadas várias árvores em conjunto, as chamadas *random forests*, de forma que cada árvore é responsável por apenas uma parte dos dados de treino. Embora cada uma delas tenha tendência a ficar "enviesada" pelos dados, o resultado final é uma média sobre todos os resultados, e que compensa o erro, aumentando sua capacidade de generalização, o que torna as *random forests* poderosas técnicas de classificação.

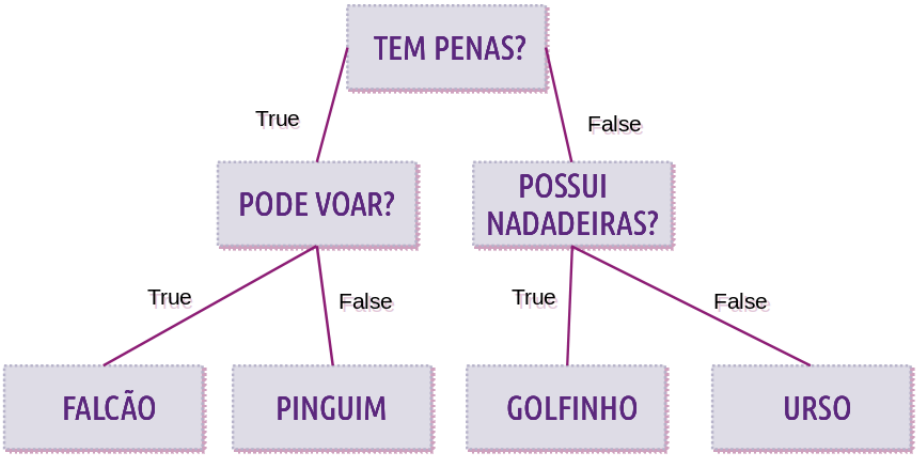

Figura 1.2: Árvore de decisão para distinguir animais a partir de suas características. [9]

Máquinas de Vetores de Suporte

Máquinas de Vetores de Suporte (do inglês *Support Vector Machines* - SVM's) são modelos para classificação e regressão, e também podem ser utilizadas para detecção de *outliers*. Esses modelos, quando usados para classificação, têm por objetivo encontrar o hiperplano que melhor segrega duas classes. A Figura 1.3 mostra um recorte do conjunto de dados *Iris Flower*, que possui 150 amostras de três espécies da flor íris (*Virginica*, *Versicolor* e *Setosa*), cujas variáveis são "Largura da pétala (cm)" e "Comprimento da pétala (cm)". Na figura, apenas as classes *Versicolor* e *Setosa* aparecem. À esquerda, a reta rosa e a vermelha separam os grupos tal que todos os dados de uma classe estão de uma lado apenas. Há infinitos planos que podem separá-las, mas o ideal é encontrar aquele que dá a maior margem entre as classes, como mostrado à direita na figura. Quanto maior a margem entre os objetos dos extremos de cada categoria, maior capacidade de generalização para novas amostras terá o classificador.

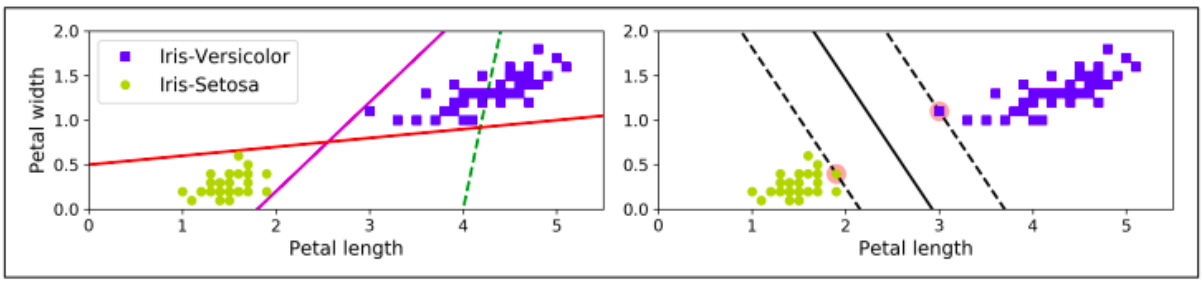

Figura 1.3: Diferentes retas separam duas classes distintas. O hiperplano ótimo é aquele que maximiza a margem entre as duas classes. [10]

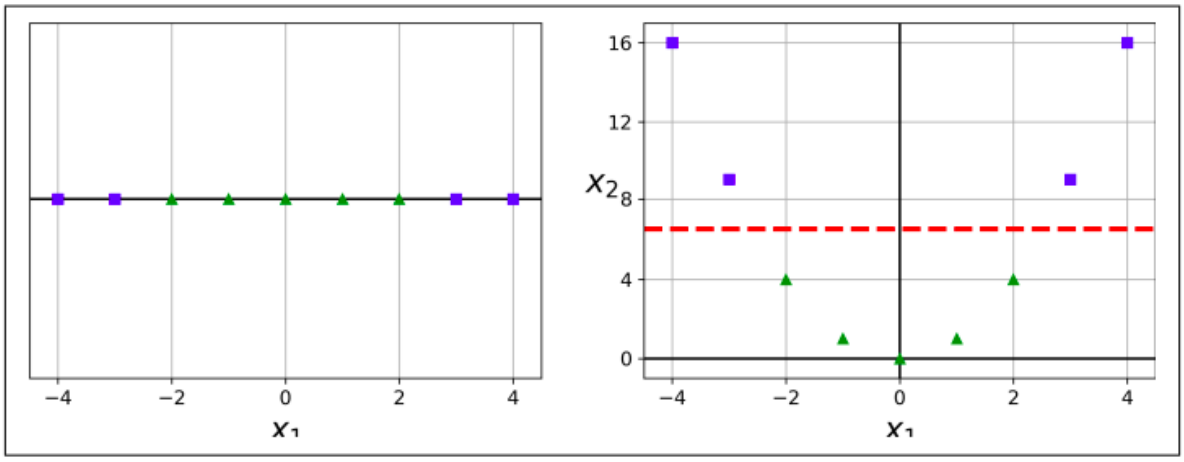

Figura 1.4: Adição de variáveis para tornar o conjunto de dados linearmente separável. [10]

Em problemas reais, nem sempre as classes são linearmente separáveis. Dessa forma, é necessário utilizar técnicas de mapeamento nos dados para uma dimensão superior, chamadas de *kernel tricks*, onde seja possível encontrar o hiperplano que melhor recorta o espaço. A Figura 1.4 mostra (à direita) a transformação $x_2 = (x_1)^2$ referente aos dados unidimensionais à esquerda. Quando em uma dimensão, não é possível encontrar uma única reta que separe as duas classes de pontos, mas a adiação de uma nova variável torna isso tangível. As SVM's são sensíveis à escala dos dados e à presença de *outliers*, e, muitas vezes, quando há sobreposição entre as classes, é preciso utilizar uma "margem suave", que permite que alguns pontos sejam erroneamente classificados.

Redes Neurais Artificiais

As Redes Neurais Artificiais (RNA's) são os mais conhecidos e um dos mais importantes modelos em *machine learning*. Inspiradas nas conexões entre os neurônios biológicos, uma RNA possui como unidade fundamental o $perceptron$, cuja estrutura básica é análoga à que compõe a arquitetura que permite que neurônios reais transmitam informações entre si. A Figura 1.5 mostra essa estrutura na sua forma mais básica.

Os elementos x_i são as entradas e os ω_i são chamados *pesos sinápticos*. Cada entrada é multiplicada por um peso tal que

$$\Sigma = \omega_1 x_1 + \omega_2 x_2 + \cdots + \omega_n x_n \tag{1.3}$$

ou seja, corresponde à soma ponderada sobre todas as entradas. A adição de um viés ($bias$) b_k resulta na função v_k, que pode, ainda, ser passada como parâmetro a uma função de ativação φ, isto é,

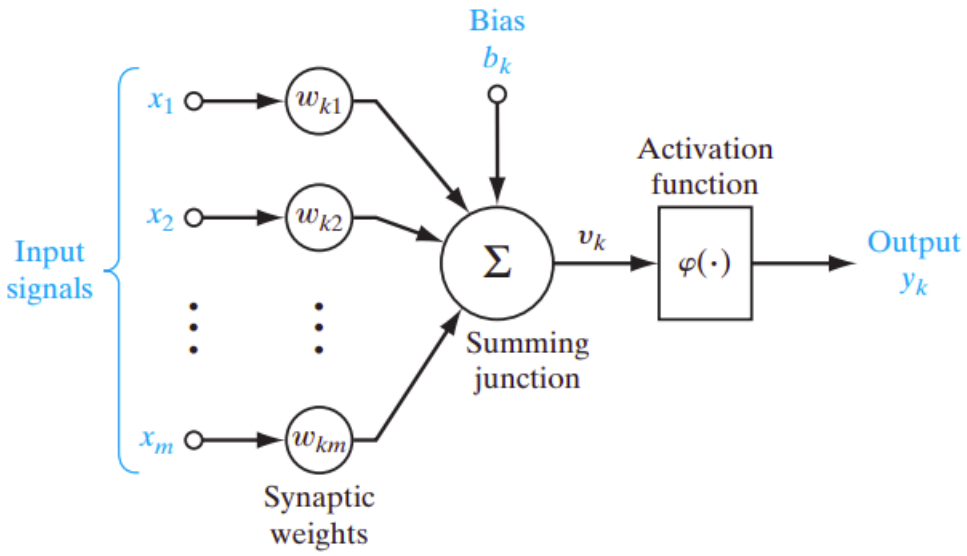

Figura 1.5: Modelo esquemático de um neurônio artificial. [11]

$$\hat{y} = \varphi(v_k) = \varphi(\Sigma + b_k). \tag{1.4}$$

As funções de ativação são usadas para transformar a entrada linear em não-linear, aumentando a complexidade do modelo. Quando a saída final alcança um determinado limite estipulado, o neurônio é ativado e a informação passa adiante. O aprendizado de uma RNA depende de múltiplas iterações; ao final de cada passagem dos dados pela rede, é calculada uma função erro, que mede o quão distante está o valor predito do valor real esperado, permitindo que os pesos ω_i sejam ajustados, para que, na próxima iteração, o erro seja menor. Tipicamente, uma rede neural pode ser constituída por várias camadas de neurônios (Figura 1.6), e quanto maior o número de neurônios e de camadas, maior complexidade em análise o modelo é capaz de alcançar.

As RNA's desmembraram-se em dezenas de outras arquiteturas mais complexas [12], como as Redes Neurais Convolucionais, que são mais comumente usadas para Visão Computacional: análise de imagens, detecção e classificação de objetos, reconhecimento facial, entre outros aspectos; e as Redes Neurais Recorrentes para Processamento de Linguagem Natural, como tradutores e assistentes virtuais e reconhecimento de fala. Hoje, algumas das atividades mais comuns do dia a dia (usar o *Google Translator*; acessar o próprio telefone apenas ficando diante da câmera; ou mesmo, à distância, agendar um alarme "para daqui a 10 minutos" com apenas alguns comandos de voz) não poderiam ser imaginadas se não fosse a existência de redes neurais artificiais, em seus inúmeros formatos e em constantes atualizações.

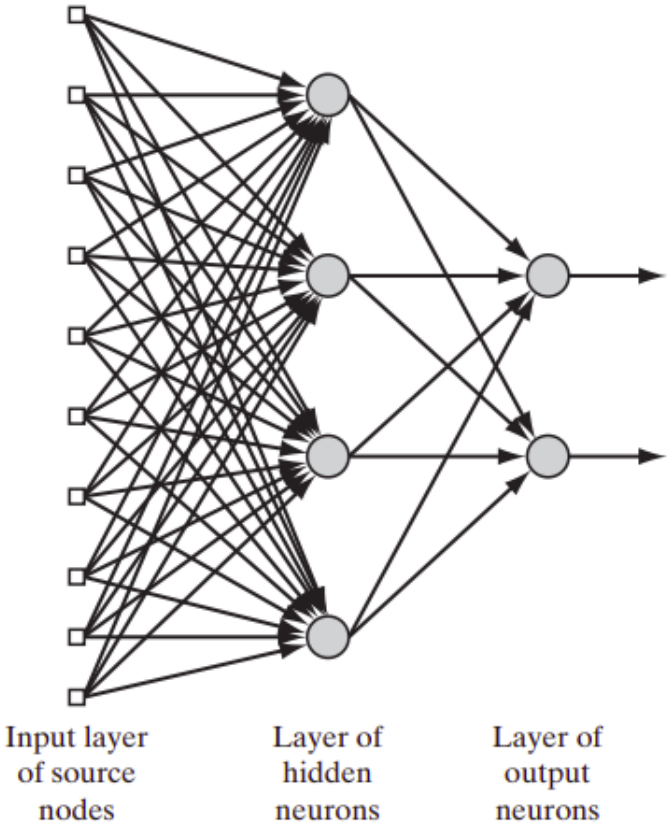

Figura 1.6: Arquitetura de uma rede neural completamente conectada do tipo *feedfoward* com 10 entradas, uma camada oculta com 4 neurônios e uma camada de saída com 2 neurônios. [11]

1.1.2 Modelos Não Supervisionados

Nas técnicas não supervisionadas, os modelos não operam baseados em experiência, isto é, não há um *label* com o qual um dado desconhecido possa ser comparado. Esses métodos se dividem entre os de clusterização e os de redução de dimensionalidade. Neste trabalho, dois dos algoritmos mais populares de ambas as categorias foram usados: o método de agrupamento *K-Means*, e a Análise de Componentes Principais. Discutiremos esses dois algoritmos com maiores detalhes no Capítulo 3. Entre os de clusterização, destacam-se ainda os Agrupamentos Hierárquicos (Figura 1.7), que podem ser representados na forma de dendrogramas (Figura 1.8), e os Modelos de Misturas Gaussianas, que promovem agrupamentos do tipo $soft$, onde cada dado em uma distribuição está associado a um grupo com certa probabilidade, podendo, assim, pertencer a um ou mais grupos.

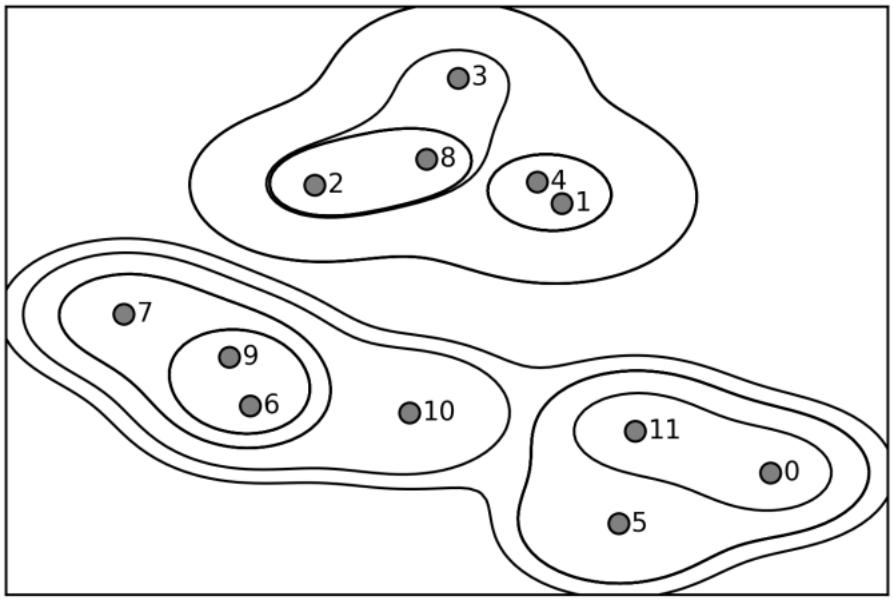

Figura 1.7: Diversos níveis de agrupamento representados pelas linhas contínuas em uma distribuição de dados sintéticos. [9]

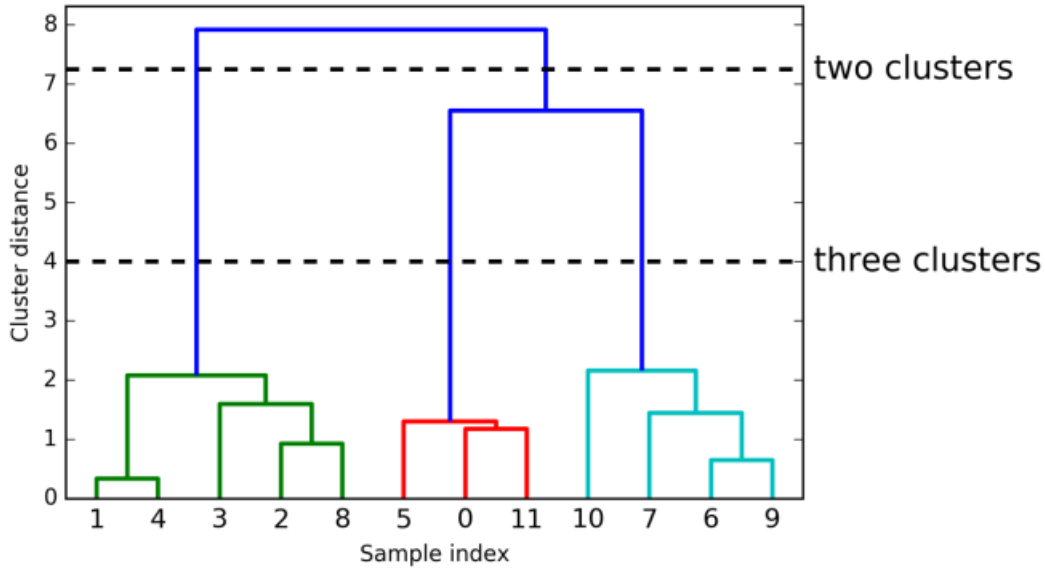

Figura 1.8: Representação gráfica da hierarquia de grupos referente à Figura 1.7 usando um dendrograma. As linhas tracejadas indicam a distância máxima necessária entre os pontos para a formação de dois e três grupos. [9]

Haja vista o enorme potencial de aplicações, essas técnicas têm sido naturalmente incorporadas no âmbito das ciências físicas como auxiliares na identificação de padrões, como, por exemplo, no estudo de transições de fase. Tais técnicas mostram-se como uma alternativa na compreensão de problemas complexos devido ao custo computacional relativamente baixo de execução, comparado aos métodos tradicionais, e pela possibilidade de estender essas análises a sistemas físicos pouco explorados. Assim, há uma grande expectativa de que essas técnicas possam lançar luz sobre fenômenos ainda desconhecidos. Apesar de muitos resultados animadores, ainda existem muitas lacunas sobre o grau de efetividade de tais metodologias e em quais circunstâncias devem ser utilizadas. Em vista disso, a análise de *toy models* continua sendo uma boa alternativa para comparação direta entre as aplicações de diferentes metodologias.

Neste trabalho, temos por objetivo analisar a aplicação de técnicas não supervisionadas para a análise de padrões em um sistema físico que apresenta transições de fase de primeira e segunda ordem: o modelo de Potts de $q-$estados, um *toy model* de particular interesse em Física da Matéria Condensada e Mecânica Estatística. Este modelo vem sendo estudado majoritariamente através de cálculos númericos e de expansão em séries. De um ponto de vista de *machine learning*, foi analisado via redes neurais [13] para $q = 3$ em duas dimensões na versão antiferromagnética, e $q = 5$ em três dimensões. Os resultados encontrados são concordantes para as temperaturas críticas apresentadas na literatura. No decorrer deste trabalho, entretanto, investigaremos o modelo de Potts a partir de técnicas de *machine learning* não supervisionadas, em redes bidimensionais e com interação ferromagnética. As técnicas que foram utilizadas envolvem redução de dimensionalidade, agrupamento por semelhança e análise topológica.

No capítulo seguinte, veremos algumas contribuições relevantes para o estudo destes fenômenos, nas quais técnicas de *machine learning* foram usadas para o estudo de sistemas clássicos, a fim de estudar a criticalidade em transições de fase de primeira e segunda ordem.

2 *Machine Learning* no contexto das transições de fase

O estudo de transições de fase e fenômenos críticos é um tema central em Física, e é objeto de extensas investigações há várias décadas. Quando, num dado sistema, a variação de algum parâmetro nas proximidades de um determinado ponto leva a mudanças "estruturais" e divergências em grandezas termodinâmicas (calor específico, susceptibilidade, etc.), diz-se que naquele ponto há a ocorrência de uma transição de fase. Os sistemas mais usais para exemplificar tal fenômeno são os fluidos sob variação de temperatura (T) e pressão (P), e sistemas magnéticos sob variação de temperatura ou campo magnético externo (H). Para quantificar uma dada transição de fase, é preciso definir um *parâmetro de ordem*. Por exemplo, para o caso de sistemas fluidos, tal parâmetro é dado pela diferença entre as densidades das fases: $\rho_L - \rho_g$ (para o caso Líquido-Gás). A Figura 2.1 mostra o diagrama de fases para um fluido no plano $P-T$, onde as regiões marcadas como Sólido, Líquido e Gás representam fases em equilíbrio, e as linhas sólidas indicam regiões de coexistência (Sólido-Gás (SG), Sólido-Líquido (SL) e Líquido-Gás (LG)).

