Universidade Federal de São Carlos
Centro de Ciências Exatas e de Tecnologia
Departamento de Estatística

TÉCNICAS DE QUANTIFICAÇÃO DE INCERTEZA EM REDES NEURAIS PARA DENSIDADES CONDICIONAIS POR MEIO DE MODELOS DE MISTURAS

Vagner Silva Santos

Trabalho de Conclusão de Curso

Universidade Federal de São Carlos
Centro de Ciências Exatas e de Tecnologia
Departamento de Estatística

Técnicas de quantificação de incerteza em redes neurais para densidades condicionais por meio de modelos de misturas

Vagner Silva Santos
Orientador: Prof. Dr. Rafael Izbicki
Coorientador: Prof. Dr. Thiago Rodrigo Ramos

Trabalho de Conclusão de Curso apresentado como parte dos requisitos para obtenção do título de Bacharel em Estatística.

São Carlos
Julho de 2025

Federal University of São Carlos
Exact and Technology Sciences Center
Department of Statistics

Techniques for uncertainty quantification in neural networks for conditional densities through mixture models

Vagner Silva Santos
Advisor: Rafael Izbicki
Co-advisor: Thiago Rodrigo Ramos

Bachelors dissertation submitted to the Department of Statistics, Federal University of São Carlos - DEs-UFSCar, in partial fulfillment of the requirements for the degree of Bachelor in Statistics.

São Carlos
Mês em inglês 2025

Vagner Silva Santos

Técnicas de quantificação de incerteza em redes neurais para densidades condicionais por meio de modelos de misturas

Este exemplar corresponde à redação final do trabalho de conclusão de curso devidamente corrigido e defendido por nome do(a) aluno(a) e aprovado pela banca examinadora.

Aprovado em 27 de junho de 2025

Banca Examinadora:

- Prof. Dr. Rafael Izbicki
- Prof. Dr. Thiago Rodrigo Ramos
- Prof. Dr. Danilo Lourenço Lopes
- Prof. Dr. Ricardo Felipe Ferreira

Aos meus familiares e amigos por todo suporte

Agradecimentos

“Agradeço aos meus pais, Angelina de Jesus Silva Santos e Valmir de Souza Santos, além do meu irmão Vailson Silva Santos, que foram pessoas que me ajudaram antes e durante toda a graduação. Aos meus amigos que me apoiaram.”

“Às vezes, dar um salto à frente significa... deixar algumas coisas para trás.”

(Ekko e Powder, *Arcane*, Temporada 2, 2024)

Resumo

A incerteza em modelos de aprendizado estatístico divide-se em dois tipos principais: a incerteza aleatória, relacionada à variabilidade intrínseca dos dados, e a incerteza epistêmica, causada pelo conhecimento incompleto sobre o processo gerador, frequentemente associada à escassez de dados. A estimação da densidade condicional permite uma boa quantificação da incerteza aleatória nestes problemas. No entanto, a quantificação da incerteza epistêmica tradicionalmente requer métodos Bayesianos, os quais, embora teoricamente fundamentados, apresentam custo computacional elevado em alta dimensão. Além disso, estes não possuem garantias de cobertura. Neste trabalho, mostramos como redes neurais com *dropout* em modelos de mistura podem ser usadas para aproximar a inferência Bayesiana via *Monte Carlo dropout*, incorporando a incerteza epistêmica, de forma a reduzir significativamente o custo computacional em comparação a redes Bayesianas. Além disso, mostramos como elas podem ser integradas naturalmente com regiões predição de (*Conformal Prediction*) para produzir intervalos preditivos com garantias teóricas de cobertura.

Palavras-chave: *Aprendizado estatístico; redes neurais; quantificação de incerteza; predição conforme; modelos de mistura.*

Abstract

The uncertainty in statistical learning models is divided into two main types: aleatoric uncertainty, related to the intrinsic variability of the data, and epistemic uncertainty, caused by incomplete knowledge about the data-generating process, often associated with data scarcity. Conditional density estimation allows for good quantification of aleatoric uncertainty. However, quantifying epistemic uncertainty traditionally requires Bayesian methods, which, although theoretically well-founded, entail high computational costs in high dimensions. Our interest in this problem arose from the scarcity of works addressing epistemic uncertainty and the search for computationally efficient and scalable alternatives. Our contribution focuses on using neural networks with *dropout* in mixture models, demonstrating that this approach: approximates Bayesian inference via *Monte Carlo dropout*, incorporating epistemic uncertainty; significantly reduces computational costs compared to Bayesian neural networks; and naturally integrates with *Conformal Prediction* regions to produce predictive intervals with theoretical coverage guarantees.

Keywords: *Statistical learning; neural networks; uncertainty quantification; conformal prediction; mixture models.*

Lista de Figuras

Lista de Tabelas

Sumário

Capítulo 1

Introdução

Nos últimos anos, os modelos estatísticos têm se consolidado como ferramentas essenciais na predição de dados, especialmente em áreas onde a precisão das previsões é crítica. No entanto, um problema recorrente com esses modelos, particularmente no contexto de aprendizado estatístico, como *Boosting*, Florestas Aleatórias e Redes Neurais, é a ausência de informações confiáveis sobre a precisão dessas predições. Esse desafio torna-se ainda mais complexo em situações com insuficiência de amostras em determinadas regiões do espaço de covariáveis, o que resulta em uma falta de confiança nas previsões realizadas.