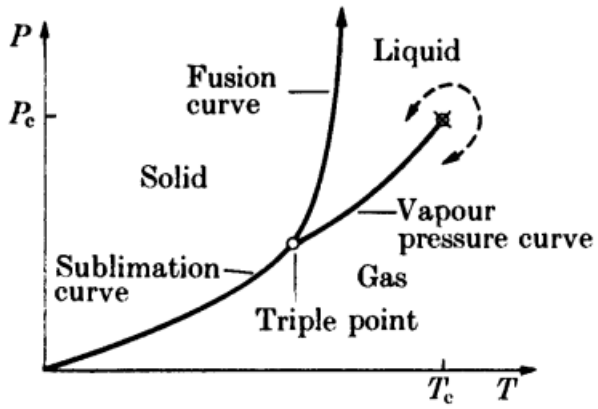

Figura 2.1: (a) Diagrama de fases para um fluido. P_c e T_c representam pressão crítica e temperatura crítica, respectivamente. As linhas tracejadas indicam que, naquele ponto, o sistema muda de fase de forma contínua. [14]

Nas proximidades das regiões de coexistência, as derivadas da energia livre apresentarão descontinuidades ou divergências que caracterizarão o tipo de transição de fase. Se, no ponto crítico, a entropia (S) e o volume (V) forem descontínuos, será chamada *transição de primeira ordem*, e haverá a presença de calor latente. Por outro lado, se essas primeiras derivadas da energia livre são funções contínuas, mas o calor específico C_v e a compressibilidade divergem, será dita *transição de segunda ordem*. No diagrama $P-T$ mostrado, as transições GS, GL e LS, indicadas pelas linhas sólidas, são todas de primeira ordem. Por outro lado, acima da curva de coexistência GL, no ponto (T_c, P_c), líquido e gás são indistinguíveis. Assim, à medida que a temperatura diminui, a diferença $\rho_L - \rho_g$ cresce continuamente de zero, e a transição nesse ponto é de segunda ordem. Nesse caso, o parâmetro de ordem deve variar como função da temperatura reduzida $\epsilon \equiv T - T_c/T_c$ elevada a um *expoente crítico*: $\rho_L - \rho_g \sim (-\epsilon)^{\beta}$. Outras grandezas, como o calor específico a volume constante, a compressibilidade e o comprimento de correlação também obedecem a essa lei, com os seus respectivos expoentes críticos α, γ e ν. As *classes de universalidade* são definidas pelas transições de fase que compartilham os mesmos expoentes críticos. Essas relações também são válidas para o caso magnético, com a devida substituição pelas grandezas análogas.

Técnicas de simulação computacional, como o método de Monte Carlo, têm um importante papel no estudo de fenômenos críticos [15], dada a possibilidade de se analisar, de forma não enviesada, sistemas interagentes, cuja solução analítica seria muito complexa. No entanto, mesmo com os avanços das últimas décadas em termos de processamento computacional e armazenamento, quando a quantidade de partículas do sistema é grande ($N \gg 1$), tais análises tornam-se tarefas custosas e limitadas. Nesse contexto, com o avanço da Inteligência Artificial, é possível explorar tais sistemas físicos a partir dessas técnicas modernas, especialmente as técnicas não supervisionadas, que não necessitam de conhecimento prévio sobre qualquer informação qualitativa sobre um dado sistema, mas são direcionadas apenas à identificação de padrões.

No contexto de Física Estatística e Matéria Condensada, em 2016, Wang [16] introduziu o uso de *machine learning* não supervisionado usando *Principal Component Analysis* (PCA) para o estudo do modelo de Ising. Essa técnica tem como ideia principal a redução da dimensionalidade de um sistema, de tal forma que seja possível extrair informações relevantes a partir de um subespaço com um número muito reduzido de parâmetros (mais detalhes, vide Cap. 3). Para estudar o modelo de Ising, foram usadas amostras de Monte Carlo descorrelacionadas, calculadas em uma rede quadrada, formando uma matriz X que continha configurações de estado do sistema para diferentes temperaturas, com T variando no domínio $T/J = (1,6,\ldots,2,9)$. A matriz X está

$$X = \begin{pmatrix} \uparrow & \downarrow & \uparrow & \cdots & \uparrow & \uparrow & \uparrow \\ & & & \vdots & & & \\ \downarrow & \uparrow & \downarrow & \cdots & \uparrow & \downarrow & \uparrow \end{pmatrix}_{M \times N}$$

Figura 2.2: Matriz formada por amostras de Monte Carlo calculadas para o modelo de Ising numa rede quadrada. [16]

graficamente ilustrada na Figura 2.2. Aqui, M é a quantidade de amostras (estados), $N = L^2$ (onde L é o tamanho linear da rede), e os vetores $\uparrow$ e $\downarrow$ representam as variáveis de *spin*. Após a aplicação de PCA, o espaço de dados, de dimensão L^2, foi reduzidao para uma dimensão inferior (nesse caso, duas dimensões/variáveis), em que as novas variáveis (y_1, y_2) são chamadas de *componentes principais*. A Figura 2.3 mostra a dispersão dos dados nesse novo espaço para três tamanhos de rede ($N = 20^2, 40^2$ e 80^2). Cada cor indica a temperatura em que aqueles dados foram gerados, conforme indicado pela barra lateral. No espaço das componentes principais, pontos próximos são semelhantes, enquanto pontos distantes possuem dissimilaridade. Isso significa que, de alguma forma, aqueles dados têm naturezas diferentes. Pode-se interpretar que as componentes principais mostram uma "fotografia" de onde ocorre a maior variância nos dados. A partir da figura, é possível notar que, para grupos de cores distintas, há um espalhamento diferente no espaço das componentes principais. Enquanto os dados em vermelho se agloremam no centro (temperatura elevada), os pontos em azul (baixa temperatura) se situam nos extremos direito e esquerdo. O aglomerado de pontos no centro descreve a alta similaridade dos dados em alta temperatura, que, de certo modo, reflete a natureza dos *spins* desordenados. Por outro lado, para temperaturas abaixo de T_c, há dois grupos distintos: como o modelo de Ising tem sua simetria Z_2, então é natural imaginar que esses dois grupos em baixa temperatura refletem a natureza da simetria do estado fundamental do modelo de Ising.

A diferença entre o sistema a baixas temperaturas e a altas temperaturas é evidente. Porém, ainda resta encontrar o ponto crítico em que isso ocorre. Assim, na Ref. [17] os autores analisaram o comportamento das componentes principais individualmente, como mostrado nas Figuras 2.4 (c) e (d) (em que p_1 e p_2 são as componentes principais e L é o tamanho linear da rede e cujos valores estão indicados na própria figura.). Dessa forma, é possível notar que a primeira componente principal tem a forma da magnetização, enquanto a segunda componente se assemelha à susceptibilidade magnética. Assim, por analogia com sistemas magnéticos, obtém-se que a região crítica está nas proximidades de $T = 2,3$ (como esperado para o modelo de Ising).

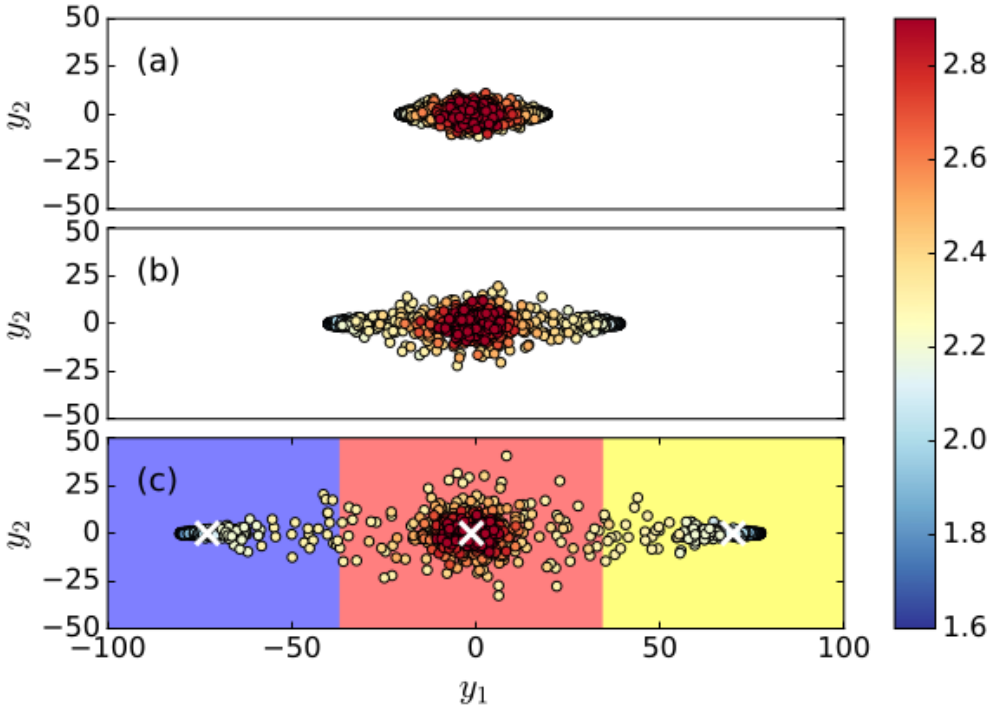

Figura 2.3: Projeção dos dados no plano das duas componentes principais para (a) $N = 20^2$, (b) $N = 40^2$ e (c) $N = 80^2$. A barra lateral à direita indica a temperatura em que os dados foram calculados. [16]

Em resumo, as componentes principais carregam informações que relacionam-se intimamente a essas grandezas físicas de interesse, podendo indicar a região crítica sem nenhuma influência externa na análise, mas apenas pelo reconhecimento de padrões nos dados.

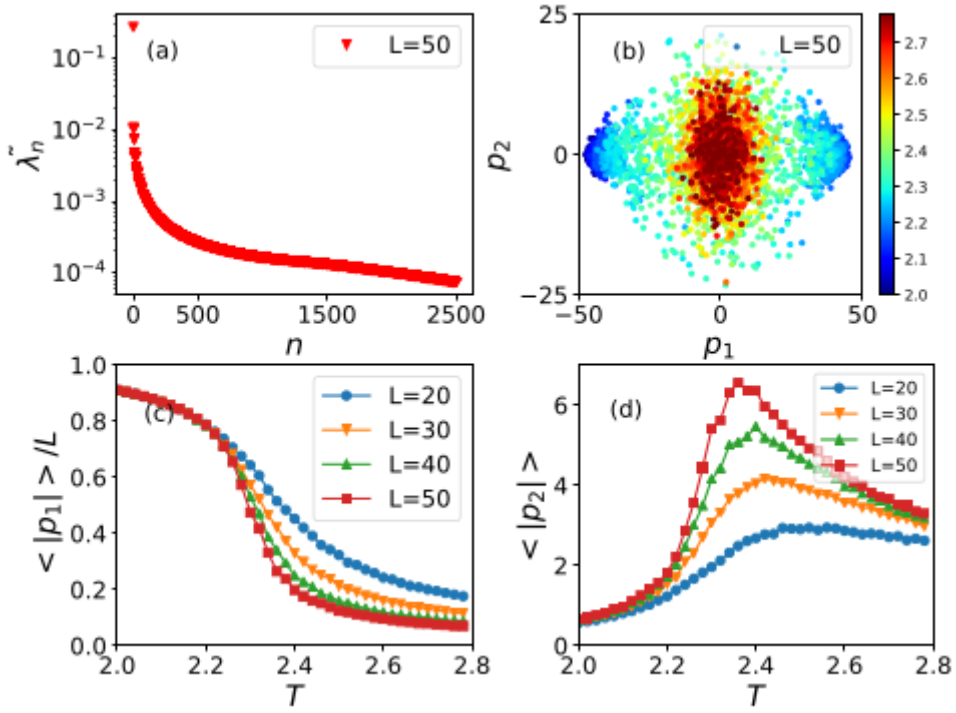

Figura 2.4: Resultados de PCA para o modelo de Ising numa rede quadrada. (a) Variâncias relativas obtidas através da normalização dos autovalores. (b) Projeção dos dados no plano das duas primeiras componentes principais. A barra de cores indica a variação de temperatura em unidades de J (c) Primeira componente principal normalizada em função da temperatura. (d) Segunda componente principal normalizada em função da temperatura. [17]

Ainda no trabalho de Hu *et. al* [17], outros modelos clássicos foram analisados, como os modelos XY e Blume-Capel (BCM), em redes quadradas, através de PCA e de *autoenconders*, um tipo particular de redes neurais. O modelo XY também foi estudado por Wang e Zhai (2017) [18] (2018) [19], e por Wetzel (2017) em três dimensões [20] utilizando PCA e uma variação de *autoencoders*. No entanto, as técnicas não supervisionadas utilizadas (PCA e Kernel PCA) conduzem a algumas limitações na identificação da transição Berezinskii-Kosterlitz-Thouless (BKT) presente neste modelo, devido ao caráter não linear do parâmetro de ordem. O valor crítico para essa transição de fase foi encontrado, com boa aproximação, através da análise da *dimensão intrínseca* de um espaço de dados (Mendes, *et. al* [21]). Os autores demonstraram que essa grandeza se comporta como um parâmetro de ordem nas proximidades de um ponto crítico, permitindo estimar temperaturas de transição também para transições de fase de primeira e segunda ordem.

Entre os modelos clássicos, é importante mencionar as soluções obtidas por Carrasquilla e Melko [22] para o modelo de Ising, utilizando redes neurais artificiais, que também marca o início dessa série de experimentações. Nesse trabalho, foram utilizadas amostras de configurações de Monte Carlo, calculadas em redes quadradas, como dados de treinamento para uma rede neural

com uma camada oculta. A Figura 2.5 mostra o resultado da camada de saída para redes triangulares. O valor exato onde o sistema passa por uma transição de fase é $T_c/J = 4/ln(3)$, indicado pela linha laranja vertical. É possível notar que os valores de saída da rede neural também convergem para esse valor crítico. O poder de generalização de uma ANN permite a estimação da forma do parâmetro de ordem de um sistema ainda que tenha sido treinada em um diferente formato de rede. Os autores apontam a importância desse resultado como a possibilidade de estudar sistemas desconhecidos a partir dos já bem estabelecidos na literatura.

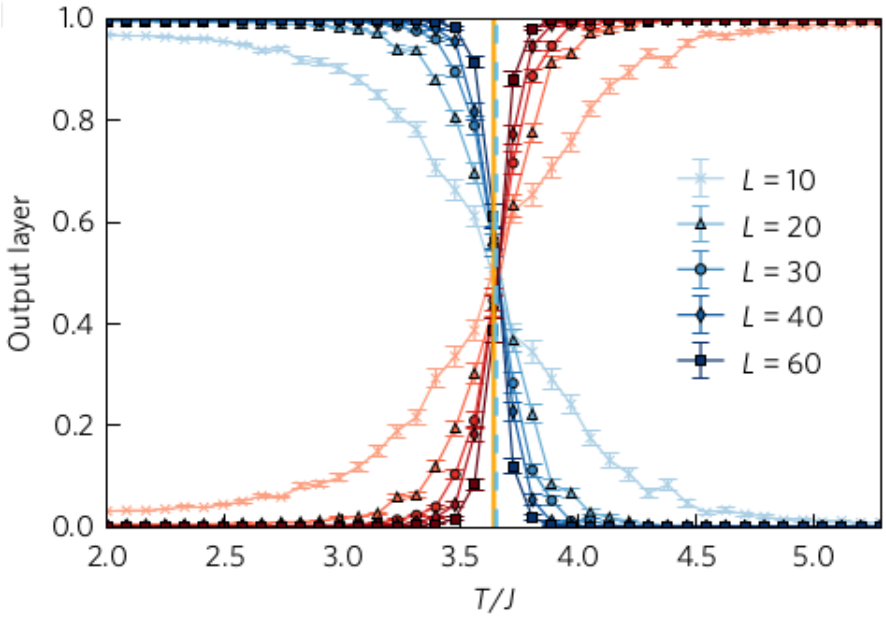

Figura 2.5: Camada de saída de uma rede neural em função de T/J treinada com amostras de Monte Carlo calculadas para o modelo de Ising em uma rede quadrada e testada em uma rede triangular, com interação ferromagnética. A análise foi feita para os tamanhos de rede (L) indicados na figura. A linha laranja vertical indica a temperatura crítica exata prevista para a rede triangular. [22]

Para trazer uma perspectiva a respeito de modelos quânticos, vale citar que o modelo de Hubbard já foi analisado sob a ótica de técnicas como PCA e redes neurais, em diferentes formas de rede [23–26], assim como o modelo de Anderson periódico [26], este através de *Topological Data Analysis (TDA)*, uma técnica que será discutida adiante e que consiste em um *mix* de modelos não supervisionados já conhecidos, como agrupamento *fuzzy* e clusterização espectral. Em todos os casos os autores identificaram pontos críticos concordantes com aqueles previstos para cada modelo.

Em resumo, as técnicas de ML têm se mostrado uma forte ferramenta auxiliar para o estudo de transições de fase quando associadas a simulações de Monte Carlo (clássico ou quântico) [27–29], para a identificação de pontos críticos. Vale mencionar que um estudo recente [30] mostrou que uma rede neural treinada com um modelo pertencente a uma determinada classe de

universalidade era capaz de reconhecer a criticalidade em outros modelos pertecentes à mesma classe, retornando com precisão razoável os expoentes críticos, mesmo com Halmitonianas e geometrias de rede não equivalentes. Com isso, pode-se ver um pouco da versatilidade desses modelos, de forma que suas aplicações podem ser concebidas para problemas diversos.

Para elevar esse estudo a modelos físicos (ou de outras áreas do conhecimento) com maior complexidade, certamente se exigirá modelos computacionais também mais complexos, mas os resultados até aqui obtidos mostram que esse caminho está sendo percorrido com êxito em sua maior parte.

3 Modelo e Metodologia

Descrevemos neste capítulo os modelos e métodos utilizados neste trabalho, porém, dando enfoque nas técnicas de *machine learning*. O leitor pode encontrar mais detalhes sobre o modelo de Potts e as técnicas de Monte Carlo nas referências [31–33].

3.1 Modelo de Potts de q-estados

O modelo de Potts (1952) é um modelo matemático que descreve a interação entre "*spins*" numa rede cristalina, cuja Hamiltoniana é

$$H = -J \sum_{ij} \delta(\sigma_i, \sigma_j), \tag{3.1}$$

onde $\delta(\sigma_i, \sigma_j)$ corresponde à função delta de Kronecker e $\sigma_i(\sigma_j)$ correspondem às variáveis de *spin*, com orientações definidas por ângulos $\theta_n = 2\pi n/q$ $(n = 0, 1, 2, \ldots, q-1)$, como mostrado na Figura 3.1. Este modelo pode ser considerado uma generalização do modelo de Ising, onde os *spins* podem orientar-se não apenas em duas, mas em q direções equidistantes. Isto é, o modelo de Potts para $q = 2$ descreve o modelo de Ising com uma normalização na escala de energia. Vale ressaltar que as diferentes orientações de *spin* não necessariamente refletem estados de *spin*, mas podem corresponder apenas a orientações de regiões mesoscópicas magnetizadas e separadas por paredes de domínio. Além de materiais magnéticos, esse modelo é capaz de simular tecidos orgânicos para o estudo de tumores e doenças [34], como também fluidos e espumas [35], e até mesmo problemas de nível social e econômico, quando envolvem interação entre as entidades [36, 37].

Esse modelo já foi extensivamente investigado em redes bidimensionais e apresenta transição de fase de primeira ordem para $q > 4$ e transição contínua para $q \leq 4$ [31]. Em particular, no caso da rede quadrada, é possível encontrar a temperatura crítica de forma exata, cujo valor é

$$\frac{k_B T_c}{J} = \frac{1}{ln(1+\sqrt{q})}, \tag{3.2}$$

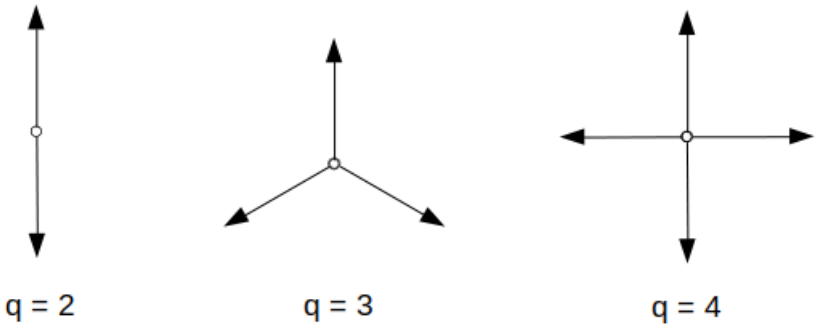

Figura 3.1: q vetores unitários que distam simetricamente representando as possíveis orientações de *spins* numa rede.

onde T_c é a temperatura do sistema e k_B é a constante de Boltzmann. Tais resultados exatos servem de base para testagem de novas técnicas, como feito neste trabalho.

3.2 Método de Monte Carlo

Os modelos estatísticos chamados de métodos de Monte Carlo são conhecidos por sua natureza probabilística e utilizam amostragens aleatórias de um sistema que evolui estocasticamente. Para isso, é necessário gerar uma série de amostras independentes que sejam representativas do sistema em questão e evoluí-las (numa cadeia de Markov) através de probabilidades que satisfaçam o balanço detalhado. Para estudar o modelo de Potts, foi utilizado o algoritmo de *heat-bath* para gerar amostras de configurações de *spins* em redes quadradas, sob variação de temperatura e com interação ferromagnética, cuja energia é dada pela equação 3.1.

Para simplificar a codificação, usaremos como exemplo o modelo de Ising (que entende-se como sendo o modelo de Potts para q =2), seguindo o algoritmo de *Metropolis*. Considere uma cadeia de *spins* $\sigma = \pm 1$, representados pelos estados de *spin* $\uparrow$ (*up*) e $\downarrow$ (*down*), numa dada configuração

$$S_1 = \uparrow\downarrow\uparrow\downarrow\downarrow\uparrow\cdots\uparrow . \tag{3.3}$$

O cálculo da energia de uma configuração é feito através da Hamiltoniana de Ising (sem o termo de campo externo), onde o somatório se dá sobre os primeiros vizinhos e $J > 0$:

$$H_S = -J\sum_{ij}\sigma_i\sigma_j . \tag{3.4}$$

Depois de inverter um *spin* (o primeiro, por exemplo) uma nova configuração S_t é obtida:

$$S_t = \downarrow\downarrow\uparrow\downarrow\downarrow\uparrow\cdots\uparrow . \tag{3.5}$$

A probabilidade de ocorrência da configuração S_t, a partir de S_1, é dada pelo fator de Boltzmann:

$$P_{flip} = \frac{w(S_t)}{w(S_1)} = e^{-\beta[H(S_t)-H(S_1)]}, \tag{3.6}$$

onde $\beta = 1/k_BT$ e k_B é a constante de Boltzmann. Se $P_{flip} > 1$, a nova configuração possui energia menor e é mais provável que a anterior, portanto, S_t é aceita ($S_t \rightarrow S_2$). Assim, deve-se prosseguir para um novo *flip*

$$S_2 = \downarrow\downarrow\uparrow\downarrow\downarrow\uparrow\cdots\uparrow \tag{3.7}$$

$$S_t = \downarrow\uparrow\uparrow\downarrow\downarrow\uparrow\cdots\uparrow \tag{3.8}$$

Dessa forma,

- se $P_{flip} > 1$, a inversão é aceita;
- se $P_{flip} < 1$:
  - calcula-se um número aleatório $\mathbf{r} \in [0,1]$.
  - se $\mathbf{r} < P_{flip}$, a inversão é aceita;
  - caso contrário, a inversão é descartada.

A aceitação de *flips* por comparação com um número aleatório, mesmo quando o novo estado possui energia maior que o anterior, simula os efeitos das flutuações. Aceitar escolhas "erradas" permite que o sistema seja ergódico. Depois de percorrida toda a rede, obtém-se um passo de Monte Carlo (MCS). Esse procedimento é feito milhares de vezes para cada valor de temperatura fixado, dentro de um intervalo $T = \{T_i, ..., T_n\}$. No entanto, apenas algumas centenas dessas amostras de MCS são coletadas, separadas por um "tempo" de iteração suficientemente longo, de forma a serem independentes umas das outras. Os dados são organizados em blocos ordenados pelas temperaturas:

$$X = \begin{bmatrix} T_i \begin{pmatrix} \uparrow & \downarrow & \uparrow \cdots \uparrow & \uparrow & \uparrow \\ & & \vdots & & \\ \uparrow & \downarrow & \uparrow \cdots \downarrow & \uparrow & \uparrow \end{pmatrix} \\ \vdots \\ T_n \begin{pmatrix} \uparrow & \downarrow & \downarrow \cdots \downarrow & \uparrow & \uparrow \\ & & \vdots & & \\ \downarrow & \uparrow & \uparrow \cdots \downarrow & \uparrow & \downarrow \end{pmatrix} \end{bmatrix}, \tag{3.9}$$

onde T_i e T_n denotam as temperaturas inicial e final, respectivamente. Dessa forma, a matriz X contém toda a evolução do sistema em análise dentro desse intervalo.

Nesse trabalho, para cada temperatura, numa faixa de 21 valores, incluindo a temperatura crítica (veja a Tab.3.1), foram feitas 478 medidas de Monte Carlo. A matriz X tem, portanto, dimensão de 478×21 linhas e $N \times N \times 2$ colunas, onde N é o tamanho linear da rede e cada *spin* é representado por um par $(cos(\theta_n), sen(\theta_n))$. Esse cálculo foi feito para redes de tamanho $N = 20, 30, 40, 50, 60$ e 80 para $q = 3$, 4 e 5. Essas matrizes compõem a base de dados que foi analisada através das técnicas que serão descritas nas seguintes subseções.

Tabela 3.1: Temperaturas utilizadas para o cálculo de Monte Carlo para cada modelo. As temperaturas críticas aproximadas são mostradas na tabela.

Modelo	Faixa de temperaturas	Temp. crítica (≈)
$q = 3$	0.90 - 1.10	0.99
$q = 4$	0.81 - 1.01	0.91
$q = 5$	0.75 - 0.95	0.85

3.3 Análise de Componentes Principais

Introduzido por Karl Pearson (1901) e desenvolvido por Harold Hotelling (1933), independentemente, a Análise de Componentes Principais [38], do inglês *Principal Component Analysis* (PCA), é uma técnica de análise multivariada das mais utilizadas. Caracteriza-se como uma técnica de *machine learning* não supervisionado e seu principal objetivo é reduzir a dimensionalidade de um conjunto de dados de n-variáveis que possua alto valor de variáveis correlacionadas, de tal forma que se mantenham conservadas as maiores variâncias deste conjunto, transpondo-o do espaço das variáveis originais para um novo espaço de menor dimensão linear, formado por eixos descorrelacionados. A estes novos eixos dá-se o nome de componentes principais (PC's). Estas possuem duas propriedades intrínsecas: a primeira é que são ortogonais entre si, descorrelacionadas; a segunda é que são escolhidas de forma que a primeira PC exiba a máxima variância dos dados, a segunda exiba a segunda maior variância e assim em diante. Ou seja, são ordenadas em termos da variabilidade "explicada" por cada PC, da maior para a menor.

A Figura 3.2 mostra, à esquerda, os eixos de máxima variância numa nuvem de pontos no espaço de duas variáveis e, à direita, a mesma nuvem de pontos no espaço gerado por estas duas componentes principais. Neste exemplo, a técnica PCA promove apenas uma rotação nos eixos de coordenadas. No entanto, é natural pensar que essa técnica pode ser compreendida para espaços n-dimensionais onde, de fato, possa ser feita uma redução de dimensionalidade.

Pode-se pensar em um conjunto de dados que exista inicialmente em três dimensões, geradas por váriaveis x_1, x_2 e x_3, e de tal forma que a PCA possa ser aplicada. As componentes principais são construídas a partir de combinações lineares destas variáveis originais. Então, se é feita a escolha de reduzir de três para duas dimensões, por exemplo, as PC's (que agora serão chamadas de $\vec{y}$ por simplificação) terão a seguinte forma:

$$\vec{y}_1 = \alpha_{11}x_1 + \alpha_{21}x_2 + \alpha_{31}x_3 \tag{3.10}$$

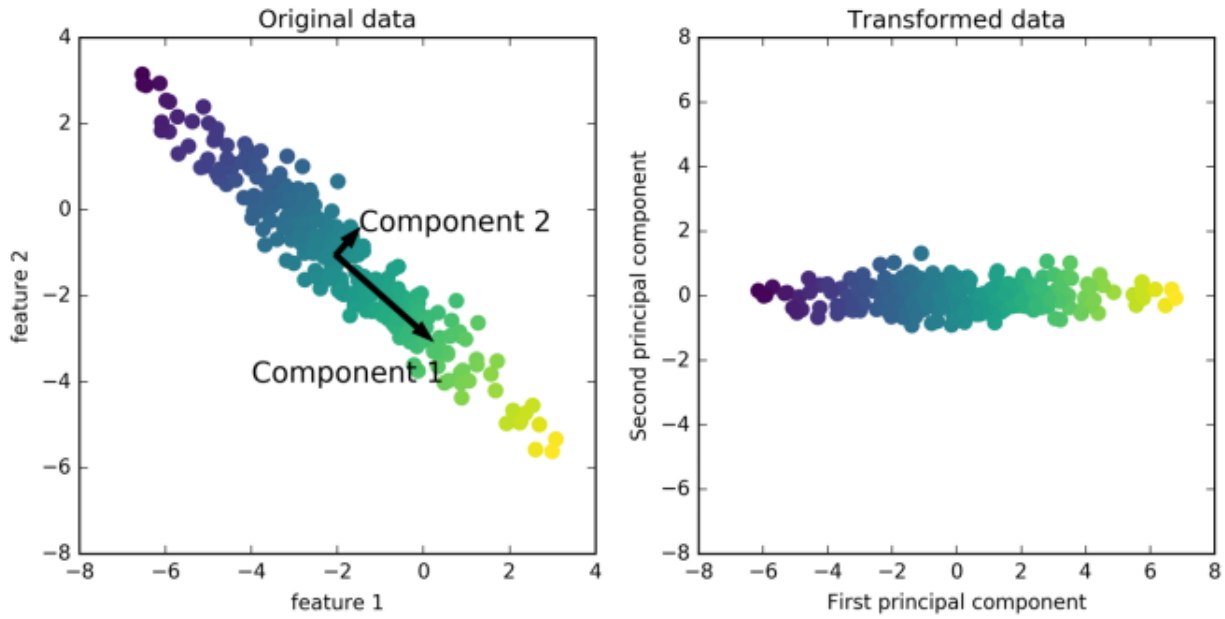

Figura 3.2: Conjunto de amostras no plano de duas variáveis. À esquerda, os eixos sobre a nuvem de pontos mostram as direções de maior variação nos dados. À direita, uma nova representação dos dados no espaço destas componentes principais. [9]

$$\vec{y}_2 = \alpha_{12}x_1 + \alpha_{22}x_2 + \alpha_{32}x_3. \tag{3.11}$$

Para o caso de n variáveis, pode-se escrever

$$\begin{pmatrix} \vec{y}_1 \\ \vec{y}_2 \\ \vdots \\ \vec{y}_n \end{pmatrix} = \begin{pmatrix} \alpha_{11} & \cdots & \alpha_{n1} \\ \alpha_{12} & \cdots & \alpha_{n2} \\ \vdots & \ddots & \vdots \\ \alpha_{1n} & \cdots & \alpha_{nn} \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \tag{3.12}$$

onde a matriz de coeficientes α_{ij} rotaciona os eixos coordenados em n dimensões para que os vetores $\vec{y}_j$ sejam as direções das componentes principais. Estes coeficientes estão sujeitos aos vínculos

$$\alpha_{1n}^2 + \alpha_{2n}^2 + \cdots + \alpha_{nn}^2 = 1, \tag{3.13}$$

ou, de forma mais compacta,

$$\vec{\alpha}_j^T \vec{\alpha}_j = 1, \tag{3.14}$$

e, para $i \neq j$,

$$\vec{\alpha}_i^T \vec{\alpha}_j = 0. \tag{3.15}$$

Matriz de covariância

Para entender como as variáveis de um conjunto de dados estão correlacionadas, pode-se definir uma grandeza chamada de *covariância*, que mede o grau de similaridade entre duas variáveis. Para o caso de um problema com m medidas de n variáveis $(\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n)$, a covariância entre duas destas variáveis (por exemplo, $\vec{x}_1$ e $\vec{x}_2$) pode ser escrita como

$$cov(\vec{x}_1, \vec{x}_2) = \frac{1}{m-1} \sum_{i=1}^{m} (x_{1i} - \bar{x}_1)(x_{2i} - \bar{x}_2), \tag{3.16}$$

onde a barra indica o valor médio daquela variável. Ou ainda

$$cov(\vec{x}_1, \vec{x}_2) = \frac{1}{m-1} \vec{x}_1^T \vec{x}_2, \tag{3.17}$$

onde os vetores $\vec{x}_1$ e $\vec{x}_2$ também estão centralizados e T indica o vetor transposto.

A covariância entre uma variável e ela própria é chamada de *variância* e é uma medida da dispersão de valores no eixo daquela variável. Altas variâncias são relevantes e dão mais peso ao cálculo das componentes principais. Por outro lado, uma alta covariância indica que duas variáveis crescem ou decrescem de forma parecida, o que gera redundância no conjunto de dados. Já uma baixa covariância indica que duas variáveis se comportam independentemente, e este é um comportamento que deve ser conservado. No espaço das componentes principais, os novos eixos expressam variâncias máximas e possuem covariâncias nulas.

Define-se *matriz de covariância* [39] a matriz C_x que contém todas as correlações entre as variáveis de um problema. Então, dada uma matriz de dados $\mathbf{X}$, com dimensão $m \times n$ (m medidas e n variáveis),

$$C_x = \frac{1}{m-1} \mathbf{X^T X}. \tag{3.18}$$

O cálculo das componentes principais

Uma componente principal cuja importância é de ordem j, pode ser definida como

$$\vec{y}_j = \mathbf{X} \vec{\alpha}_j, \tag{3.19}$$

onde $\vec{\alpha}_j$ é um vetor da matriz de rotação da equação 3.12. O objetivo é encontrar quais valores para $\vec{\alpha}_j$ maximizam um determinado $\vec{y}_j$ para que ele seja a primeira componente principal. Depois, qual outro $\vec{\alpha}_j$ maximiza outro $\vec{y}_j$ para que ele seja a segunda componente principal, e assim em diante. A rotação em si não minimiza a quantidade de novos eixos, mas a minimização se dá quando algumas componentes carregam suficientemente das informações tal que as demais

possam ser descartadas. Desta forma, o objetivo é encontrar o argumento $\vec{\alpha}_1$ que maximiza a variância de $\vec{y}_1$, ou seja,

$$arg_{\vec{\alpha}_1}\left[max\left[var(\vec{y}_1)\right]\right], \tag{3.20}$$

para que $\vec{y}_1$ seja a primeira componente principal. Fazendo uso da eq. 3.17, a variância de $\vec{y}_1$ pode ser escrita como

$$cov(\vec{y}_1,\vec{y}_1) = var(\vec{y}_1) = \frac{1}{m-1}\vec{y}_1^T\vec{y}_1, \tag{3.21}$$

e, a partir da eq. 3.19,

$$var(\vec{y}_1) = \frac{1}{m-1}(\mathbf{X}\vec{\alpha}_1)^T(\mathbf{X}\vec{\alpha}_1) \tag{3.22}$$

$$var(\vec{y}_1) = \frac{1}{m-1}(\vec{\alpha}_1^T\mathbf{X}^T\mathbf{X}\vec{\alpha}_1) \tag{3.23}$$

$$var(\vec{y}_1) = \vec{\alpha}_1^T\left(\frac{1}{m-1}\mathbf{X}^T\mathbf{X}\right)\vec{\alpha}_1, \tag{3.24}$$

onde a matriz de covariância está identificada entre parênteses. Assim, a função a ser maximizada pode ser escrita como

$$var(\vec{y}_1) = \vec{\alpha}_1^T C_x \vec{\alpha}_1. \tag{3.25}$$

Para encontrar o máximo desta função sujeita ao vínculo 3.14, pode-se utilizar o método dos *multiplicadores de Lagrange*, utilizado em cálculos de otimização de funções sob determinadas restrições. Por exemplo, uma função $f(x)$ sujeita a um vínculo $g(x) = 0$, pode ser reescrita como uma nova função L a ser maximizada:

$$L(x,\lambda) = f(x) + \lambda(0 - g(x)), \tag{3.26}$$

onde λ é um multiplicador de Lagrange, e a igualdade do vínculo aparece na forma da subtração entre parênteses. A maximização pode ser feita tomando-se a derivada de L e igualando-a a zero:

$$\nabla L(x,\lambda) = 0. \tag{3.27}$$

A forma geral para a maximização de uma função de uma única variável sujeita a p restrições é

$$\nabla L(x,\lambda_1,\lambda_2\cdots\lambda_p) = \nabla(f(x) + \sum_i^p \lambda_i(c_i - g_i(x))) = 0, \tag{3.28}$$

onde $f(x)$ está sujeita aos múltiplos vínculos $g_i(x) = c_i$, e os λ_i são os multiplicadores de Lagrange. Com isso, reescrevemos a variância a ser maximizada sob o vínculo 3.14 como

$$var(\vec{y}_1) = \vec{\alpha}_1^T C_x \vec{\alpha}_1 + \lambda(1 - \vec{\alpha}_1^T \vec{\alpha}_1). \tag{3.29}$$

Então, derivando o lado direito em relação a α_1^T e igualando a zero, obtém-se

$$C_x \vec{\alpha}_1 - \lambda \vec{\alpha}_1 = 0, \tag{3.30}$$

ou ainda,

$$C_x \vec{\alpha}_1 = \lambda \vec{\alpha}_1. \tag{3.31}$$

Assim, λ é um autovalor da matriz de covariância e $\vec{\alpha}_1$ seu autovetor associado. Usando este resultado na eq. 3.25, obtém-se

$$var(\vec{y}_1) = \vec{\alpha}_1^T \lambda \vec{\alpha}_1 \tag{3.32}$$

$$var(\vec{y}_1) = \lambda \vec{\alpha}_1^T \vec{\alpha}_1, \tag{3.33}$$

e, portanto,

$$var(\vec{y}_1) = \lambda. \tag{3.34}$$

Para que a $var(\vec{y}_1)$ seja máxima, λ deve ter o maior valor possível. Assim, obtemos que a variância da primeira componente principal é o maior autovalor da matriz de covariância, e com a direção dada pelo autovetor $\vec{\alpha}_1$ associado a este autovalor. Agora, deve-se procurar o vetor $\vec{\alpha}_2$ que maximiza a $var(\vec{y}_2)$ para que ele seja a segunda componente principal, isto é,

$$arg_{\vec{\alpha}_2} \left[max \left[var(\vec{y}_2) \right] \right]. \tag{3.35}$$

Para isso, escrevemos a variância de $\vec{y}_2$ como

$$var(\vec{y}_2) = \vec{\alpha}_2^T C_x \vec{\alpha}_2, \tag{3.36}$$

mas agora sujeita também à restrição 3.15, tal que

$$\begin{cases} \vec{\alpha}_2^T \vec{\alpha}_2 = 1 \\ \vec{\alpha}_2^T \vec{\alpha}_1 = 0 \end{cases} \tag{3.37}$$

Uma escolha qualquer para a maximização da segunda componente principal poderia retornar qualquer outra direção, mas procuramos aquela que seja ortogonal à primeira. Então, utilizado a forma dos multiplicadores de Lagrange, obtém-se que

$$var(\vec{y}_2) = \vec{\alpha}_2^T C_x \vec{\alpha}_2 + \lambda(1 - \vec{\alpha}_2^T \vec{\alpha}_2) + \phi(0 - \vec{\alpha}_2^T \vec{\alpha}_1) \tag{3.38}$$

onde λ e ϕ são os multiplicadores de Lagrange. Diferenciando em relação a $\vec{\alpha}_2$ e igualando a zero esta expressão, obtém-se

$$C_x \vec{\alpha}_2 - \lambda \vec{\alpha}_2 - \phi \vec{\alpha}_1 = 0. \tag{3.39}$$

Multiplicando esta expressão por $\vec{\alpha}_1^T$, obtém-se

$$\vec{\alpha}_1^T C_x \vec{\alpha}_2 - \lambda \vec{\alpha}_1^T \vec{\alpha}_2 - \phi \vec{\alpha}_1^T \vec{\alpha}_1 = 0. \tag{3.40}$$

Dado que os primeiros dois termos são nulos e $\alpha_1^T \alpha_1 = 1$, então $\phi = 0$. Portanto, pelo resultado da eq. 3.39

$$C_x \vec{\alpha}_2 - \lambda \vec{\alpha}_2 = 0, \tag{3.41}$$

ou

$$C_x \vec{\alpha}_2 = \lambda \vec{\alpha}_2. \tag{3.42}$$

Novamente, λ é um autovalor de C_x e $\vec{\alpha}_2$ seu autovetor associado. Substituindo este resultado na eq. 3.36, obtém-se que

$$var(\vec{y}_2) = \vec{\alpha}_2^T \lambda \vec{\alpha}_2 \tag{3.43}$$

$$var(\vec{y}_2) = \lambda \vec{\alpha}_2^T \vec{\alpha}_2, \tag{3.44}$$

e, finalmente,

$$var(\vec{y}_2) = \lambda. \tag{3.45}$$

Então, para que a $var(\vec{y}_2)$ seja máxima, λ deve ser o mais alto em valor quanto possível. Como o primeiro autovalor já foi utilizado, λ deve ser o segundo maior autovalor da matriz de covariância, e seu autovetor associado dá a direção da segunda componente principal. A extensão para todas as outras componentes é análoga e a variância de uma componente principal pode ser escrita como

$$var(\vec{y}_j) = \lambda_j, \tag{3.46}$$

onde os λ_j são os autovalores da matriz de covariância de $\mathbf{X}$. Por fim, a diagonalização de C_x a partir de seus autovalores e autovetores provê uma base ortogonal e com vetores direcionados às maiores variações de uma dada distribuição de dados $\mathbf{X}$ no espaço de n-variáveis, tal que

$$C_x \vec{\alpha}_j = \lambda_j \vec{\alpha}_j \tag{3.47}$$

onde $\vec{\alpha}_j$ é um autovetor de importância j na base $\vec{\alpha} = \{\vec{\alpha}_1, \vec{\alpha}_2, \cdots, \vec{\alpha}_j, \cdots, \vec{\alpha}_n\}$. Os autovalores λ_j escolhidos em ordem decrescente ($\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_j \geq \cdots \geq \lambda_n \geq 0$), representam a variância correspondente ao longo dos autovetores associados. É possível ainda denotar os autovalores normalizados na forma

$$\lambda_{norm}^j = \frac{\lambda_j}{\sum_{i=1}^n \lambda_i}, \tag{3.48}$$

onde cada autovalor normalizado "explica" o quanto da variância está ao longo do seu autovetor associado. A representação no espaço das componentes principais é a projeção dos dados originais nos autovetores escolhidos. Cada j-ésima componente principal $\vec{y}_j$ é definida como um vetor coluna que é a multiplicação da matriz $\mathbf{X}$ pelo j-ésimo autovetor $\vec{\alpha}_j$, ou seja,

$$\vec{y}_j = \mathbf{X}\vec{\alpha}_j. \tag{3.49}$$

Os maiores autovalores associados a estes autovetores, então, possuem maior significância, em termos da variância, o que leva à possibilidade de representar os dados num espaço de menor dimensão.