Essa carência de informação é frequentemente descrita como *incerteza epistêmica*, uma das duas principais formas de incerteza que afetam os modelos preditivos, juntamente com a *incerteza aleatória*. A incerteza epistêmica surge da falta de conhecimento sobre o processo gerador dos dados e é especialmente problemática em redes neurais, que, embora amplamente reconhecidas pelo seu elevado poder preditivo, demandam grandes volumes de dados para evitar lacunas de conhecimento durante o treinamento. Essas lacunas são mais comuns em regiões menos densamente amostradas, resultando em um aumento da incerteza epistêmica.

As abordagens para quantificação de ambos os tipos de incerteza geralmente envolvem a construção de regiões de predição para novas observações. Uma técnica que tem ganhado destaque é o uso das Regiões de Predição Conformes (*Conformal Prediction Regions*). Essa metodologia produz intervalos preditivos com cobertura marginal válida, mesmo sob suposições fracas sobre a distribuição dos dados (Shafer e Vovk, 2008; Angelopoulos e Bates, 2022). Sua principal vantagem é garantir uma cobertura formal em cenários com pouca informação sobre a distribuição das variáveis.

O uso de redes neurais com *dropout* (Srivastava *et al.*, 2014), aplicadas a densida-

des condicionais por meio de modelos de misturas (Bishop, 1994), também oferece uma oportunidade de melhorar a quantificação de incerteza. O *dropout*, além de atuar como uma técnica de regularização, também permite capturar a incerteza epistêmica, simulando diferentes configurações de redes durante a predição. De fato, redes com *dropout* se aproximam de uma rede neural Bayesiana, mas sem seu alto custo computacional.

Neste trabalho, visamos combinar estas redes neurais com predições conforme. Ao fazer isso, espera-se uma quantificação mais robusta da incerteza epistêmica, especialmente em cenários com dados multimodais ou assimétricos. Essa combinação possibilita um entendimento mais profundo de eventos raros e comportamentos irregulares dos dados, fornecendo intervalos preditivos confiáveis mesmo em situações onde há insuficiência de amostras ou alta variabilidade.

No campo da quantificação de incertezas, alguns trabalhos se destacam. Um dos pioneiros é o artigo *"Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning"* (Gal e Ghahramani, 2016), que introduz o uso de redes neurais Bayesianas com *dropout* como uma aproximação prática para inferência variacional em modelos de aprendizado profundo (*deep learning*). Essa abordagem oferece uma maneira eficiente de quantificar incertezas, permitindo estimar tanto a incerteza epistêmica quanto a aleatória de forma escalável em arquiteturas profundas.

Outro trabalho relevante é o artigo *"Concrete Dropout"* (Gal *et al.*, 2017), que também utiliza *dropout* como ferramenta para estimar a incerteza do modelo. Contudo, este estudo propõe uma nova variante do *dropout*, baseada em um relaxamento contínuo das máscaras discretas de *dropout*, conhecida como *Concrete Dropout*. Essa nova abordagem não só melhora o desempenho do modelo, como também gera incertezas mais calibradas, fundamentadas em avanços recentes do aprendizado profundo bayesiano.

Adicionalmente, o artigo *"Integrating Uncertainty Awareness into Conformalized Quantile Regression"* (Rossellini *et al.*, 2024) apresenta inovações ao incorporar a noção de *Uncertainty Awareness* na Regressão Quantílica Conformalizada (*Conformalized Quantile Regression*). A principal contribuição é a separação explícita entre incerteza epistêmica e incerteza aleatória, permitindo a geração de intervalos preditivos mais confiáveis, com cobertura quase constante em todo o espaço de covariáveis.

Esses três estudos ilustram avanços importantes na quantificação de incertezas, abordando tanto o aspecto prático quanto teórico da incerteza em aprendizado profundo e modelos estatísticos.

A demanda por intervalos preditivos com cobertura condicional precisa motiva o desenvolvimento de técnicas que integrem ambos os tipos de incerteza. Nosso trabalho avança o estado da arte ao propor o **EPICSCORE**, método que combina redes de densidade de mistura com predição conforme utilizando *Monte Carlo dropout* como aproximação Bayesiana eficiente para quantificar a incerteza epistêmica nos escores. Diferente das abordagens existentes, nossa formulação preserva as garantias conformais enquanto incorpora informação epistêmica via distribuição preditiva do escore, oferecendo uma solução computacionalmente eficiente que supera limitações de métodos puramente Bayesianos ou conformais. Os experimentos em dados simulados e 8 conjuntos reais demonstram que o **EPICSCORE** produz intervalos mais estreitos e com indicativos de que a cobertura condicional nominal está sendo mantida em comparação as alternativas como CQR e UACQR, com destaque para seu desempenho em regiões de alta incerteza epistêmica onde métodos tradicionais falham em fornecer cobertura condicional adequada.

Esta monografia está organizada da seguinte forma: no Capítulo 2, apresentamos os fundamentos teóricos das redes neurais artificiais, abordando sua arquitetura, treinamento e técnicas de regularização. O Capítulo 3 discute os conceitos de incerteza aleatória e epistêmica, além de métodos para sua quantificação. No Capítulo 4, introduzimos nossa proposta principal, o **EPICSCORE**, detalhando sua formulação teórica e implementação. O Capítulo 5 apresenta os experimentos realizados em dados simulados e reais, com a avaliação comparativa dos métodos. Por fim, o Capítulo 6 traz as conclusões do trabalho, sumarizando as contribuições e sugerindo direções futuras de pesquisa.