Todos os cálculos mostrados nesta seção são relativos à definição e obtenção das componentes principais, do ponto de vista matemático. Porém, os passos para a implementação do código seguem apenas o seguinte roteiro:

Dada uma matrix $\mathbf{X}$ de dimensão $m \times n$ (m medidas de n-variáveis):

1. Centraliza-se a matriz $\mathbf{X}$ em torno da média 0;
2. Calcula-se a matriz de covariância de $\mathbf{X}$ na forma da equação 3.18;
3. Calcula-se os autovalores e autovetores da matriz de covariância;
4. Seleciona-se os autovetores associados aos maiores autovalores pelo critério da equação 3.48;
5. Projeta-se a matriz $\mathbf{X}$ sobre estes autovetores. Os vetores resultantes serão chamados Componentes Principais.

Assim, um problema originalmente com dez variáveis (isto é, que existe em dez dimensões) pode ter apenas os dois maiores autovalores da sua matriz de covariância carregando 80% da variância total, tal que a visualização dos dados possa ser simplificada de dez para duas dimensões. A perda de 20% da variância certamente significa alguma coisa, mas espera-se que o ganho obtido por esta redução da dimensionalidade seja suficientemente compensatório.

3.4 Agrupamento *K-Means*

K-Means é um método de *machine learning* não supervisionado cujo objetivo é o de agrupar dados por similaridade, em torno de um valor médio chamado $centroide$. Em termos numéricos, essa semalhança pode ser estabelecida como sendo a distância euclideana entre pontos no espaço

(*n*-dimensional), isto é, pontos próximos são semelhantes e pontos distantes não. Cada grupo formado define uma classe, com significado estabelecido pelo eixos das variáveis. Por exemplo: no gráfico de compras realizadas por clientes de uma loja, um dos eixos pode indicar os tipos de mercadoria vendidos, enquanto outro eixo significa a faixa etária dos compradores. Um método de clusterização aplicado a esse problema pode mostrar a relação de compradores de determinada faixa etária com a venda de um tipo específico de mercadoria. Pode ainda mostrar que pessoas da classe "20 a 40 anos" consomem mais objetos eletrônicos que pessoas da classe "50 a 70 anos", por exemplo, ou que as mesmas pessoas que compram teclados também compram *mouses*, de forma que seja possível promover anúncios e ofertas personalizadas para cada perfil de cliente. Esse tipo de *insight* pode ser obtido quando esses padrões existem e quando os dados podem ser interpretados corretamente pelo método em questão.

A Figura 3.3 mostra, novamente, o conjunto de dados *Iris Flower* (já referenciado no Capítulo 1), que possui 150 amostras de três espécies da flor íris, indicadas na figura, sendo as variáveis "Comprimento da pétala (cm)" e "Largura da Pétala (cm)". O grupo de amostras da categoria *Setosa*, no canto inferior esquerdo, é bem definido e sem sobreposição de pontos com os outros dois grupos, enquanto as categorias *Versicolor* e *Virginica* possuem um extremo em comum, não sendo possível separar as duas classes linearmente por uma reta, tal que todos os pontos *Versicolor* fiquem de um lado, e todos os pontos *Virginica* fiquem de outro.

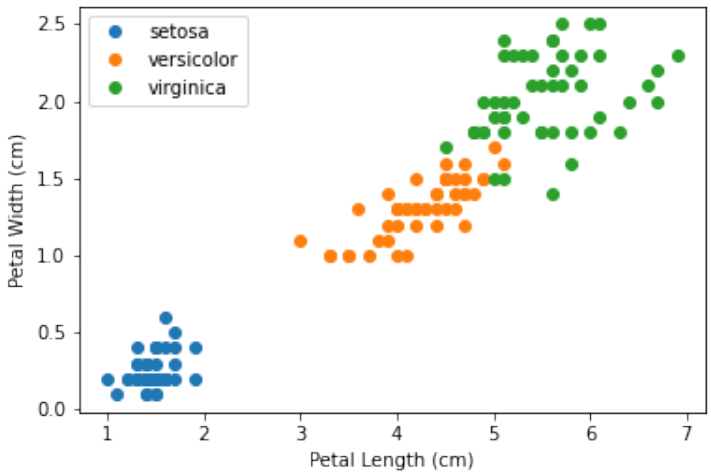

Figura 3.3: Conjunto de dados *Iris Flower*. veja *sklearn.datasets.load_iris.*

A Figura 3.4 mostra o mesmo conjunto de dados sem os rótulos, e a Figura 3.5 mostra os grupos formados após um processo de clusterização usando *K-Means*. É possível notar que a classe *setosa* não sofre nenhuma perda após o agrupamento, devido ao seu estado de grupo isolado, enquanto um limite espacial entre as outras duas classes foi definido. Assim, se

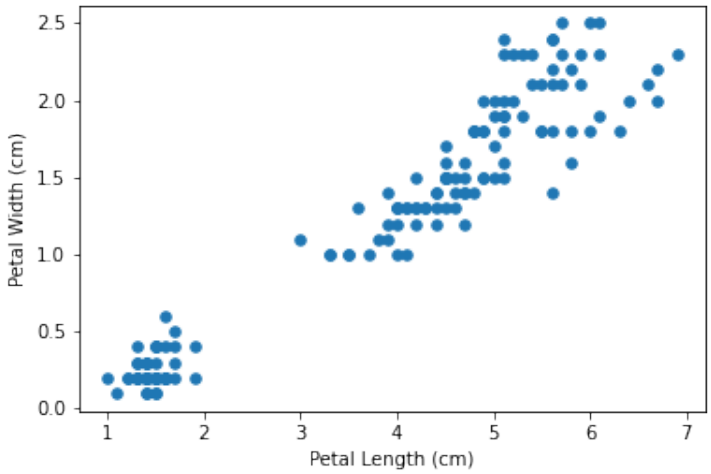

Figura 3.4: Conjunto de dados *Iris Flower* com os rótulos removidos.

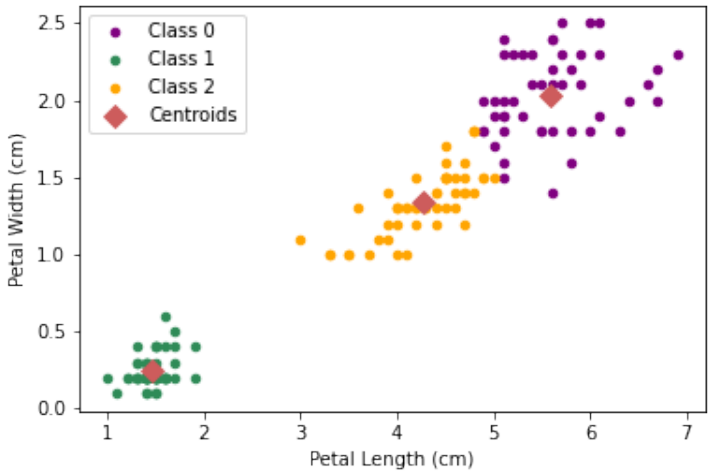

Figura 3.5: Conjunto de dados *Iris Flower* após um processo de clusterização usando *K-Means*. Os centroides dos grupos estão indicados.

comparada à Figura 3.3, esta nova formação de grupos possui um limite rígido, e alguns pontos foram erroneamente classificados, isto é, alguns dados *Versicolor* foram associados à classe 0 (que sabe-se ser *Virginica*) e vice-versa. No processo de agrupamento *K-Means*, o espaço é segmentado por retas de decisão, que definem os limites para o pertencimento a um determinado grupo, como mostrado na Figura 3.7, no final desta seção. Essa forma de clusterização é do tipo *hard*, e não é possível que um dado pertença a grupos distintos simultaneamente, o que favorece o erro nas bordas em distribuições de dados que sofrem sobreposição. Este método é falho para dados com formas mais genéricas, como mostrado na Figura 3.6, e possui melhor desempenho com dados que possuem distribuição grosseiramente esférica. Para contornar esta limitação, existem outras técnicas de clusterização, como a espectral, que será tratada mais adiante.

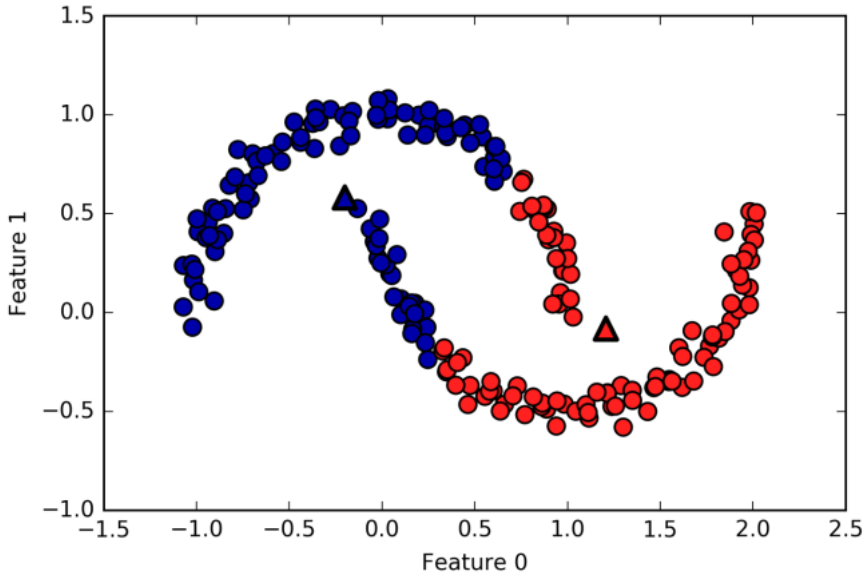

Figura 3.6: K-Means falha em agrupar dados com formas complexas. [9]

Definição dos centroides

A posição dos centroides (pontos fixos) é estabelecida com a definição de uma função objetivo J, que opera sobre as distâncias dos dados a esses pontos fixos inicialmente distribuídos de forma aleatória, cada um representando um grupo que deverá ser formado. Cada dado será associado ao ponto fixo mais próximo a ele, o que é um problema de otimização. Para um conjunto de n dados com coordenadas $\vec{x}_i$, define-se a função objetivo como sendo

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} a_{ij} \|\vec{x}_i - \vec{\mu}_j\|^2. \tag{3.50}$$

Aqui, μ_j indica a posição dos pontos fixos, representantes temporários de cada grupo, e k é a quantidade de grupos. $a_{ij} = 1$ para $j = argmin\ \|\vec{x}_i - \vec{\mu}_j\|^2$ e 0 se não. A minimização de J é feita através de $\nabla J = 0$, e que resulta em

$$\mu_j = \frac{\sum_{i=1}^{n} a_{ij}\vec{x}_i}{\sum_{i=1}^{n} a_{ij}}. \tag{3.51}$$

Ou seja, o valor mínimo de J ocorre quando o ponto fixo equivale ao valor médio das coordenadas dos dados que formam os clusters [40]. Daí vem o nome $K\text{-}Means$ (K-Médias).

Esse algoritmo é um dos mais simples entre os de clusterização e, definida uma quantidade k de grupos, pode ser implementado da seguinte forma:

1. Define-se k centroides (pontos fixos) aleatoriamente distribuídos;

2. Calcula-se a distância dos dados para cada centroide, de modo que os dados são associados ao centroide mais próximo a eles, formando-se k clusters;

3. Calcula-se a média aritmética das coordenadas dos dados de cada cluster (i.e., "centro de massa"). Os centroides são atualizados para estes valores médios;

4. Retorna-se ao passo 2 até a convergência, isto é, até que não haja mais troca de dados entre os clusters.

Esse processo está exemplificado na Figura 3.7. As retas imaginárias indicam o limite espacial para os dados pertencerem a um ou a outro grupo. Note que alguns dados mudam de lado a cada iteração. Isso deixa de ocorrer na "Atualização dos Centroides (3)" (penúltima figura), e o processo é finalizado. A quantidade de iterações é variável para cada problema, e é preciso estabelecer um valor mínimo na implementação do código. Quanto mais bem definidos os grupos, mais rapidamente ocorrerá a convergência. Para o caso de dados sobrepostos, onde não é possível visualizar as limitações de uma dada classe, várias inicializações diferentes devem ser experimentadas, e toma-se como a final aquela que retorna o melhor mínimo local.

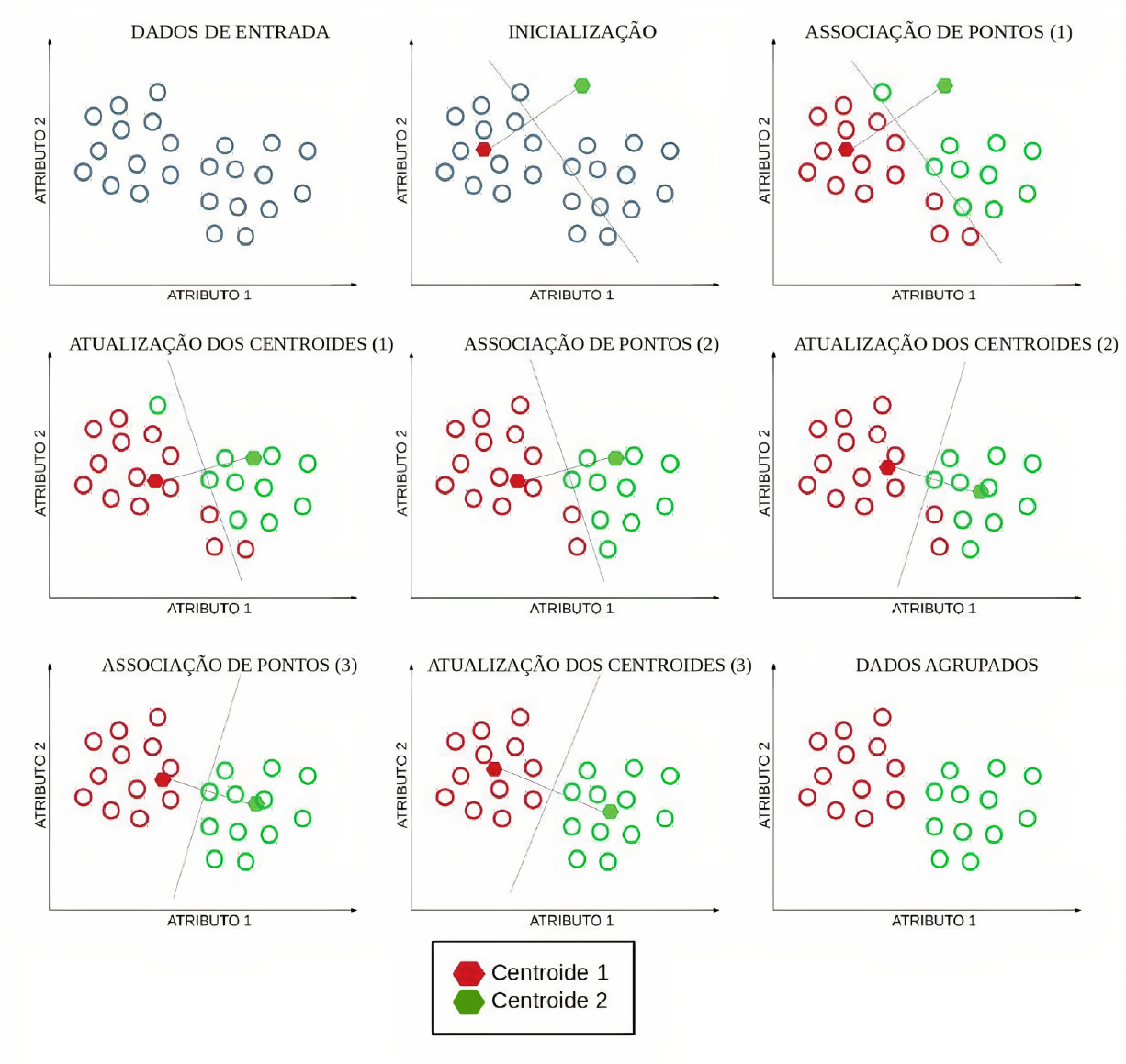

Figura 3.7: Exemplo: Processo de clusterização de uma distribuição de dados com dois grupos a serem formados.

3.5 Análise Topológica de Dados

Homologia Persistente

A Análise Topológica de Dados, do inglês *Topological Data Analysis* (TDA) consiste em um conjunto de técnicas para analisar dados do ponto de vista de sua topologia. Descrevemos o espaço dos dados em função de suas características topológicas através de uma ferramenta chamada *Homologia Persistente*, que opera uma filtração de *Vietoris-Rips* (um cálculo baseado em grafos e no surgimento e desaparecimento de elementos chamados *invariantes topológicos*). Essa filtração considera a existência de uma classe de objetos geométricos, os *simplexos* (Figura 3.8). A união de vários simplexos é chamada *complexo simplicial*, em que os simplexos podem conectar-se uns aos outros através de vértices, arestas ou faces. A Figura 3.9 mostra um

complexo simplicial formado por simplexos diversos. Os triângulos não preenchidos (Ω_1, Ω_2) são classificados como buracos e podem ser definidos a partir dos *números de Betti* ($b_n(Y)$, $n \in \mathbb{Z}$), que referem-se a buracos n-dimensionais no espaço Y. O número de *Betti* b_0 conta o número de componentes conectados, enquanto b_1 conta o número de *loops*. Assim, se Y é um círculo, $b_1(Y) = 1$. Esses objetos são invariantes topológicos em Y.

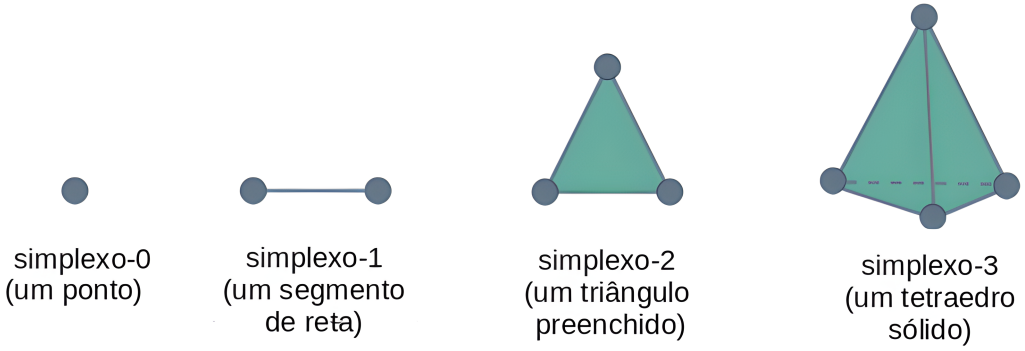

Figura 3.8: Um simplexo é a generalização de um triângulo para uma dimensão arbitrária. [41]

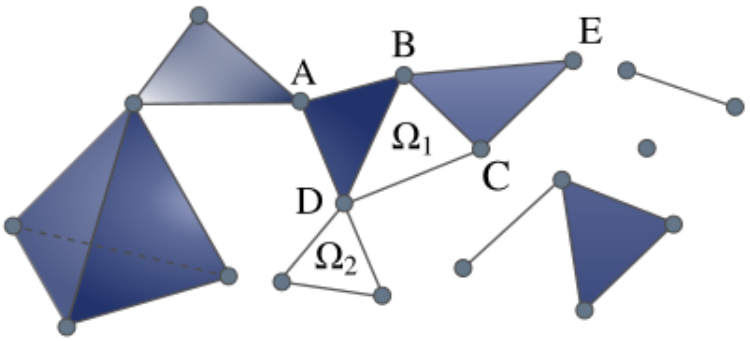

Figura 3.9: Complexo simplicial formado por 19 vértices (simplexo-0), 24 arestas (simplexo-1), 8 faces triangulares (simplexo-2) e 1 tetraedro sólido (simplexo-3). [41]

Para o objetivo desse trabalho, o espaço que queremos mapear é o das configurações de Monte Carlo calculadas sob diversas temperaturas, que serão sempre referencidas por "X". A filtração de *Vietoris-Rips* consiste em um mapeamento dos dados X através da noção de complexos simpliciais, e que ocorre em um espaço munido de um parâmetro de proximidade ε, que pode ser definido como o raio de discos inicialmente centrados em todas as amostras de X. Com a variação de ε, surgem os invariantes topológicos (por simplicidade, trataremos apenas de b_0 (componentes conectados) ou b_1 (*loops*)). Para cada invariante topológico será definido um *par persistente* (b,d) em que b indica o momento de surgimento e d o momento de desaparecimento do invariante topológico. Na Figura 3.10, o *loop* Ω_1 surge em $b = \varepsilon_1$ e desaparece em $d = \varepsilon_3$; enquanto o *loop* Ω_2 surge em $b = \varepsilon_2$ e desaparece em $d = \varepsilon_4$. A partir da definição do tempo de vida ($|d-b|$) dos invariantes topológicos, pode-se construir um *diagrama de persistência* D. Assim, para cada conjunto de dados X há um diagrama de persistência D_X, tal que para dois conjuntos de dados distintos (X_1, X_2), os seus respectivos diagramas de persistência (D_{X_1}, D_{X_2}) devem ser topologicamente diferentes.

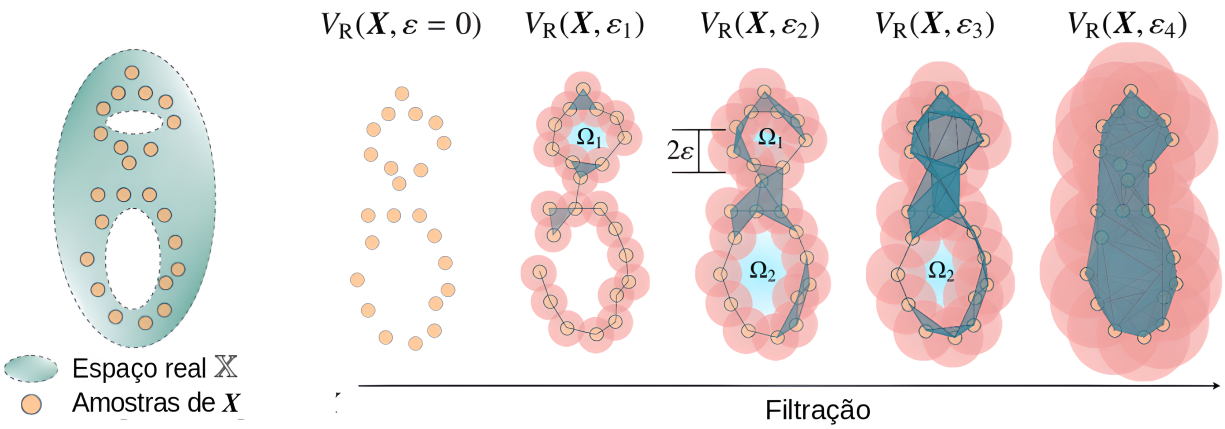

Figura 3.10: Filtração de *Vietoris-Rips* para um conjunto de amostras X. A variação topológica em função de ε é mostrada na figura. [41]

Define-se uma distância d, uma métrica bem estabelecida (*Bottleneck, p-Wasserstein*) que expresse a desigualdade entre dois grupos de amostras, de tal forma que

$$\tilde{d}(X_1, X_2) = d(D_{X_1}, D_{X_2}). \tag{3.52}$$

Assim, para um conjunto de dados $(X_1, X_2, \ldots, X_n)$, é possível construir uma matriz de distâncias M_X, com elementos $m_{ij} = \tilde{d}(X_i, X_j)$. Portanto, quanto mais similares as amostras X_i e X_j, menor o valor de m_{ij}. A matriz M_X pode ser usada para fins de clusterização, com o objetivo de separar amostras topologicamente diferentes em grupos distintos.

Clusterização espectral

Como mencionado anteriormente, técnicas de clusterização como *K-means* não são eficientes em distribuições de dados com formas complexas. A clusterização espectral é uma técnica que permite que a matriz L_G de um grafo (conjunto de vértices conectados por arestas), seja analisada a partir de seus autovalores e, sob determinados critérios, grupos de nós adjacentes possam ser agrupados [42].

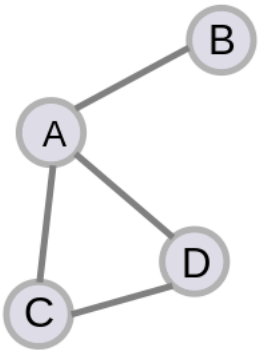

Figura 3.11: Grafo com 4 vértices e 4 arestas.

Pode-se construir um grafo de uma matriz X utilizando técnicas como *K-Nearest Neighbors*, em que cada nó é associado aos k elementos mais próximos; ou a partir da construção de uma matriz de similaridade S de elementos

$$s_{ij} = \exp\left(\frac{-\left\|s_i - s_j\right\|^2}{2\sigma^2}\right), \tag{3.53}$$

na qual $s_{ij} \in [0, 1]$, e o parâmetro σ controla a largura da distribuição Gaussiana. Assim, todos os elementos estão conectados por um peso dado por s_{ij}, que mede o grau de similaridade entre dois elementos, e a autossimilaridade $s_{ii} = 1$. Depois de transformar uma matriz X em um grafo, é necessário calcular a sua matriz Laplaciana L_G, definida por

$$L_G = D - A, \tag{3.54}$$

em que D é uma matriz diagonal que conta o número de conexões feito por cada vértice, e a matriz de adjacências A contém entradas que representam a presença (1) ou ausência (0) de uma aresta entre cada par de vértices. A Figura 3.12 mostra as matrizes D e A relacionadas ao grafo da Figura 3.11.

D =

	A	B	C	D
A	3	0	0	0
B	0	1	0	0
C	0	0	2	0
D	0	0	0	2

A =

	A	B	C	D
A	0	1	1	1
B	1	0	0	0
C	1	0	0	1
D	1	0	1	0

Figura 3.12: Matrizes para o cálculo de L_G relativas ao grafo da Figura 3.11.

A matriz L_G possui algumas propriedades interessantes. Os seus autovalores devem ser analisados em ordem crescente: os nulos contam o número de componentes conectados. Isso significa que, para o grafo da Figura 3.11, que está inteiramente conectado, há apenas um autovalor nulo. Se a aresta que conecta A e B não existisse, esse valor seria igual a dois, pois os vértices ADC continuariam formando um componente conectado, enquanto o vértice B seria um componente conectado sozinho. Quando há apenas um autovalor nulo, o autovetor associado ao primeiro menor autovalor diferente de zero é chamado *vetor de Fiedler*. Esse autovetor pode ser usado para fazer uma bipartição no grafo, separando-o em dois grupos, mesmo que em formas mais complexas, como o da Figura 3.13. Em geral, p autovetores de L_G ordenados segundo seus correspondentes autovalores (do menor para o maior, excluindo-se o primeiro), são utilizados para particionar os dados em p grupos.

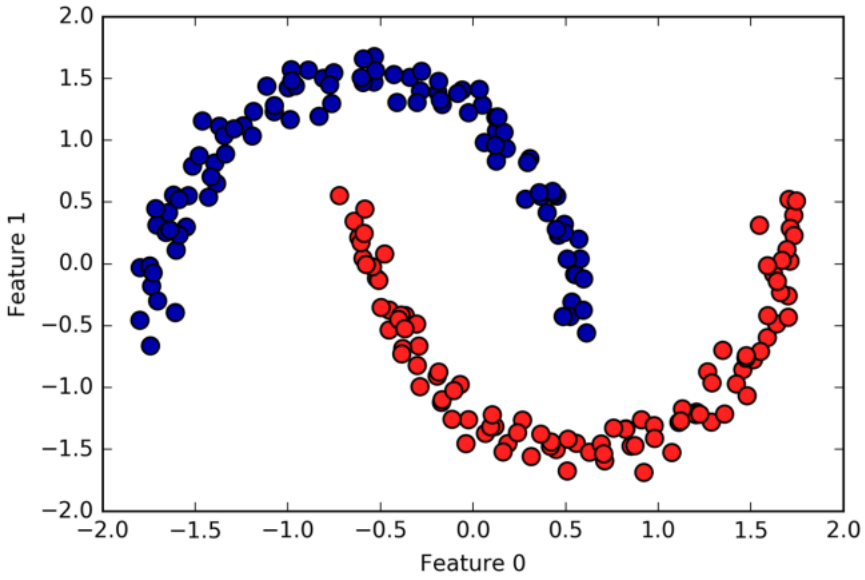

Figura 3.13: Clusterização espectral para dados distribuídos em forma de semicírculo. Essa técnica permite a correta identifação de formas não linearmente separáveis. [9]

Utilizaremos essas técnicas em combinação para analisar as matrizes de distâncias M_X geradas para os diagramas de persistência de $\{X_1, X_2, \ldots, X_n\}$, seguindo o mesmo roteiro utilizado na referência [26], onde os autores utilizaram uma versão mais relaxada chamada de *Clusterização Espectral Difusa* (*Fuzzy Spectral Clustering*), que une os conceitos principais de clusterização espectral e de *Fuzzy C-Means*. A diferença entre este tipo de agrupamento e aquele feito por *K-Means* é ilustrado na Figura 3.14. A clusterização difusa acrescenta a possibilidade de uma amostra pertencer a todos os grupos com certa probabilidade. Essa medida de "pertencimento" é feita através de uma *função de pertinência*

$$l(l_0, l_1) : \{X_1, X_2, \ldots, X_n\} \rightarrow [0, 1], \tag{3.55}$$

tal que $l_0(X_i)$ dá o grau de pertencimento da amostra X_i ao grupo l_0, e $l_1(X_i)$ dá o grau de pertencimento da amostra X_i ao grupo l_1. A Figura 3.15 ilustra a forma das funções de pertinência associadas à Figura 3.14 (direita), em que cada curva representa o grau de pertencimento aos grupos "roxo" (l_0) e "rosa" (l_1). As funções de pertinência são complementares, isto é, $l_0(X_i) + l_1(X_i) = 1$ para todos os X_i, e o ponto onde estas curvas se cruzam será considerado um ponto crítico.

Essas ideias podem ser compactadas no seguinte roteiro:

1. calcula-se o mapeamento das amostras $\{X_1, X_2, \ldots X_n\}$ para um espaço com características topológicas através da Homologia Persistente;

2. calcula-se os diagramas de persistência D_{X_i} para os mapeamentos de X_i;

3. calcula-se as matrizes de distâncias $M(m_{ij})$ entre diferentes diagramas de persistência, em que $m_{ij} = d(D_{X_i}, D_{X_j})$;

4. as matrizes de distâncias serão usadas para fins de clusterização, em que amostras com diferentes topologias serão discriminadas. A clusterização difusa retornará uma função de similaridade entre os diagramas de persistência, com a qual é possível estimar um ponto crítico.

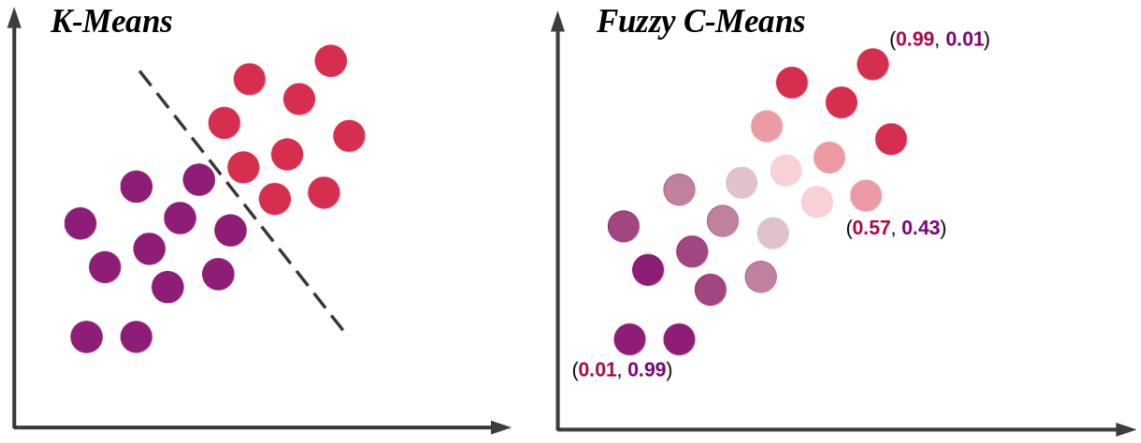

Figura 3.14: No agrupamento feito por *K-Means* (esquerda) há a definição de uma borda rígida que separa os grupos, enquanto em *Fuzzy C-Means* (direita) os dados possuem um certo grau de pertencimento a todos os grupos, indicado pelos valores entre parênteses.

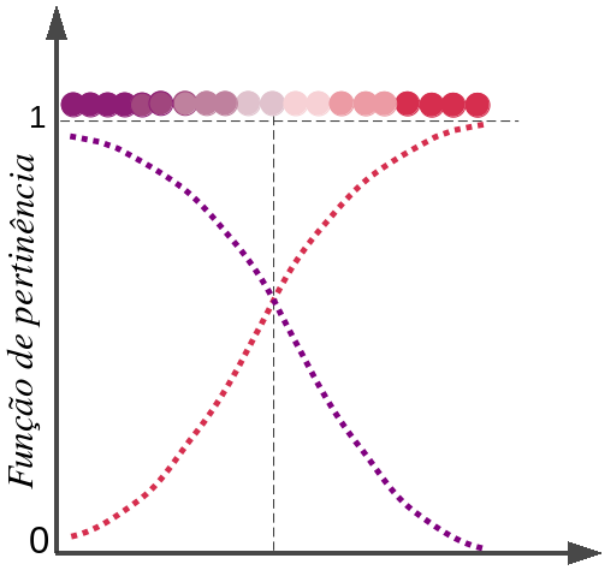

Figura 3.15: Funções de pertinência relativas aos grupos "roxo" e "rosa" ilustrados na Figura 3.14 (direita).

4 Resultados

4.1 Análise das componentes principais: casos q = 3, 4 e 5

Analisamos o modelo de Potts em duas dimensões, através das técnicas descritas no capítulo anterior, utilizando medidas de Monte Carlo no intervalo da temperatura crítica para cada modelo. Para todos os valores de q, dois autovalores se sobressaem em relação aos demais, de forma que duas componentes principais carregam o comportamento mais significativo do sistema, como mostrado, por exemplo, na Figura 4.1 para q = 3. Assim, no que segue, apenas essas componentes principais serão avaliadas.

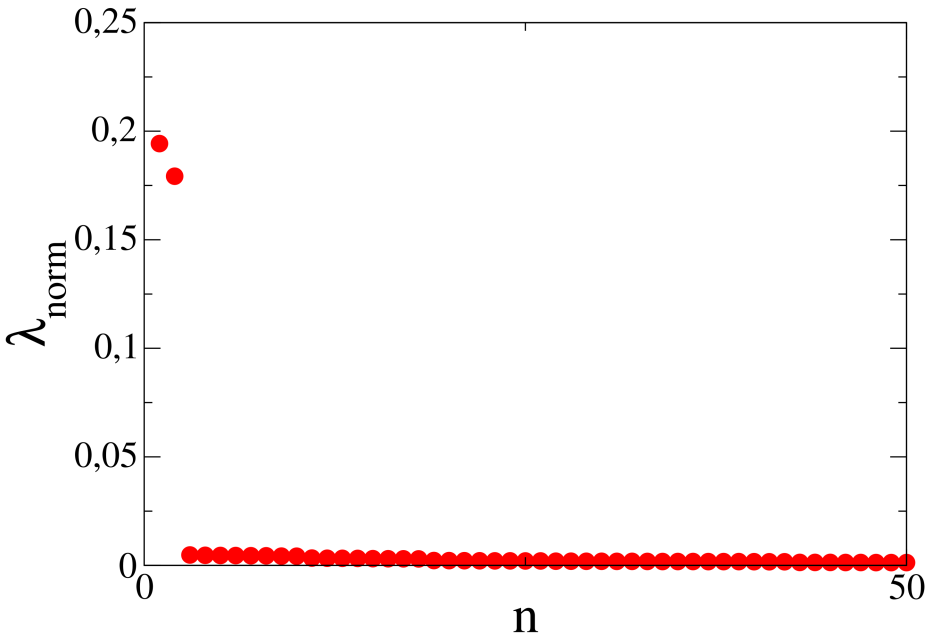

Figura 4.1: 50 primeiros autovalores normalizados obtidos através de PCA para o caso q=3, N=50.

Na Figura 4.2, apresentamos a projeção dos dados no plano das duas componentes principais, y_1 e y_2. Aqui, é importante observar o comportamento da distribuição dos dados quando o sistema passa pela temperatura crítica, $T_c \approx 0.99$. No estado ordenado ($T < T_c$), os pontos encontram-se bem localizados em três regiões equidistantes (verde/azul), enquanto para o estado desordenado ($T > T_c$) (vermelho), há apenas um grupo no centro do espaço. Essa mudança de "topologia" é similar àquela discutida para o modelo de Ising no Capítulo 2. Isto é, os três grupos que se formam para $T < T_c$ nos fornecem informações sobre a simetria do estado fundamental desse modelo, que é triplamente degenerado.

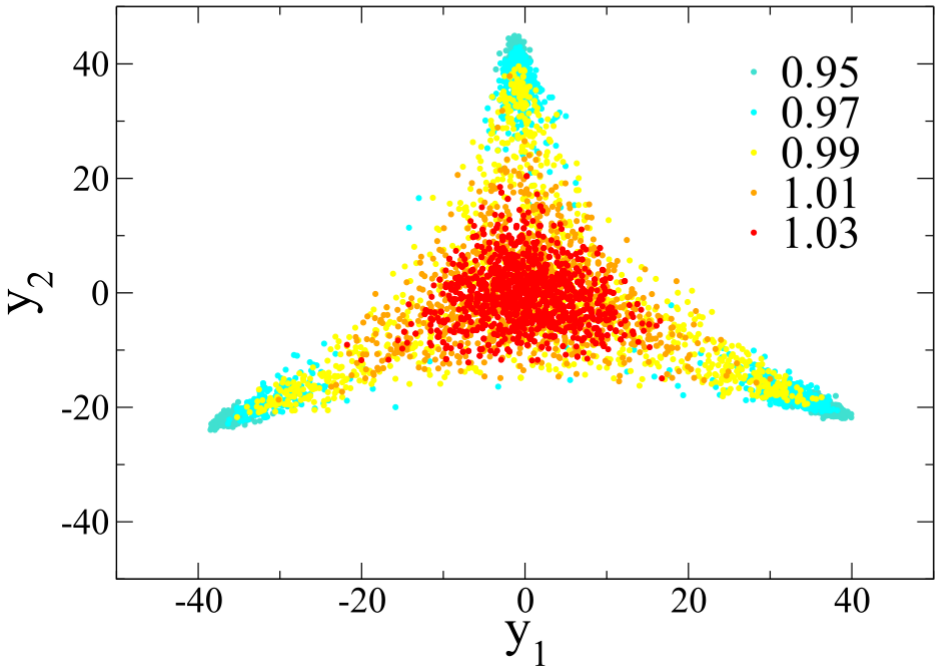

Figura 4.2: Projeção dos dados no plano das duas primeiras componentes principas. As cores indicam a temperatura na qual os dados foram gerados. (y_1, y_2).