Capítulo 2

Regressão usando Redes Neurais Artificiais

Neste capítulo, apresentamos brevemente a motivação e introduzimos o desenvolvimento das redes neurais artificiais. Após essa introdução, nos concentramos em mostrar como as redes neurais são usadas em problemas de regressão e classificação, destacando sua capacidade de aprendizado com os dados. Em seguida, exibimos como é a arquitetura geral de uma rede, detalhando cada um dos seus componentes (camadas, neurônios, pesos, função de ativação).

Com a arquitetura definida, mostramos como os componentes atuam no aprendizado, permitindo a captura até mesmo de comportamentos não lineares. O seguinte processo explicado consiste na escolha da função de perda e sua utilização para ajustar os pesos. Em seguida, elencamos alguns algoritmos de otimização eficientes e populares no treinamento do modelo, como *back-propagation* e Adam. Outro tópico importante se trata de formas de regularização da rede, fornecendo à rede a habilidade de mitigar o problema de overfitting, melhorando a capacidade de generalização. Dentre as técnicas abordadas se encontram, *Dropout*, *Batch Normalization* e *Early Stopping*.

Por fim, discutimos sobre o uso de redes neurais na estimação de modelos densidade de misturas. Uma abordagem que combina a robustez da rede neural com a flexibilidade dos modelos baseados em misturas de densidades. Detalhamos como essa combinação é realizada e o procedimento de estimação dos parâmetros.

Este capítulo, portanto, fornece uma visão abrangente de redes neurais aplicadas à regressão, permitindo um melhor entendimento dos seus fundamentos e o uso delas em aplicações práticas.

2.1 Introdução às redes neurais artificiais

As redes neurais artificiais (ou simplesmente redes neurais, em inglês *Neural Networks*), são modelos computacionais inspirados no funcionamento do cérebro humano, desenvolvidas para reconhecimento de padrões e aprender a partir de dados. Elas são compostas por unidades chamadas neurônios, que se organizam em camadas: uma camada de entrada, um conjunto de camadas ocultas e uma camada de saída. A camada de entrada é responsável pela recepção das variáveis preditoras. As camadas ocultas pelo processamento e pela maior parte do aprendizado a partir dos dados e a camada de saída, que produz o resultado final do modelo, podendo ser uma predição ou classificação. Cada neurônio recebe entradas, essas informações são processadas por meio de uma função de ativação e transmitidas para os neurônios da próxima camada. Por fim, a conexão entre dois neurônios possui um peso associado, que representa a importância daquela conexão. O valor desses pesos são ajustados durante a etapa de treinamento da rede sob alguma métrica de desempenho.

As redes neurais surgiram na década de 1940, quando os primeiros modelos foram propostos, como abordado em Mcculloch e Pitts (1943). Entretanto, no decorrer dos anos, as redes neurais presenciaram diversos períodos de baixa, sendo ofuscadas por outras técnicas de inteligência artificial mais populares e viáveis computacionalmente. Em virtude do eventual aumento da capacidade de processamento dos computadores e da disponibilidade de grandes conjuntos de dados, as redes retomaram sua relevância no cenário científico.

2.2 Treinamento da rede neural

A primeira etapa é a construção da arquitetura da rede neural, definindo o número de camadas ocultas (*hidden layers*), suas funções de ativação, a quantidade de unidades (ou neurônios) em cada camada e os pesos associados a essas conexões. Cada peso é um parâmetro da rede que determina a força da influência de uma entrada sobre um neurônio em uma camada subsequente. Assim, o treinamento da rede neural consiste em ajustar os pesos sob uma função de perda arbitrária. A Figura 2.1 ilustra uma rede neural genérica, onde cada conexão entre os neurônios tem um peso que é ajustado durante o processo de aprendizagem.

No contexto do aprendizado profundo, uma rede possui duas ou mais camadas ocultas.