Para o caso q=4 obtivemos um resultado análogo. A Figura 4.3 mostra a formação de quatro grupos para temperaturas abaixo da temperatura crítica, que aqui vale $T_c \approx 0.91$, e é também concordante com a simetria da Hamiltoniana para este modelo.

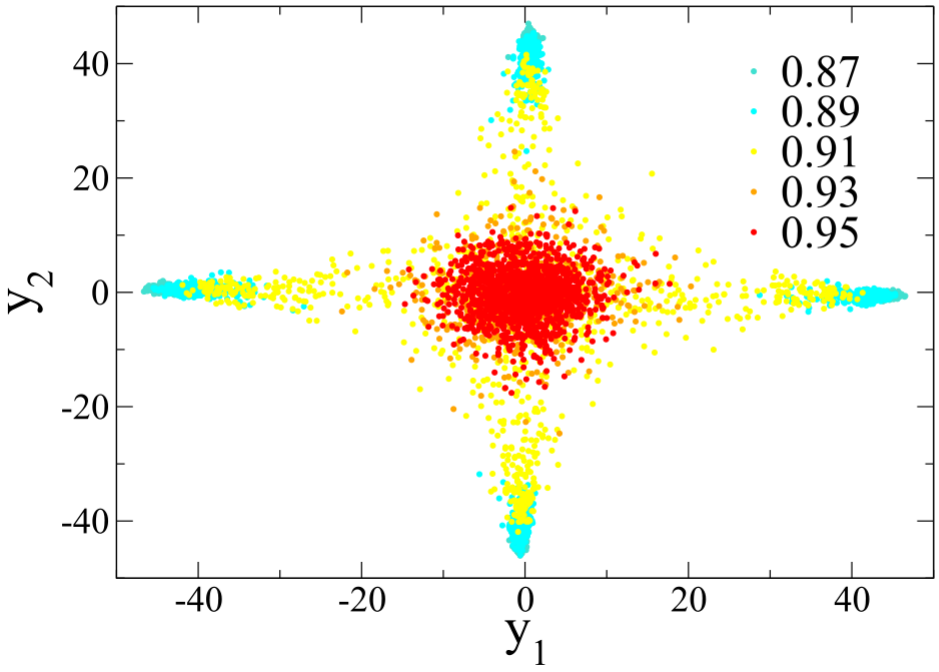

Figura 4.3: Resultados obtidos através de PCA para o caso q=4 e N=50. A projeção nas duas primeiras componentes principas mostra o comportamento dos dados com a variação de temperatura.

O mesmo procedimento é realizado para q = 5, projetando os dados no espaço gerado pelas duas principais componentes. A Figura 4.4 mostra a distribuição dos dados nas proximidades da temperatura crítica. Aqui, os cinco grupos para $T < T_c$ também refletem as simetrias do estado fundamental, que para esse caso é de 5 estados. Para $T > T_c$, esses grupos deixam de existir e os pontos formam um aglomerado no centro do espaço, indicando desordem no sistema. Note que, comparada à Figura 4.2, a Figura 4.4 mostra que há menos pontos intermediários calculados entre as pontas e o centro. De fato, como se espera que para q = 5 o sistema passe por uma transição de fase de primeira ordem, esse comportamento serve como um indicativo da ocorrência de um "salto" em medidas de ordenamento, e que reflete nas componentes principais. Ainda assim, esse não é um fator definitivo e, portanto, faz-se necessário avaliação mais rigorosa quanto a isso.

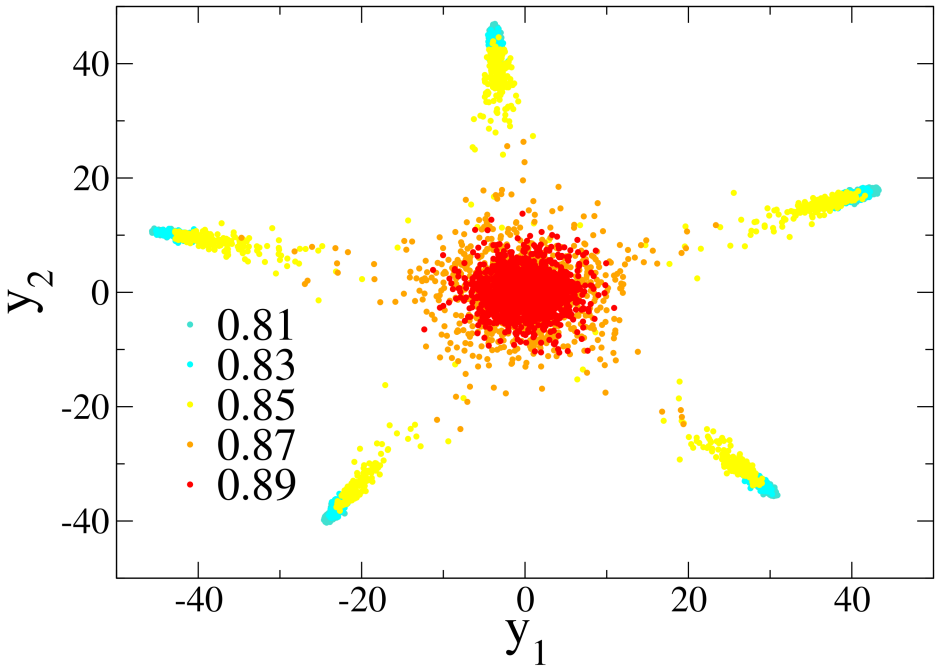

Figura 4.4: Resultados obtidos através de PCA para o caso q=5 e N=50. A projeção nas duas primeiras componentes principas mostra o comportamento dos dados com a variação de temperatura.

Para se investigar mais detalhadamente a ocorrência de transições de fase, é preciso definir grandezas que identifiquem as regiões críticas. Porém, essas grandezas devem depender exclusivamente das componentes principais, para que não haja nenhum viés na análise. Deste modo, definimos as grandezas

$$\Gamma = \frac{\left\langle \sqrt{y_1^2 + y_2^2} \right\rangle}{N} \tag{4.1}$$

e

$$U(\Gamma) = 1 - \frac{\langle \Gamma^4 \rangle}{3 \langle \Gamma^2 \rangle^2}. \tag{4.2}$$

De modo simplificado, o parâmetro Γ define um valor médio dos dados em relação à origem no subespaço de PCA. Desta forma, podemos usá-lo como um análogo do parâmetro de ordem do sistema, como mostrado na Figura 4.5 (a), para o caso q = 3. É possível notar que a região crítica, ou o ponto em que ocorre uma mudança de estado no espaço das componentes principais, é aproximadamente o valor esperado para a temperatura crítica do modelo, $T_c \approx 0.99$. Em vista disso, $U(\Gamma)$ exerce o papel de um análogo dos Cumulantes de Binder. Isto é, o cruzamento de $U(\Gamma)$ para diferentes tamanhos de rede deve indicar a região do ponto crítico. A Figura 4.5 (b) mostra o comportamento de $U(\Gamma)$ para o caso q=3 em diferentes tamanhos de rede, que

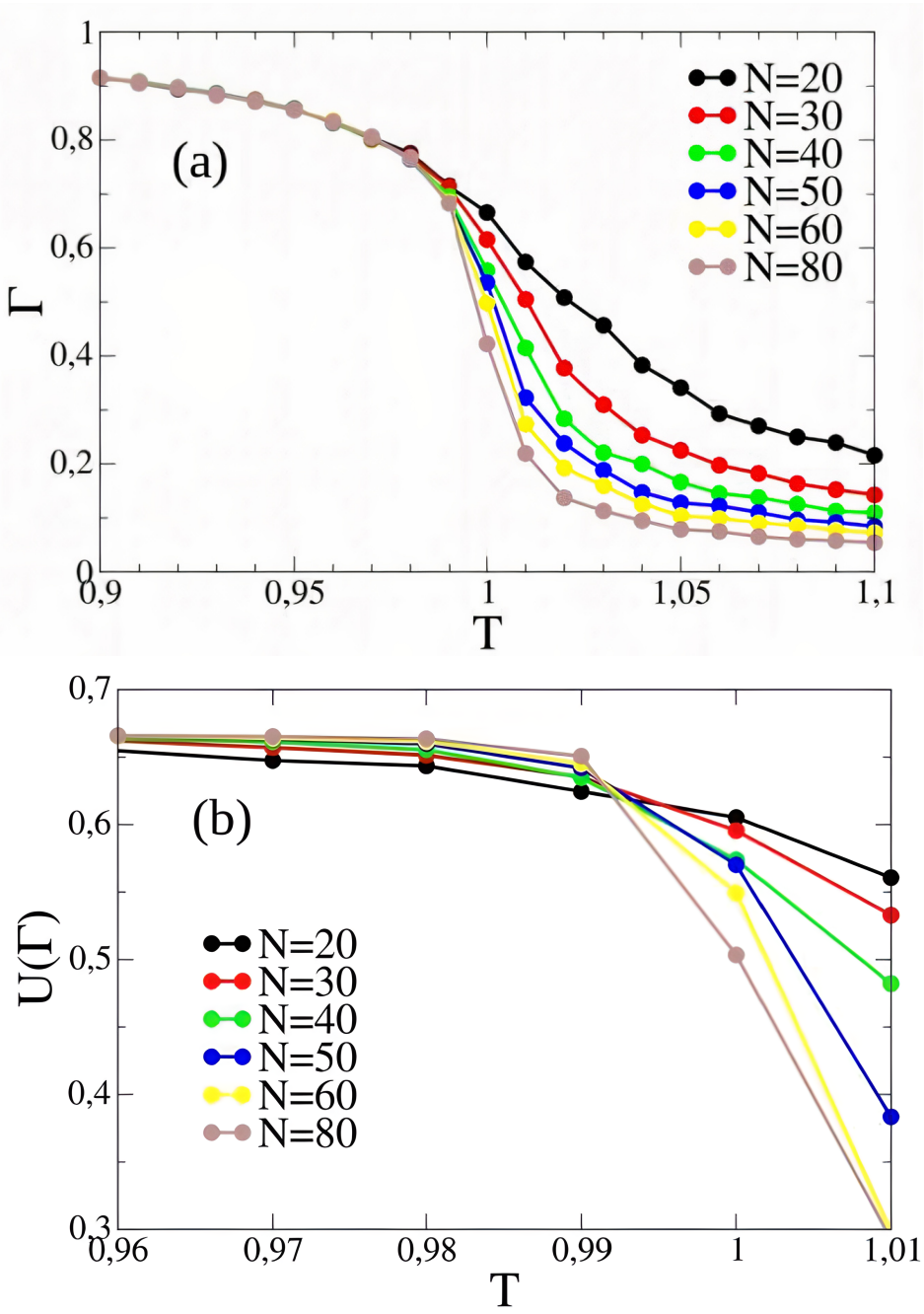

Figura 4.5: Resultados obtidos através de PCA para o caso q=3. (a) Γ versus temperatura e (b) $U(\Gamma)$ versus temperatura. Os tamanhos lineares das redes estão indicados na figura.

concorda com o esperado para este parâmetro.

Semelhantemente, os gráficos para as grandezas Γ e $U(\Gamma)$ em função da temperatura para o caso q = 4, mostrados na Figura 4.6, indicam a temperatura crítica em que o sistema muda de fase, conforme o significado dessas grandezas. Esses resultados também são concordantes com o valor esperado para esse modelo, $T_c \approx 0.91$.

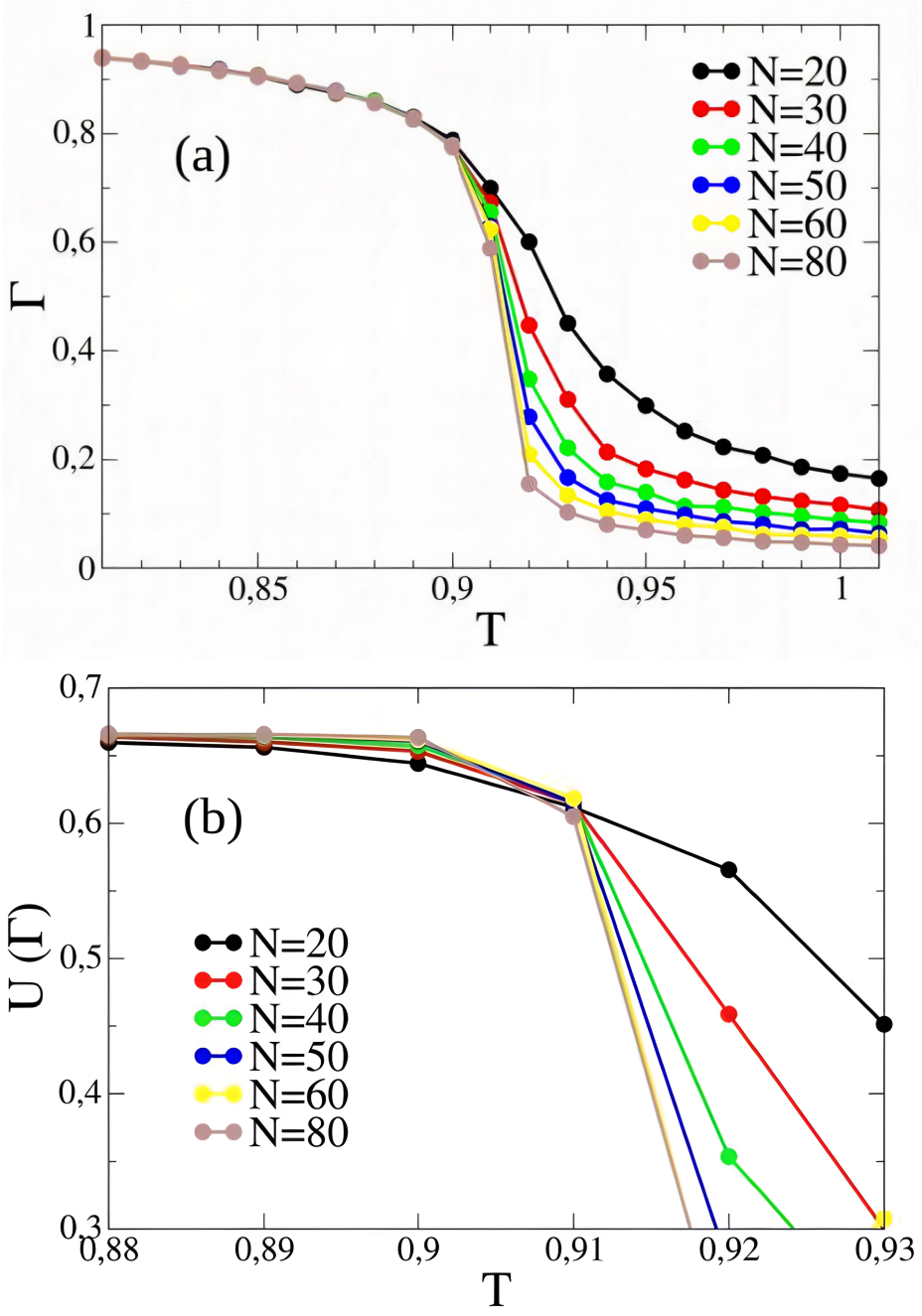

Figura 4.6: Resultados obtidos através de PCA para o caso q=4. **a)** Γ versus temperatura e **b)** $U(\Gamma)$ versus temperatura. Os tamanhos lineares das redes estão indicados na figura.

Já o caso $q = 5$ é mais sutil, por exibir transição de fase de primeira ordem. A Figura 4.7 (a) exibe $\Gamma(T)$ para diferentes tamanhos de rede, onde podemos observar uma mudança abrupta para $T \approx 0,85$; essa descontinuidade fica mais evidente para redes maiores. Diferentemente dos casos anteriores, o comportamento de $U(\Gamma)$, mostrado na Figura 4.7 (b), não mostra com clareza onde as curvas se cruzam, um indicativo de que a transição de fase já não é mais de segunda ordem. Apesar disso, há indícios claros, tanto para $\Gamma(T)$ quando para $U(\Gamma)$, que a transição deve ocorrer para $T_c \approx 0,85$, em bom acordo com a literatura.

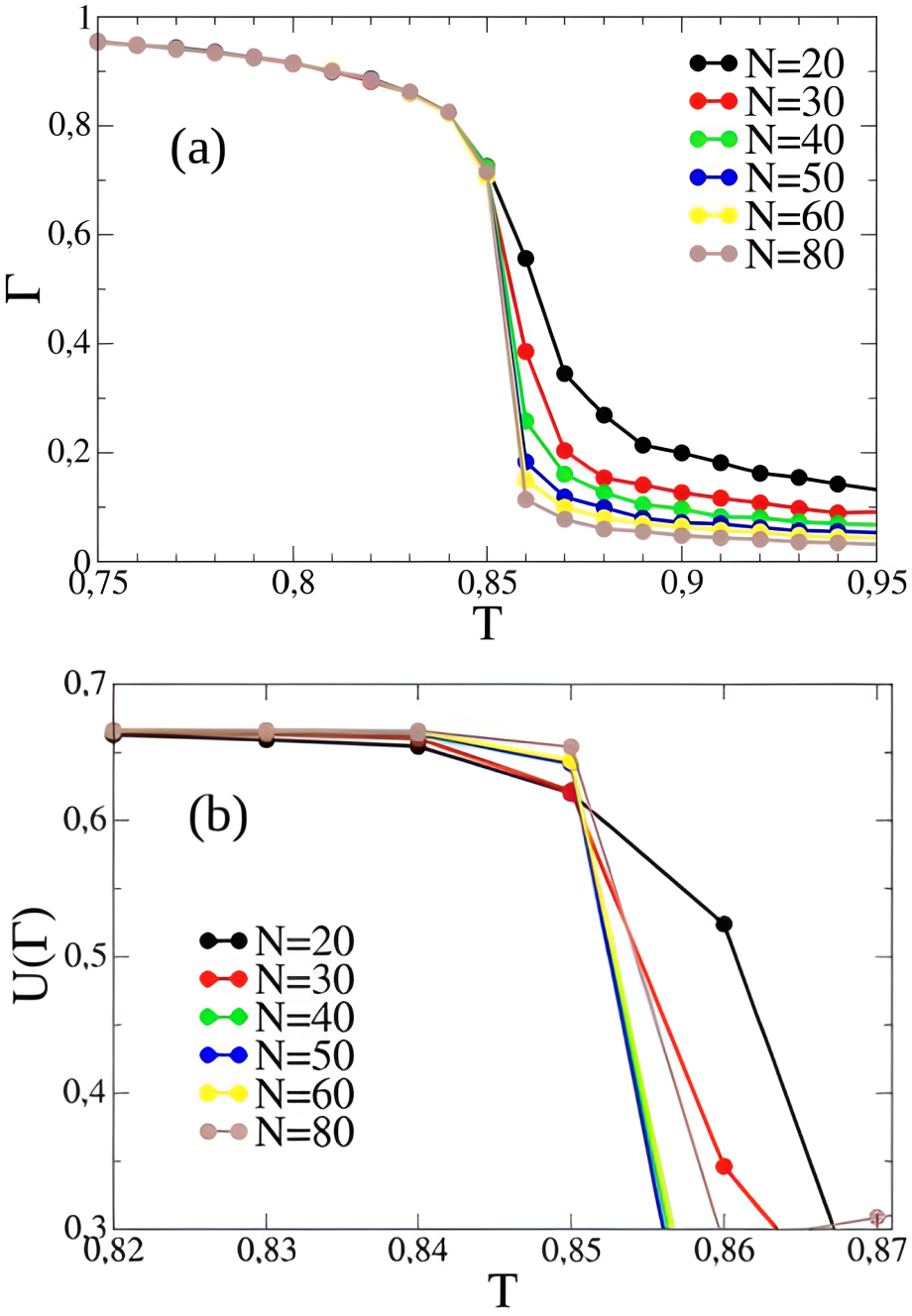

Figura 4.7: Resultados obtidos através de PCA para o caso q=5. (a) Γ versus temperatura e (b) $U(\Gamma)$ versus temperatura. Os tamanhos lineares das redes estão indicados na figura.

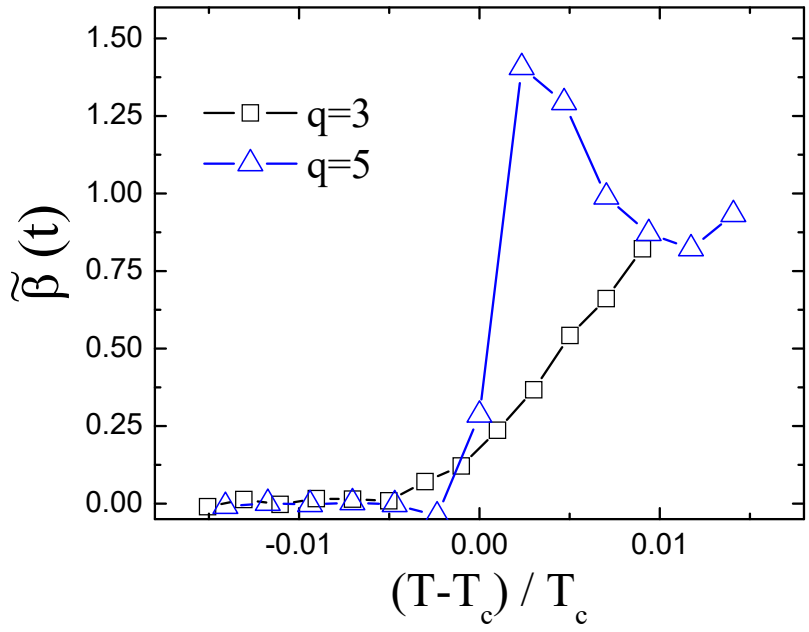

Figura 4.8: Expoente crítico $\tilde{\beta}$ em função da temperatura reduzida $t \equiv (T - T_c)/T_c$ para $q = 3$ e $q = 5$.

Com esses resultados, obtemos as simetrias do estado fundamental do modelo de Potts, além de uma precisão razoável para os valores críticos das transições de fase. Porém, ainda resta uma análise quantitativa que caracterize os tipos de transição de fase.

Os resultados obtidos para a grandeza $U(\Gamma)$ sugerem que, nas proximidades do ponto crítico, deve-se ter

$$\Gamma_L(T) = L^{-\beta/\nu} f[L^{1/\nu}(T - Tc)], \tag{4.3}$$

segundo uma lei de escala, em que β e ν são os expoentes críticos para o parâmetro de ordem e o comprimento de correlação, respectivamente. Pela equação 4.3, $\Gamma_L(T)/L^{-\beta/\nu}$ tende a um valor constante à medida que $T \to T_c$, e é independente de L. Com isso, pode-se definir o expoente crítico

$$\beta/\nu = \tilde{\beta}(t) = -\frac{ln[\Gamma_{L_2}(t)/\Gamma_{L_1}(t)]}{ln(L_2/L_1)}, \tag{4.4}$$

em que $L_2 > L_1$ e $t \equiv (T - T_c)/T_c$. A Figura 4.8 mostra os valores de $\tilde{\beta}(t)$ para q=3 e 5, $L_2 = 80$ e $L_1 = 40$, nas proximidades de T_c. Para $q = 3$, o valor de $\tilde{\beta}(0) \approx 0,17(5)$ concorda razoavelmente com o valor exato $\beta/\nu = 2/15$, e $\tilde{\beta}(t)$ exibe uma curva suave em torno de $t = 0$. Já para $q = 5$, há uma descontinuidade no ponto $\tilde{\beta}(0)$, e, para $T < T_c$, $\tilde{\beta} \approx 0$. Esse resultado é consistente com uma transição de fase de primeira ordem e mostra que a técnica PCA em conjunto com uma teoria bem estabelecida como a de *escala de tamanho finito* também pode indicar a ordem da transição de fase.

4.2 *K-Means* e PCA para q=3 e q=5

Caso q =3

Utilizamos a técnica *K-Means Clustering* em junção aos resultados obtidos através de PCA para fazer uma análise do comportamento dos *clusters* para cada temperatura separadamente. A Figura 4.9 mostra alguns recortes da Figura 4.2, nos quais é possível ver o estado dos grupos para valores distintos de temperatura, calculados abaixo e acima da temperatura crítica. Com essa técnica e com a definição de grupos, pode-se estabelecer para cada um deles um centroide representativo. Esses centroides estão indicados na figura pelos quadrados vermelhos. Assim, pode-se medir o grau de aproximação desses grupos em função da variação de temperatura apenas usando o centroide como referência. Sem as cores, é possível notar que, para temperaturas elevadas, os três grupos iniciais deixam de existir e apenas um grupo central de dados remanesce.

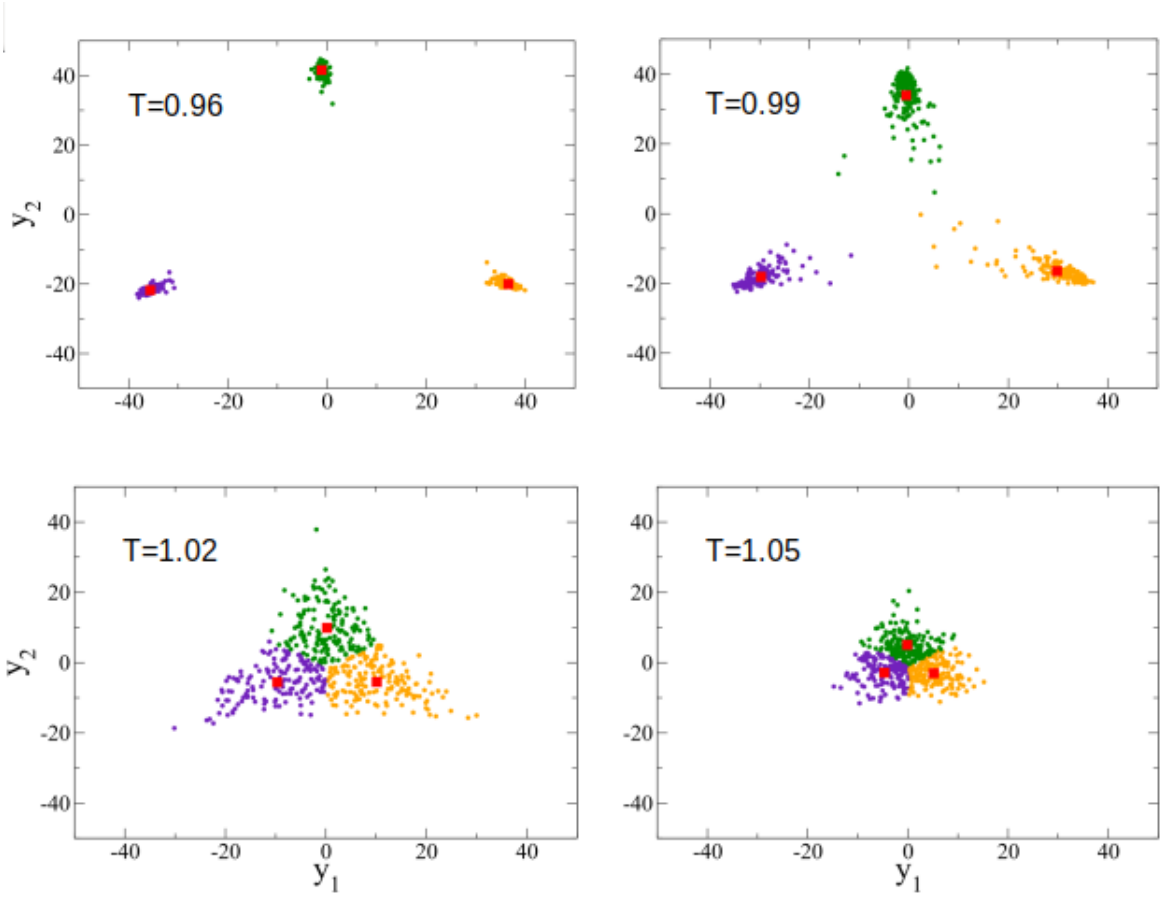

Figura 4.9: Comportamento dos *clusters* para o caso q=3 e N=50 para as temperaturas indicadas nas figuras.

Para estudar esta mudança, calculamos o valor médio das distâncias dos *clusters* até o centro do espaço, tal que $< k > = \sum_{i=1}^{n_k} k_i/n_k$, onde k_i é a distância dos centroides à origem e n_k é o número de grupos, como mostrado na Figura 4.10 (a). Com isso, pode-se fazer uma interpolação polinomial e calcular o ponto de inflexão. Este valor será considerado a temperatura crítica para essa análise por *K-Means*. Com cálculo semelhante para os demais tamanho de rede ($N = 30, 40, 50, 60$ e 80), podemos obter os valores críticos em função de $1/N$, como mostra a Figura 4.10 (b), de forma a encontrar T_c no limite termodinâmico. É possível perceber que este valor, obtido através de uma extrapolação, difere apenas na terceira casa decimal do valor exato previsto para o modelo de Potts de 3 estados (indicado na figura), mesmo com essa aproximação calculada para apenas alguns pontos.

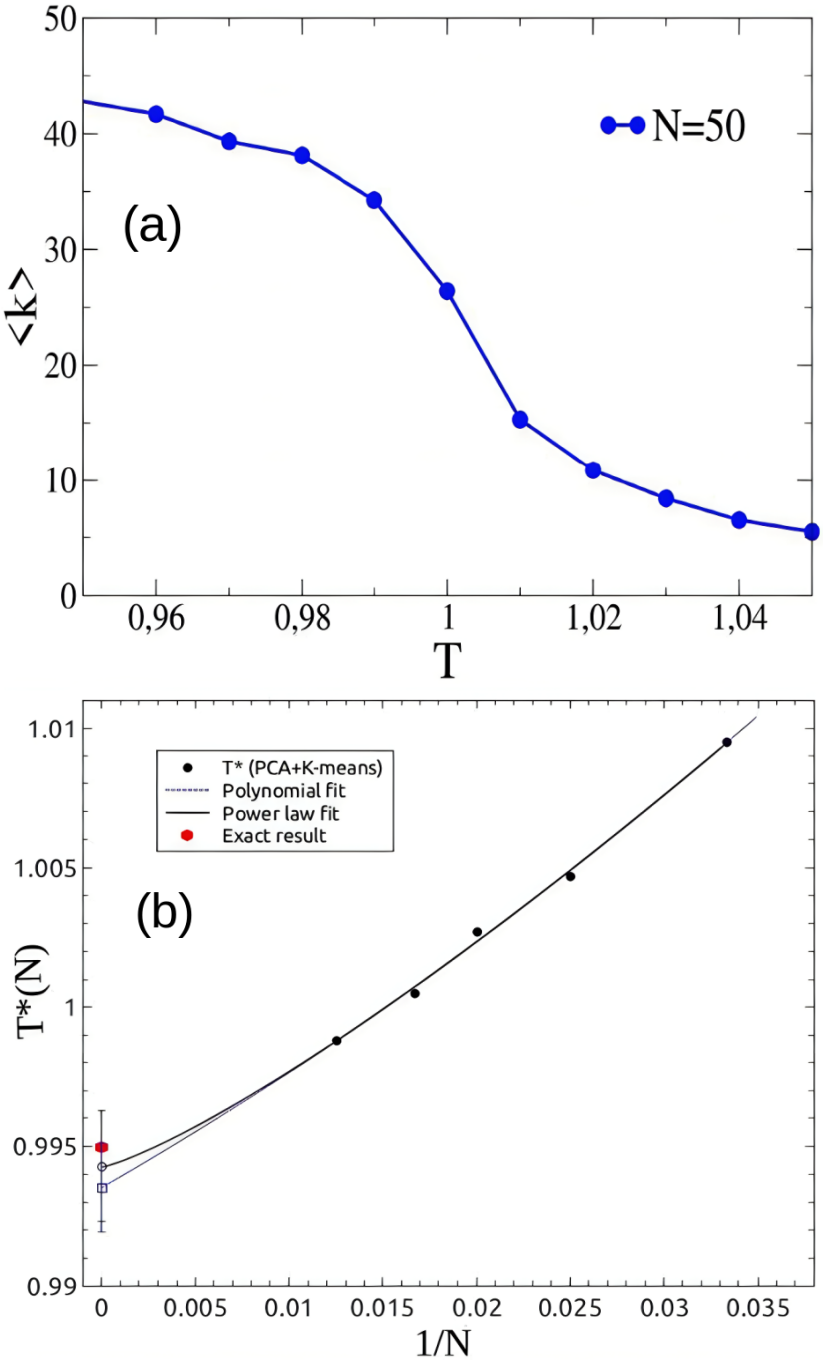

Figura 4.10: (a) Valores médios das distâncias dos centroides até o centro do espaço versus temperatura para o caso q=3, N=50. (b) Uma aproximação para os pontos de inflexão feita para vários tamanhos de rede indica o ponto crítico no limite em que se aumenta o tamanho do sistema.

Caso q = 5

De modo similar, utilizamos a técnica *K-means* associada aos resultados de PCA, para analisar o comportamento dos dados para q = 5 à medida que se aumenta a temperatura do sistema, como mostra a Figura 4.11.

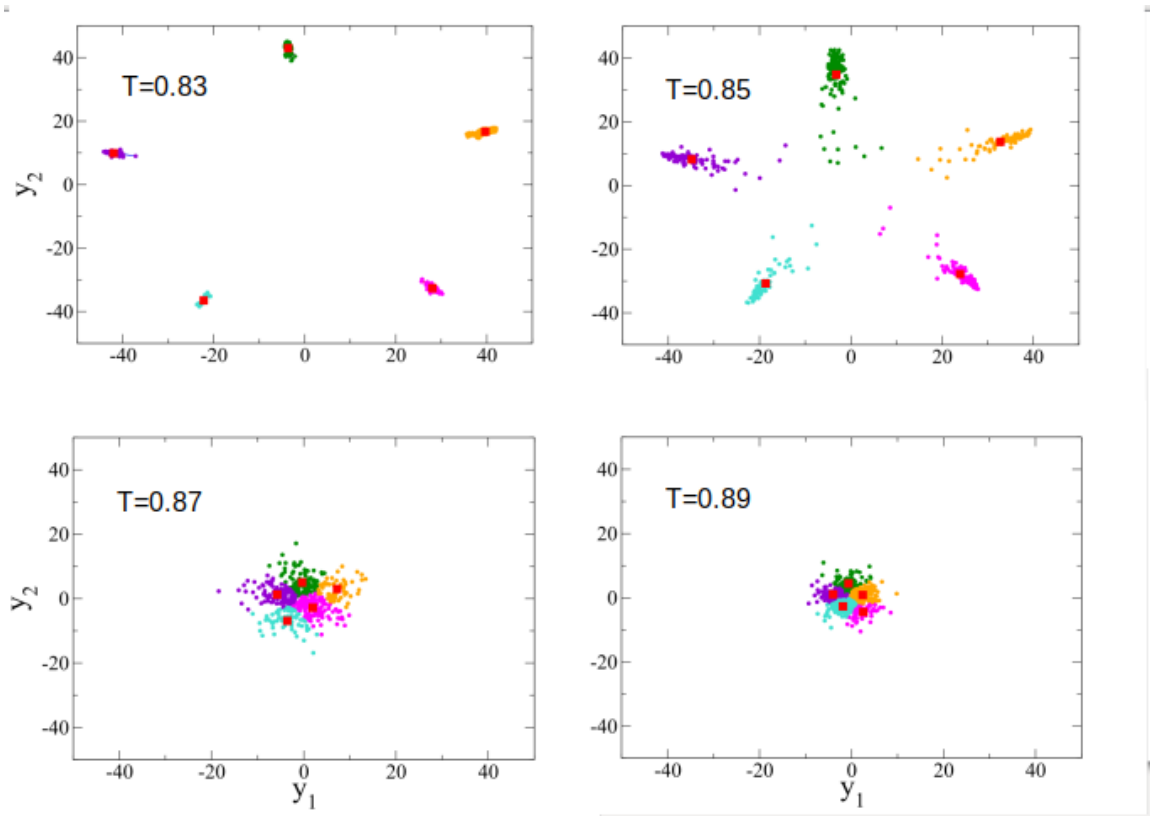

Figura 4.11: Comportamento dos clusters para o caso q=5 e N=50 para as temperaturas indicadas nas figuras.

Assim como para o caso q=3, analisamos as distâncias médias desses centroides até a origem, no espaço das componentes principais (Figura 4.12 (a)). Calculando o ponto de inflexão para a curva de aproximação, e com o mesmo cálculo para os demais tamanhos de rede, obtemos o valor crítico no limite termodinâmico. O valor exato para a temperatura crítica prevista para o modelo de Potts de 5 estados e o valor crítico obtido por extrapolação são mostrados na Figura 4.12 (b). Assim como para o caso q = 3, o erro relativo entre o valor previsto por extrapolação e o valor exato é inferior a 1%.

A análise feita por *K-means* também retorna com grande precisão os valores críticos previstos, concordando, dentro das barras de erro, com os resultados obtidos apenas usando PCA. Esses resultados são capturados exclusivamente através da identificação dos diferentes estados de ordenamento que ocorrem nas matrizes geradas via método de Monte Carlo, sem nenhuma influência externa que indique a existência de uma transição de fase, nem o local de um ponto crítico, confirmando a eficácia de PCA também para o estudo do modelo de Potts.

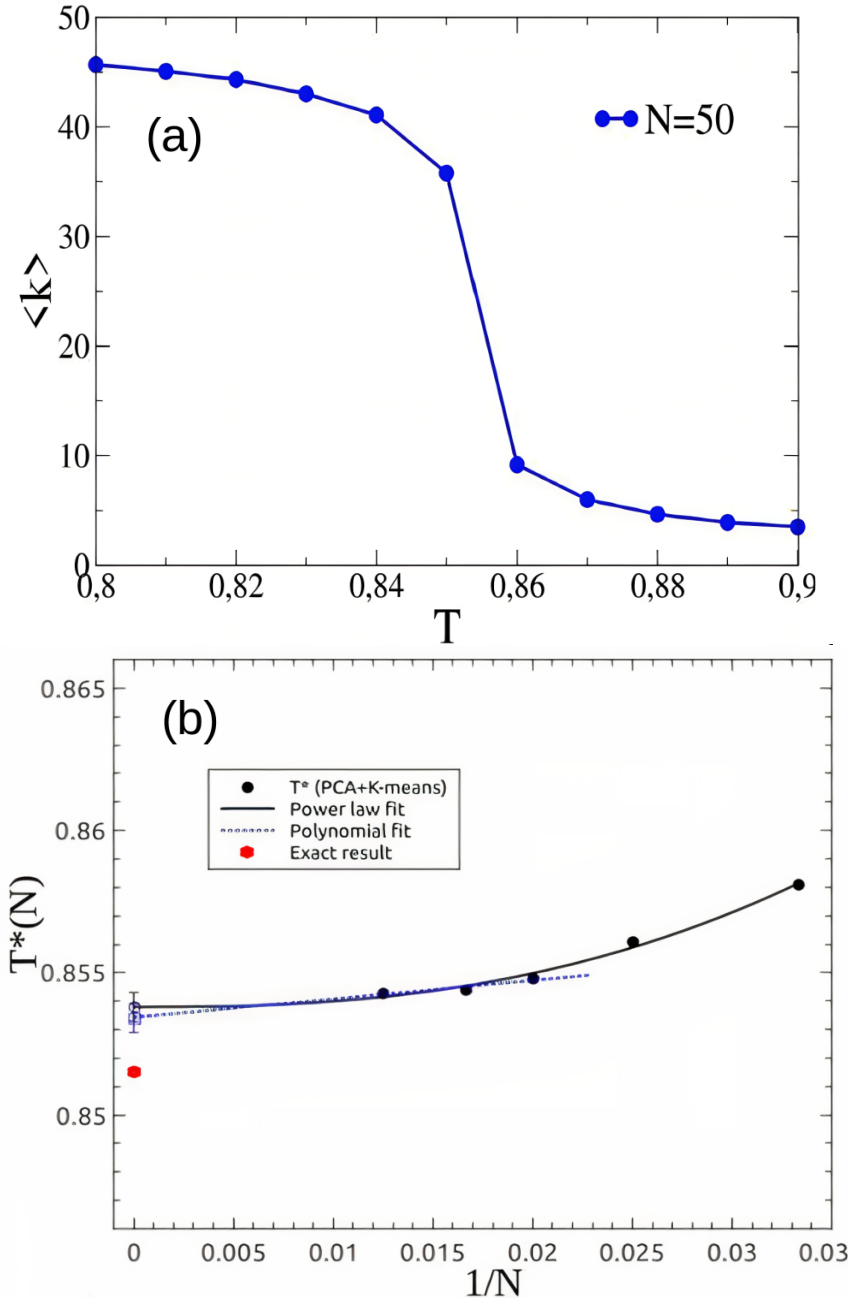

Figura 4.12: (a) Valores médios das distâncias dos centroides até o centro do espaço versus temperatura para o caso q = 5, N = 50. (b) Uma aproximação para os pontos de inflexão feita para vários tamanhos de rede indica o ponto crítico no limite em que se aumenta o tamanho do sistema.