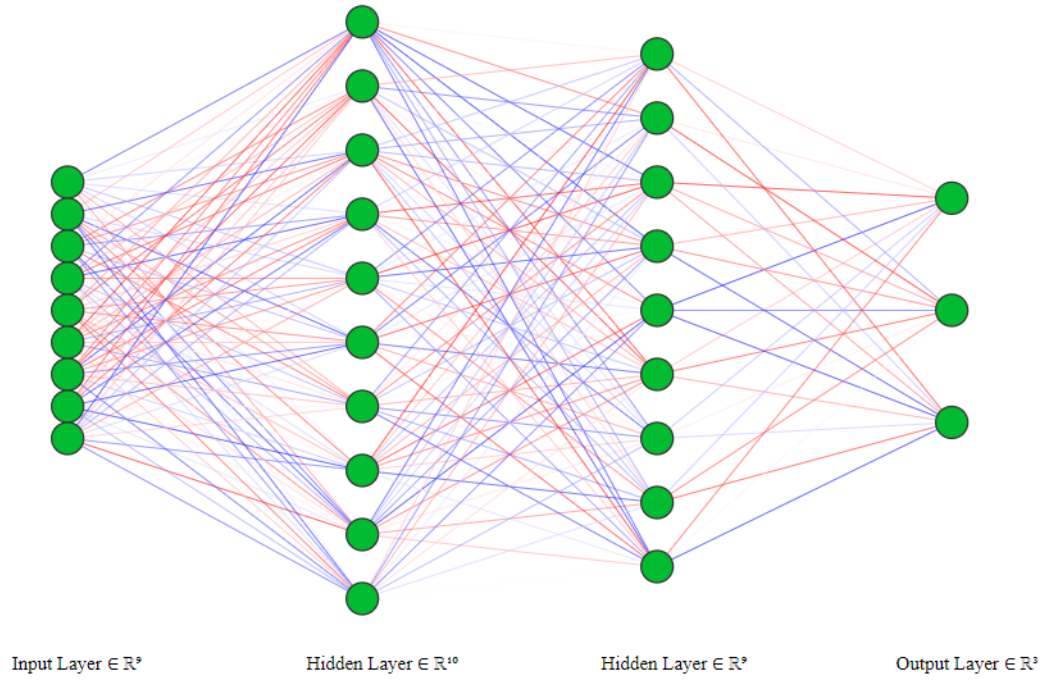

Figura 2.1: Ilustração de uma rede neural artificial genérica. Os círculos verdes representam os neurônios, organizados em camadas: a camada de entrada (*Input Layer*), camadas ocultas (*Hidden Layers*) e a camada de saída (*Output Layer*). As conexões entre os neurônios são indicadas por arestas, onde o peso de cada conexão é simbolizado pelas cores: azul para pesos negativos e vermelho para pesos positivos. Este diagrama foi gerado utilizando a ferramenta online disponível em http://alexlenail.me/NN-SVG/.

Considerando $x_1^l, x_2^l, \ldots, x_{m_l}^l$ como os $m_l \in \mathbb{N}$ neurônios da camada oculta l, podemos expressar o j-ésimo neurônio da camada oculta $l+1$ da forma

$$x_j^{l+1} = a\left(\beta_{0,j}^l + \sum_{i=1}^{m_l} w_{i,j}^l x_i^l\right),$$

em que $\beta_{0,j}^l$ é o viés do neurônio j na camada $l+1$, $w_{i,j}^l$ representa o peso da conexão do neurônio i da camada l para o neurônio j da camada $l+1$, e $a(\cdot)$ é uma função de ativação não linear aplicada à soma ponderada (veja alguns exemplos na Figura 2.2).

2.2.1 Erro e funções de perdas

O ajuste de uma rede neural requer uma função objetivo a ser minimizada. No geral, realiza-se a escolha de uma função de perda $\mathcal{L}$ que mensura a discrepância entre o valor

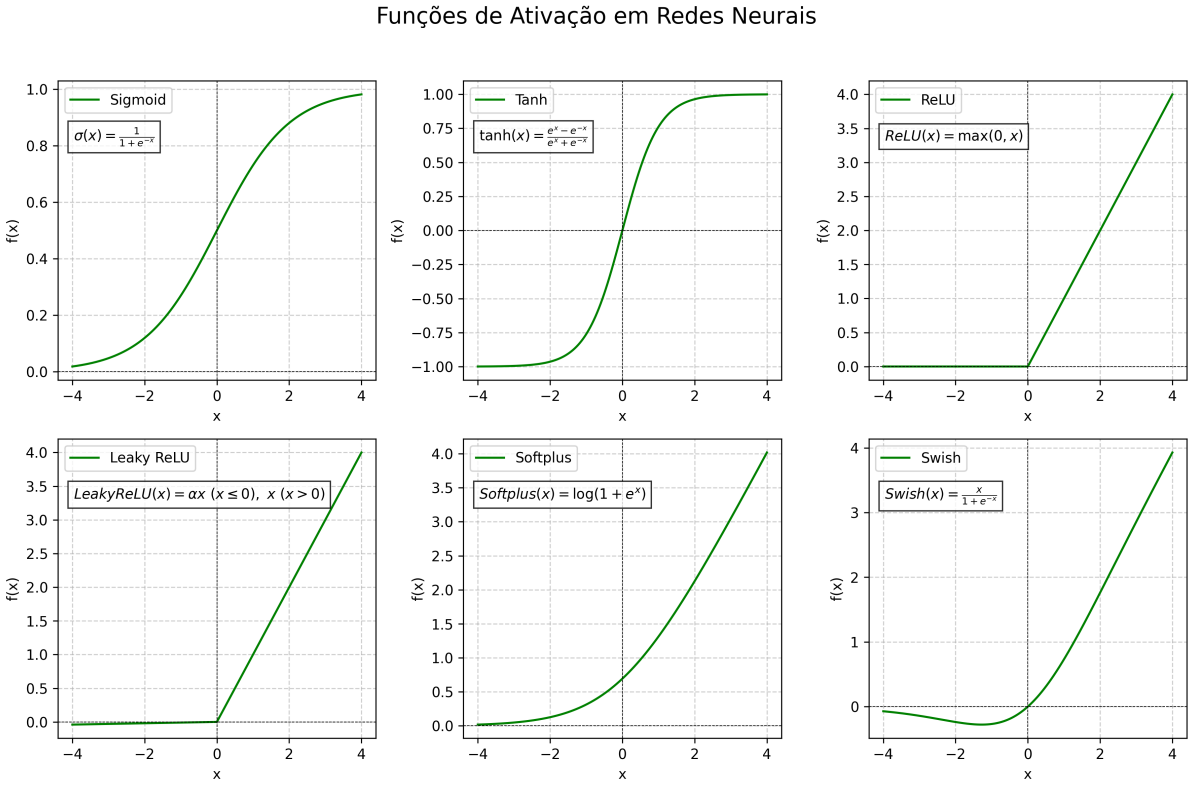

Figura 2.2: Representação gráfica de algumas das principais funções de ativação utilizadas em redes neurais artificiais: *Sigmoid*, *Tanh*, *ReLU*, *Leaky ReLU*, *Softplus* e *Swish*. Cada gráfico mostra a equação matemática correspondente e sua respectiva curva, ilustrando o comportamento da função em relação à entrada x. Essas funções são essenciais para incorporar não linearidades nos modelos, proporcionando às redes a capacidade de aprender representações mais complexas dos dados.

observado y e saída final do modelo $g_{\boldsymbol{w}}(x)$. A função de perda pode ser definida como:

$$\mathcal{L}(g_{\boldsymbol{w}}(x), y) = \frac{1}{n}\sum_{i=1}^{n} \ell(g_{\boldsymbol{w}}(x_i), y_i),$$

onde $\ell(g_{\boldsymbol{w}}(x_i), y_i)$ é a perda individual para a i-ésima observação.

Em tarefas de regressão, dentre as principais funções de perda individual temos o Erro Quadrático Médio (EQM) e o Erro Absoluto Médio (EAM). Seja $\boldsymbol{w}$ o vetor de pesos da rede neural, então:

$$EQM: \quad \ell(g_{\boldsymbol{w}}(x_i), y_i) = (g_{\boldsymbol{w}}(x_i) - y_i)^2, \tag{2.1}$$

$$EAM: \quad \ell(g_{\boldsymbol{w}}(x_i), y_i) = |g_{\boldsymbol{w}}(x_i) - y_i|, \tag{2.2}$$

em que os valores $g_{\boldsymbol{w}}(x_i)$ dependem do vetor de pesos $\boldsymbol{w}$ da rede neural.

Nos problemas de classificação, a métrica mais usual é a entropia cruzada (*cross-entropy - CE*). Seja $\mathbb{1}_{\{y_i=j\}}$ a função indicadora que vale 1 se $y_i = j$ e 0 caso contrário. A

entropia cruzada é definida como:

$$CE(g_{\boldsymbol{w};0}(x_i), \ldots, g_{\boldsymbol{w};K}(x_i)) = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{K} \mathbb{1}_{\{y_i=j\}} \log(g_{\boldsymbol{w};j}(x_i)), \tag{2.3}$$

sendo K o número total de classes e $g_{\boldsymbol{w};j}(x_i)$ a probabilidade predita pelo modelo para a classe j da observação i.

2.2.2 Algoritmos de otimização

Os pesos são ajustados durante o treinamento da rede por meio de um processo de otimização. Rumelhart *et al.* (1986) descrevem um algoritmo bastante usual para obtenção dos pesos, chamado de retropropagação (*back-propagation*). Esse método calcula o gradiente da perda com relação a cada peso da rede, com o objetivo de minimizar a função de perda para que a rede aprenda a mapear corretamente as entradas para as saídas. A equação básica de atualização dos pesos em uma camada l é dada pela forma:

$$\Delta w_{i,j}^{l} = -\eta \frac{\partial \mathcal{L}}{\partial w_{i,j}^{l}},$$

em que $w_{i,j}^{l}$ representa o peso do j-ésimo neurônio na $(l-1)$-ésima camada para o i-ésimo neurônio na l-ésima camada. Além disso, $\Delta w_{i,j}^{l}$ é a variação do peso $w_{i,j}^{l}$, η é a taxa de aprendizado, e $\frac{\partial \mathcal{L}}{\partial w_{i,j}^{l}}$ é o gradiente da função de perda $\mathcal{L}$ em relação ao peso $w_{i,j}^{l}$. Esse procedimento ajusta os pesos iterativamente, propagando o erro da camada de saída (*output layer*) para as camadas anteriores, permitindo que a rede aprenda padrões a partir dos dados.

Algoritmos de otimização baseados em gradientes estocásticos desempenham um papel importante em muitos campos das ciência e engenharias. Nessas áreas, podemos considerar os diversos problemas como a otimização de alguma função objetivo escalar parametrizada que requer maximização ou minimização em relação aos seus parâmetros. Nesse contexto, Kingma e Ba (2017) propõe o *Adam*, um método eficiente para otimização estocástica que utiliza somente gradientes de primeira ordem com baixa necessidade de memória. A ideia do método é calcular taxas individuais de aprendizagem adaptativa para diferentes parâmetros a partir de estimativas do primeiro e segundo momentos dos gradientes. O nome Adam é derivado de *adaptive moment estimation* (em português, estimativa de momento adaptativo).

2.3 Técnicas de regularização: *Dropout*, *Batch Normalization*, *Early Stopping*

O problema de *overfitting* é identificado durante o treinamento de redes neurais, especialmente em arquiteturas mais complexas, devido ao uso excessivo das funções ajustadas aos dados. Srivastava *et al.* (2014) propõem o método *dropout*, que consiste em, durante o treinamento, remover temporariamente e de forma aleatória algumas unidades da rede, tanto da camada oculta quanto da camada de entrada. Essas unidades são excluídas com uma probabilidade fixa p, e suas conexões de entrada e saída são removidas.

Em redes neurais profundas, uma técnica popular e eficaz que acelera consistentemente a sua convergência, é a chamada *Batch Normalization* (normalização em lote), veja os detalhes em Ioffe e Szegedy (2015). Nesse mesmo artigo, eles definem o *deslocamento covariante interno* (termo original em inglês, *Internal Covariate Shift*) como a mudança na distribuição das ativações da rede causada por alteração nos parâmetros da rede durante o treinamento. A *Batch Normalization* surge justamente para reduzir esse efeito, consistindo na divisão do conjunto de dados em lotes chamados *minibatches* (mini-lotes, em português). Para assim, em cada iteração do treinamento, primeiro normaliza-se (subtrair pela média e dividir pelo desvio padrão) as entradas utilizando as estatística do *minibatch* atual e em seguida aplica-se um coeficiente de escala e um deslocamento de escala.