4.3 TDA para os casos q = 3, 4 e 5

A fim de fazer a aplicação da Análise Topológica de Dados, utilizamos a mesma base de dados de configurações de Monte Carlo usada anteriormente. Inicialmente foram calculados os diagramas de persistência para todas as amostras calculadas por temperatura. A Figura 4.13 mostra um mapa de calor construído para a matriz de distâncias para o caso $q = 5$, $N = 80$, no qual é possível ver as relações de similaridade entre os diagramas de persistência comparados aos pares. Quanto mais escuros os valores no mapa, mais topologicamente diferentes são aquelas amostras. Assim, nota-se a partir da figura que as distâncias entre os diagramas de persistência calculados para $0,75 < T < 0,85$ ($0,86 < T < 0,95$) (parte superior esquerda e inferior direita) são menores, significando que entre eles há maior similaridade. Enquanto a comparação entre um diagrama de uma amostra calculada abaixo de $T = 0.85$ e outro relativo a uma amostra calculada acima desse valor possuem similaridade baixa ou nula. Como se espera, esse valor corresponde à temperatura crítica para o modelo de Potts de 5 estados.

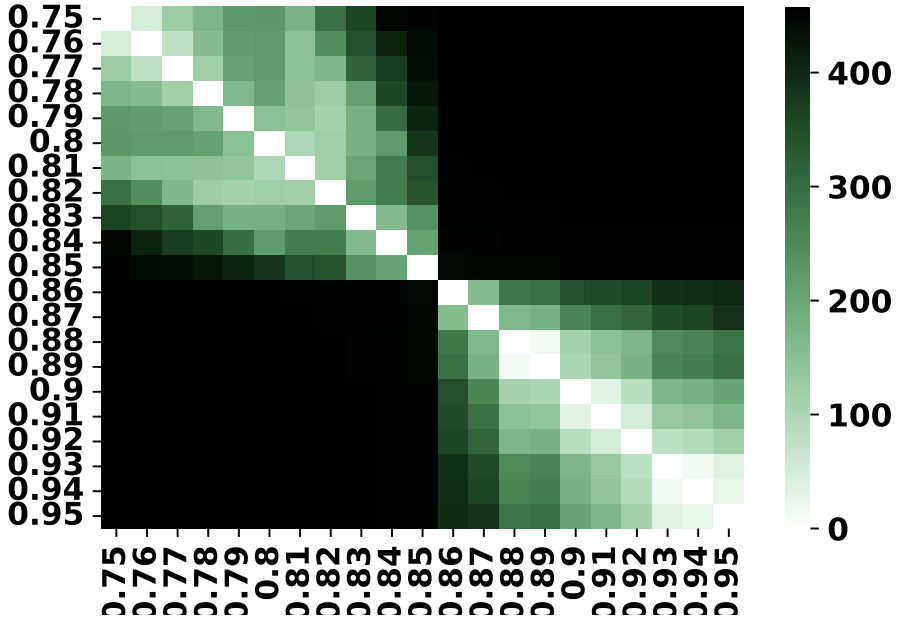

Figura 4.13: Mapa de calor calculado através de Homologia Persistente. Representa a matriz de distâncias para o modelo de Potts com $q = 5$ e $N = 80$. Os valores nos eixos indicam as diferentes temperaturas.

A partir das matrizes de distâncias, são construídas as matrizes de similaridade para o cálculo espectral, onde deve ser feita a segunda parte dessa análise, descrita no Capítulo 3. As funções de pertinência calculadas para todos os modelos (q = 3, 4 e 5), mostradas na Figura 4.14, mostram as relações de similaridade para determinados valores de T/J na faixa de temperaturas

de cada modelo. Abaixo de um determinado valor crítico, em todos os casos, os valores da função estão próximos de 1 (que é o máximo); e, acima desse valor crítico, estão próximos de 0. Como se espera, dados pertencentes à mesma fase devem compartilhar resultados semelhantes. Note que as funções ilustradas na Figura 4.14 são associadas ao grupo "ordenado". A função complementar (relativa ao grupo "desordenado") pode ser usada para analisar o ponto onde as curvas das funções de pertinência se cruzam, como mostra a Figura 4.15 para o caso $q = 5$, $N = 40$. Com análise semelhante para os demais sistemas, observamos que os valores críticos obtidos correspondem àqueles esperados para as temperaturas críticas em todos os valores de q com boa aproximação. A Figura 4.16 mostra que esses valores críticos exibem certa independência dos tamanhos das redes, de modo que a análise de um expoente crítico não pode ser feita para esse método a partir da Equação 4.3. Por outro lado, resultados consistentes para as temperaturas críticas são obtidos mesmo para redes pequenas.

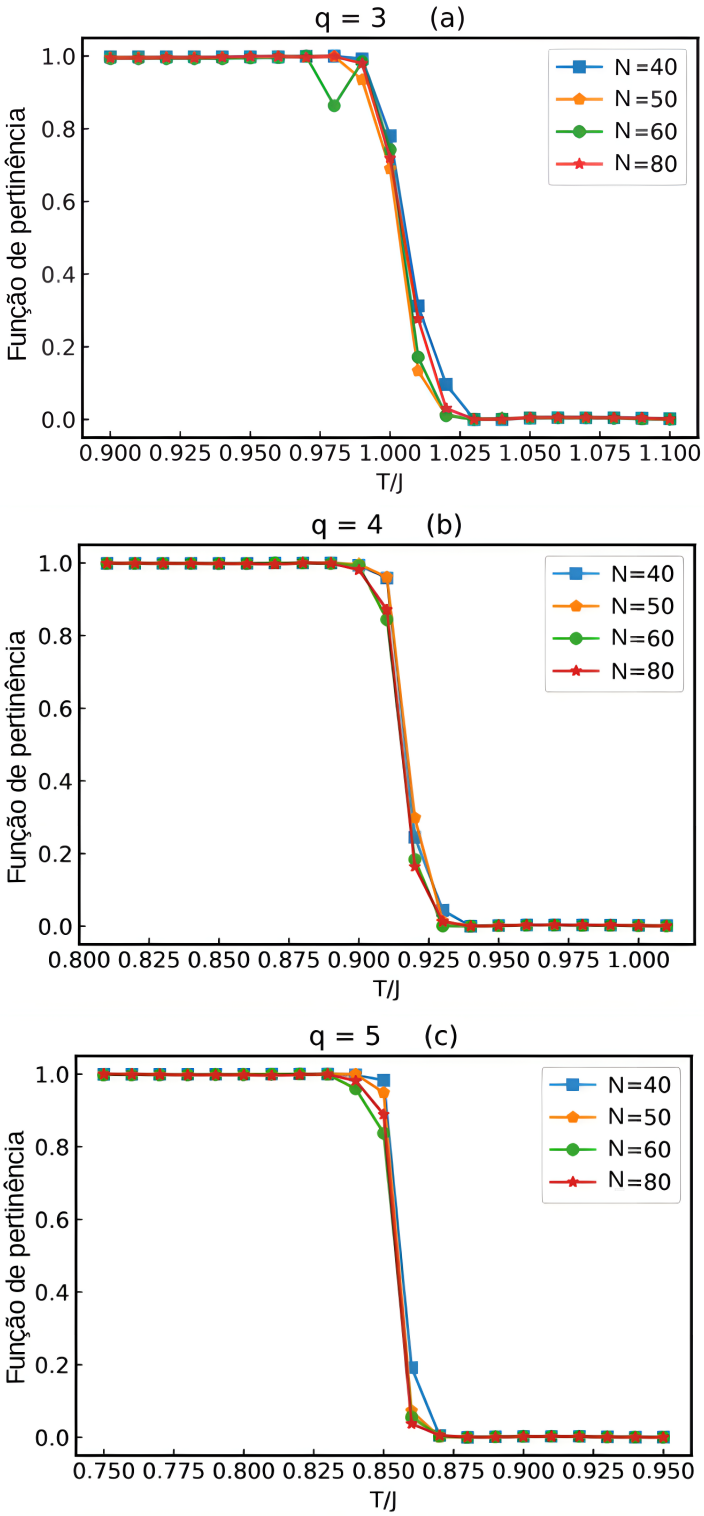

Figura 4.14: Funções de pertinência para os casos (a) q = 3, (b) q = 4 e (c) q = 5. Diferentes símbolos referem-se a diferentes tamanhos lineares de rede.

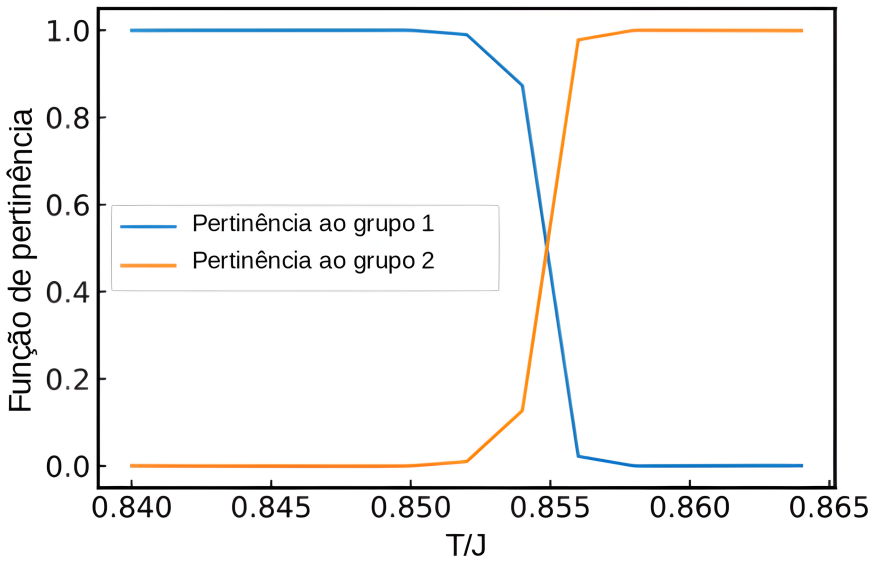

Figura 4.15: Forma das funções de pertinência para o caso $q = 5$, $N = 40$, relativas ao grupo "ordenado" (grupo 1) e "desordenado" (grupo 2) . O ponto de cruzamento das curvas indica a temperatura crítica para esse sistema.

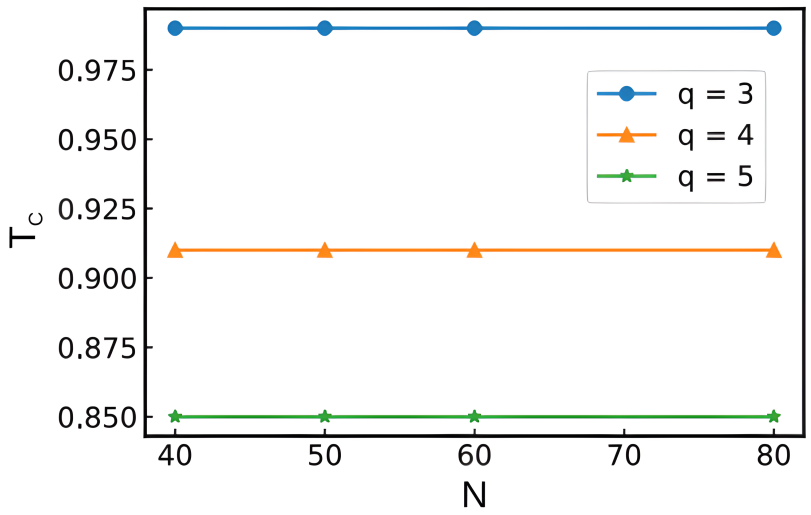

Figura 4.16: Temperaturas críticas obtidas através de TDA para o modelo de Potts de q-estados. Esse resultado mostra que, para essa técnica, os valores críticos são independentes do tamanho do sistema (N).

5 Conclusões

Neste trabalho, analisamos o modelo de Potts de q-estados para q = 3, 4 e 5, através de técnicas não supervisionadas de *machine learning*: Análise de Componentes Principais (PCA), Agrupamento *K-means* e Análise Topológica de Dados (TDA). Foram analisadas matrizes de configurações de Monte Carlo calculadas para o modelo de Potts, sob variação de temperatura, em diferentes tamanhos de rede. PCA e *K-Means* são técnicas para redução de dimensionalidade e clusterização, respectivamente, enquanto TDA permite que um sistema seja analisado de um ponto de vista topológico.

Através das componentes principais de PCA, estimamos a forma do parâmetro de ordem para o modelo de Potts, assim como a temperatura crítica de cada modelo analisado com boa precisão. O uso de *K-Means* em conjunto com PCA reforça esse resultado. As ordens das transições de fase foram corretamente identificadas através da definição de um expoente crítico construído a partir das componentes principais. Resultados semelhantes para as temperaturas críticas foram obtidos a partir de TDA. Essa técnica mostrou-se menos sensível aos tamanhos das redes, retornando valores consistentes com aqueles previstos para as temperaturas críticas mesmo em redes pequenas.

Esperamos que esse trabalho contribua para o estudo de fenômenos críticos, de um ponto de vista das técnicas de *machine learning*, e que, no futuro, essas análises possam ser estendidas a redes maiores e sistemas mais complexos, como aqueles que apresentam frustração, vidros de *spin* ou mesmo sistemas quânticos com a presença do problema de sinal.

Referências Bibliográficas

[1] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, *Machine Learning An Artificial Intelligence Approach*. Springer-Verlag Berlin Heidelberg, 1983.

[2] R. Ferizal, S. Wibirama, and N. A. Setiawan, "Gender recognition using pca and lda with improve preprocessing and classification technique," *7th International Annual Engineering Seminar (InAES)*, pp. 1–6, 2017.

[3] I. Frommholz, H. M. al Khateeb, M. Potthast, Z. Ghasem, M. Shukla, and E. Short, "On textual analysis and machine learning for cyberstalking detection.," *Datenbank Spektrum*, vol. 16, p. 127–135, 2016.

[4] K. E. Adikari, S. Shrestha, D. T. Ratnayake, A. Budhathoki, S. Mohanasundaram, and M. N. Dailey, "Evaluation of artificial intelligence models for flood and drought forecasting in arid and tropical regions," *Environmental Modelling Software*, vol. 144, p. 105136, 2021.

[5] S.-C. B. Lo, H.-P. Chan, J.-S. Lin, H. Li, M. T. Freedman, and S. K. Mun, "Artificial convolution neural network for medical image pattern recognition," *Neural Networks*, vol. 8, no. 7, pp. 1201–1214, 1995.

[6] S. A. Begum and O. M. Devi, "Fuzzy algorithms for pattern recognition in medical diagnosis," *Assam University Journal of Science and Technology*, vol. 7, no. 2, 2011.

[7] E. Cambria and B. White, "Jumping nlp curves: A review of natural language processing research [review article]," *IEEE Computational Intelligence Magazine*, vol. 9(2), 48-57, 2014.

[8] M. Hirz and B. Walzel, "Sensor and object recognition technologies for self-driving cars," *Computer-Aided Design and Applications*, vol. 15, pp. 1–8, 01 2018.

[9] A. C. Müller and S. Guido, *Introduction to Machine Learning with Python*. O'Reilly Media, 2017.

[10] A. Gron, *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Inc., 1st ed., 2017.

[11] H. S., *Neural Networks and Learning Machines.* Pearson Education, Inc. 3rd ed., 1983.

[12] F. Van Veen and S. Leijnen, "The neural network zoo.." `https://www.asimovinstitute.org/neural-network-zoo`, 2019.

[13] D. R. Tan, C. D. Li, W. P. Zhu, and F. J. Jiang, "A comprehensive neural networks study of the phase transitions of potts model," *New J. Phys.*, vol. 22, Jun 2020.

[14] H. Stanley, *Introduction to Phase Transitions and Critical Phenomena*. International series of monographs on physics, Oxford University Press, 1987.

[15] K. Binder, "Applications of monte carlo methods to statistical physics," *Reports on Progress in Physics*, vol. 60, pp. 487–559, may 1997.

[16] L. Wang, "Discovering phase transitions with unsupervised learning," *Physical Review B*, vol. 94, Nov 2016.

[17] W. Hu, R. R. P. Singh, and R. T. Scalettar, "Discovering phases, phase transitions, and crossovers through unsupervised machine learning: A critical examination," *Physical Review E*, vol. 95, Jun 2017.

[18] C. Wang and H. Zhai, "Machine learning of frustrated classical spin models. i. principal component analysis," *Phys. Rev. B*, vol. 96, p. 144432, Oct 2017.

[19] C. Wang and H. Zhai, "Machine learning of frustrated classical spin models (ii): Kernel principal component analysis," *Frontiers of Physics*, vol. 13, Jun 2018.

[20] S. J. Wetzel, "Unsupervised learning of phase transitions: From principal component analysis to variational autoencoders," *Physical Review E*, vol. 96, Aug 2017.

[21] T. Mendes-Santos, X. Turkeshi, M. Dalmonte, and A. Rodriguez, "Unsupervised learning universal critical behavior via the intrinsic dimension," *Physical Review X*, vol. 11, Feb 2021.

[22] J. Carrasquilla and R. G. Melko, "Machine learning phases of matter," *Nature Physics*, vol. 13, 2017.

[23] H. Saito, "Solving the bose–hubbard model with machine learning," *Journal of the Physical Society of Japan*, vol. 86, p. 093001, Sep 2017.

[24] K. Ch'ng, J. Carrasquilla, R. G. Melko, and E. Khatami, "Machine learning phases of strongly correlated fermions," *Phys. Rev. X*, vol. 7, p. 031038, Aug 2017.

[25] E. Khatami, "Principal component analysis of the magnetic transition in the three-dimensional fermi-hubbard model," *Journal of Physics: Conference Series*, vol. 1290, p. 012006, oct 2019.

[26] A. Tirelli and N. C. Costa, "Learning quantum phase transitions through topological data analysis," *Physical Review B*, vol. 104, Dec 2021.

[27] C. Giannetti, B. Lucini, and D. Vadacchino, "Machine learning as a universal tool for quantitative investigations of phase transitions," *Nuclear Physics B*, vol. 944, p. 114639, 2019.

[28] A. L. Ferguson, "Machine learning and data science in soft materials engineering," *Journal of Physics: Condensed Matter*, vol. 30, p. 043002, dec 2017.

[29] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, "Machine learning and the physical sciences," *Reviews of Modern Physics*, vol. 91, Dec 2019.

[30] V. Chertenkov and L. Shchur, "Universality classes and machine learning," *Journal of Physics: Conference Series*, vol. 1740, p. 012003, 01 2021.

[31] F. Y. Wu, "The potts model," *Rev. Mod. Phys.*, vol. 54, 1982.

[32] R. B. Potts, "Some generalized order-disorder transformations," *Proc. Cambridge Phil. Soc.*, vol. 48, pp. 106–109, 1952.

[33] A. W. Sandvik, A. Avella, and F. Mancini, "Computational studies of quantum spin systems," *AIP Conference Proceedings*, 2010.

[34] L. SUN, Y. F. CHANG, and X. CAI, "A discrete simulation of tumor growth concerning nutrient influence," *International Journal of Modern Physics B*, vol. 18, no. 17n19, pp. 2651–2657, 2004.

[35] S. Sanyal and J. A. Glazier, "Viscous instabilities in flowing foams: a cellular potts model approach.," *J Stat Mech.*, vol. 10, 2006.

[36] T. C. Schelling, "Dynamic models of segregation," *Journal of mathematical sociology*, vol. 1, no. 2, pp. 143–186, 1971.

[37] C. SCHULZE, "Potts-like model for ghetto formation in multi-cultural societies," *International Journal of Modern Physics C*, vol. 16, p. 351–355, 2005.

[38] I. T. Jolliffe, *Principal Component Analysis*. Springer Series in Statistics, 2002.

[39] J. Shlens, "A tutorial on principal component analysis," 2014.

[40] C. C. Aggarwal and C. K. Reddy, *Data Clustering: Algorithms and Applications*. CRC Press, 2014.

[41] Q. H. Tran, M. Chen, and Y. Hasegawa, "Topological persistence machine of phase transitions," *Phys. Rev. E*, vol. 103, p. 052127, May 2021.

[42] U. von Luxburg, "A tutorial on spectral clustering," *Max Planck Institute for Biological Cybernetics*, 2007.