Formalmente, se denotamos $x \in \mathcal{B}$ uma entrada para normalização em lote (BN) que é um *minibatch* $\mathcal{B}$, teremos a seguinte expressão para a normalização em lote que transforma x:

$$BN(x) = \gamma \left(\frac{x - \hat{\mu}_{\mathcal{B}}}{\hat{\sigma}_{\mathcal{B}}} \right) + \beta, \tag{2.4}$$

em que $\hat{\mu}_{\mathcal{B}}$ é média amostral e $\hat{\sigma}_{\mathcal{B}}$ é o desvio-padrão da amostra do *minibatch* $\mathcal{B}$. Note que γ e β deverão ser aprendidos juntos com os demais parâmetros da rede.

Além das técnicas citadas anteriormente, temos o *Early Stopping* que é amplamente utilizada para prevenção de *overfitting* em redes neurais. Segundo Prechelt (1996), essa abordagem consiste em interromper o treinamento assim que o erro na validação começar a aumentar significativamente, indicando um possível superajuste nos dados. Além da sua simplicidade na implementação, se mostrou mais eficiente na regularização que outras técnicas em algumas situações. Entretanto, a seleção do critério de parada ideal ocasi-

ona em um *trade-off* entre tempo de treinamento e a performance o modelo, apesar da capacidade de melhorar para critérios de parada mais lentos e gerais, acompanharia um tempo de treinamento maior.

2.4 Redes de Densidade de Mistura: Estimação de densidades condicionais

A importância de estimar a densidade condicional reside em sua capacidade de combinar flexibilidade com eficiência na geração de estimativas pontuais. A literatura recebeu boas formas para estimar de densidades condicionais. Um dos métodos que se destacaram são as Redes de Densidade de Mistura (do inglês *Mixture Density Networks - MDNs*) (Bishop, 1994). Em seu trabalho, ele apresenta uma abordagem combinando Redes Neurais com modelos de misturas para representação de distribuições condicionais complexas. A ideia central é estimar os parâmetros de múltiplas distribuições Gaussianas (ou normais) através de uma rede neural, permitindo que a MDN capture as incertezas e multimodalidade nos dados de saída. Esse método modela a distribuição condicional $\mathbf{Y}|\mathbf{x}$ como

$$f(\mathbf{y}|\mathbf{x}) = \sum_{k=1}^{m} \alpha_k(\mathbf{x})\phi_k(\mathbf{y}|\mathbf{x}), \tag{2.5}$$

em que m é o número de componentes na mistura, os parâmetros $\alpha_k(\mathbf{x})$ são chamados de *coeficientes de mistura*, as funções $\phi_k(\mathbf{y}|\mathbf{x})$ representam as densidades condicionais da vetor resposta $\mathbf{y}$, com $\mathbf{y} \in \mathbb{R}^c$, no k-ésimo componente, c é a dimensão da resposta $\mathbf{y}$. Neste trabalho, convencionamos em usar o componente Gaussiano da forma

$$\phi_k(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi)^{c/2}\sigma_k(\mathbf{x})^c} \exp\left\{-\frac{\|\mathbf{y} - \boldsymbol{\mu}_k(\mathbf{x})\|^2}{2\sigma_k(\mathbf{x})^2}\right\} \tag{2.6}$$

que, devido à sua forma geral, é capaz de modelar qualquer variável resposta $\mathbf{y} \in \mathbb{R}^c$, independentemente do domínio. Essa flexibilidade é fundamental para a metodologia proposta, pois permite capturar relações complexas nos dados sem fixar restrições.

No modelo da equação (2.6), o vetor $\boldsymbol{\mu}_k(\mathbf{x})$ representa o centro do k-ésimo kernel, com entradas μ_{kr}. Em (2.6) realizamos as suposições de que as entradas do vetor resposta em cada componente da distribuição são independentes e possuem a mesma variância $\sigma_k(\mathbf{x})^2$.

Definimos $\boldsymbol{z}$ como sendo as variáveis de saída da rede neural e elas serão usadas para

estimar os parâmetros. Primeiro particionamos elas em $\boldsymbol{z}^{\alpha}$, $\boldsymbol{z}^{\mu}$ e $\boldsymbol{z}^{\sigma}$. Os coeficientes de mistura são obtidos usando a função de ativação softmax na variáveis de saída $\boldsymbol{z}^{\alpha}$ pois ela garante que $\sum_{k=1}^{m} \alpha_k(\mathbf{x}) = 1$,

$$\alpha_k(\mathbf{x}) = \frac{\exp\left(z_k^{\alpha}\right)}{\sum_{j=1}^{M} \exp\left(z_j^{\alpha}\right)}.$$

As variâncias são estimadas por meio de $\boldsymbol{z}^{\sigma}$, sendo um parâmetro de escala, aplicamos a função exponencial garantindo que tenhamos um valor positivo e diferente de 0,

$$\sigma_k(\mathbf{x}) = \exp z_k^{\sigma}.$$

Por fim, os componentes dos centros são obtidos com $\boldsymbol{z}^{\mu}$, simplesmente

$$\mu_{kr} = z_{kr}^{\mu},$$

em que k indexa o componente da mistura e r a dimensão de $\mathbf{y}$.

O treinamento da rede para obtenção dos componentes da mistura exige uma função objetivo a ser otimizada. Usualmente a escolha consiste no negativo da função logaritmo da verossimilhança. Assim, utilizamos a seguinte função de erro

$$E = \sum_{i=1}^{n} E_i, \tag{2.7}$$

em que E_i é o erro contribuído pela i-ésima observação, sendo da forma

$$E_i = -\ln \sum_{k=1}^{m} \alpha_k(\mathbf{x}_i)\phi_k(\mathbf{y}_i|\mathbf{x}_i), \tag{2.8}$$

o termo $\phi_k(\mathbf{y}|\mathbf{x})$ é dado pela expressão em (2.6).

Capítulo 3

Quantificação de incerteza

Neste capítulo, introduzimos o conceito de incerteza e apresentamos algumas definições conhecidas pela área da estatística. Além disso, comentamos sobre uma divisão dos tipos de incertezas e a importância de quantificá-la adequadamente para a realização de boas predições.

Essa busca por predições mais confiáveis acaba resultando na exploração de formas de mensurar a incerteza. Uma abordagem relevante é o uso da inferência Bayesiana, aproveitando sua possibilidade de capturar explicitamente os tipos de incerteza. No entanto, apesar dessa vantagem, a inferência Bayesiana é uma alternativa bem computacionalmente custosa.

Uma abordagem que vem ganhando destaque na quantificação de incerteza são as regiões de predição conforme (Shafer e Vovk, 2008; Angelopoulos e Bates, 2022). Essa metologia é mais eficiente para quantificar a incerteza aleatória, mas possui limitações ao lidar com a incerteza epistêmica. Como resposta a essas limitações, novas alternativas têm sido desenvolvidas para aprimorar a quantificação da incerteza. Entre elas, destacamos uma técnica baseada na incorporação da incerteza consciente em regressões quantílicas conformalizadas (Rossellini *et al.*, 2024).

Dada a complexidade computacional enfrentada pela abordagem Bayesiana, apresentamos uma teoria que utiliza o *dropout* na rede neural como uma aproximação Bayesiana. Assim, introduzimos um novo método de quantificação de incerteza que utiliza essa aproximação Bayesiana como uma forma de incorporar a incerteza epistêmica na abordagem de região de predição conforme.

Este capítulo, portanto, explora o conceito de incerteza e sua importância e, em seguida, detalha formas de quantificá-la com o objetivo de melhorar a qualidade das

predições.

3.1 O que é incerteza?

A incerteza é uma das noções mais importantes na construção da metodologia dos modelos de aprendizado estatístico. Nesse contexto, ela é associada com a predição da variável resposta Y com as covariáveis $\mathbf{x}$ sendo dividida em duas categorias. Segundo Kiureghian e Ditlevsen (2009)

- **Incerteza Aleatória** (ou *aleatoric uncertainty*) se refere a aleatoriedade intrínseca de um fenômeno. Em outras palavras, o mesmo conjunto de covariáveis x estão associados a diferentes valores de y.

- **Incerteza Epistêmica** (*epistemic uncertainty*) a palavra epistêmica de origem grego επιστημη (*episteme*), que significa conhecimento. Assim, uma incerteza epistêmica diz respeito àquela resultante da falta de conhecimento (ou dados). Nas tarefas de predição, essa incerteza provém do fato que desconhecemos a distribuição real $Y|\mathbf{x}$.

Uma das formas de modelar as duas formas de incerteza é utilizando a inferência Bayesiana. Nesse contexto, precisamos de uma notação adicional. Assim como no cenário frequentista, o modelo Bayesiano requer uma família de distribuições que modela a incerteza aleatória de $Y|\mathbf{x}$, com $Y \in \mathbb{R}$ e $\mathbf{X} \in \mathbb{R}^d$. Essa classe é denotada por

$$\mathcal{F} = \{f(y|\mathbf{x}, \theta) : \theta \in \Theta\},$$

em que Θ representa um espaço paramétrico ou não paramétrico (ou seja, um espaço que não pode ser mapeado para $\mathbf{R}^d$). Além disso, uma característica fundamental do modelo Bayesiano é a incorporação de uma informação prévia sobre θ, chamada de *distribuição a priori*, que, por simplicidade, assumimos ter densidade $f(\theta)$ em relação a alguma medida dominante. Essa distribuição a *priori* captura a incerteza epistêmica sobre o processo gerador dos dados.

Seja $D = \{(\mathbf{X}_1, Y_1), (\mathbf{X}_2, Y_2), \ldots, (\mathbf{X}_n, Y_n)\}$ o conjunto de treinamento. Na inferência Bayesiana, assumimos que dado θ, os pontos dos dados são independentes e possuem a mesma distribuição condicional $f(y|\mathbf{x}, \theta)$. Assim a predição nesse cenário é feita calcu-

lando a densidade preditiva

$$f(y|\mathbf{x}, D) = \int f(y|\mathbf{x}, \theta) f(\theta|D) d\theta,$$

em que, sob a suposição de que um novo ponto $(\mathbf{x}, y)$ é independente de D (dado θ), $f(\theta|\mathbf{x}, D)$ é a denominada *distribuição a posteriori* e $f(y|\mathbf{x}, \theta)$ um modelo para a incerteza aleatória. Uma das formas de expressar a incerteza matematicamente, em termos de variância, é dada por:

$$\mathbb{V}[Y|\mathbf{x}, D] = \mathbb{E}_{\theta \sim f(\theta|D)}[\mathbb{V}[Y|\mathbf{x}, \theta]] + \mathbb{V}_{\theta \sim f(\theta|D)}[E[Y|\mathbf{x}, \theta]],$$

o primeiro termo da soma captura a incerteza aleatória, enquanto a segunda corresponde à incerteza epistêmica. A incerteza aleatória pode ser quantificada por meio da estimação de densidades condicionais $f(y|\mathbf{x})$. Alguns dos métodos populares para estimar $f(y|\mathbf{x})$ incluem aproximações paramétricas (como assumir que $Y|\mathbf{x}$ segue uma distribuição gaussiana), o *FlexCode* (Izbicki e Lee, 2017) e Redes de Densidade de Mistura (Bishop, 1994).

Entre os principais métodos para quantificar as incertezas aleatória e epistêmica, destacam-se as regiões de **predição conforme**. No entanto, estudos como os de Calonico *et al.* (2018) e Cheng e Chen (2019) mostram que o uso de $\hat{f}(y|\mathbf{x})$ para medir a incerteza epistêmica nem sempre é direto ou eficaz.

Diante disso, Nemani *et al.* (2023) propõem uma abordagem alternativa, explorando a eficácia de **modelos Bayesianos** na captura da incerteza epistêmica. Essa abordagem se mostra promissora devido a possibilidade de modelar simultaneamente a incerteza aleatória e a epistêmica relacionadas à variável Y, melhorando a precisão e a confiabilidade das predições.

3.2 Redes Neurais Bayesianas

As Redes Neurais Bayesianas (BNNs) são um tipo de rede neural que integra princípios Bayesianos na modelagem, permitindo lidar explicitamente com a incerteza nos parâmetros do modelo. A principal ideia é tratar os pesos da rede como variáveis aleatórias, em vez de valores fixos, o que permite representar naturalmente a incerteza epistêmica. Isso difere das redes neurais tradicionais, em que os pesos são ajustados durante o treinamento e permanecem fixos na fase de predição.

Nas BNNs, assume-se uma distribuição *a priori* sobre os pesos, sendo uma escolha comum a distribuição gaussiana com média 0, dada por:

$$p(\mathbf{w}) = \frac{1}{Z_W(\alpha)} \exp\left(-\frac{\alpha}{2}\|\mathbf{w}\|^2\right), \tag{3.1}$$

em que o termo $Z_W(\alpha)$ é chamado de *fator de normalização*, com

$$Z_W(\alpha) = \left(\frac{2\pi}{\alpha}\right)^{W/2}, \tag{3.2}$$

sendo W representando o número total de pesos da rede neural.

O objetivo das BNNs é calcular a distribuição *a posteriori* dos pesos, $p(\mathbf{w} \mid D)$, dadas as observações D. No entanto, encontrar a distribuição *a posteriori* exata é computacionalmente inviável para redes neurais complexas. Por isso, são empregados métodos de aproximação, tais como:

- **Inferência Variacional**: aproxima a distribuição *a posteriori* por uma distribuição mais simples, minimizando a divergência entre a aproximação e a distribuição real.
- **Monte Carlo *Dropout***: aplica *dropout* durante a fase de predição para simular diferentes realizações dos pesos, permitindo a estimação da incerteza epistêmica.

As Redes Neurais Bayesianas capturam a incerteza epistêmica (incerteza esta, associada aos parâmetros do modelo), o que é crucial em cenários com dados insuficientes ou ruidosos. Essa característica se alinha ao objetivo deste trabalho de quantificar a incerteza epistêmica em redes neurais. Além disso, no contexto de modelagem de densidades condicionais utilizando modelos de misturas, as BNNs podem ser componentes chave para melhorar a robustez e fornecer intervalos preditivos mais confiáveis, especialmente em situações com multimodalidade ou assimetria nos dados.

3.3 Regiões de Predição Conforme

A necessidade de quantificar a incerteza motivou o desenvolvimento de regiões de predição conforme, cujo objetivo principal é utilizar os dados observados para a obtenção de regiões de predições válidas (ou seja, regiões de predições para Y_{n+1} com uma cobertura garantida) sob suposições fracas. Um aspecto interessante é que, em geral, a suposição de

independência e identicamente distribuído (i.i.d.) já é suficiente para garantir a validade das regiões de predição. No entanto, essa suposição pode ser relaxada e substituída por uma condição mais fraca, conhecida como permutabilidade, sem comprometer a validade dos resultados (Angelopoulos e Bates, 2023; Shafer e Vovk, 2008; Vovk *et al.*, 2005; Izbicki, 2025).

Um método comum para construção dessas regiões conforme consiste em dividir o conjunto de dados em dois subconjuntos: treinamento D_1 e calibração D_2. Em seguida, um escore de não conformidade $s : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ é treinado usando D_1. A função $s(\mathbf{x}, y)$ serve para quantificar a plausibilidade de um rótulo y dado uma instância $\mathbf{x}$. Um valor alto de $s(\mathbf{x}, y)$ sugere que a predição de y para uma instância com covariáveis $\mathbf{x}$ é improvável. No contexto de regressão, alguns exemplos de *scores* de não conformidade são:

- **[Regresion-split]** (Divisão de regressão): $s(\mathbf{x}, y) = |y - \hat{g}(\mathbf{x})|$, em que $\hat{r}$ é uma estimativa pontual de Y (Lei *et al.*, 2018). Ele quantifica a distância de y a previsão pontual $\hat{g}(\mathbf{x})$. Esse escore é extremamente útil para avaliação de modelos de regressão, pois mede diretamente a discrepância entre os valores observados e preditos.

- **[CQR]** (Regressão Quantílica Conformalizada): $s(\mathbf{x}, y) = \max\{\hat{q}_{\alpha_1}(\mathbf{x}) - y, y - \hat{q}_{\alpha_2}(\mathbf{x})\}$, sendo $\hat{q}_{\alpha_1}$ e $\hat{q}_{\alpha_2}$ quantis estimados (Romano *et al.*, 2019) com $\alpha_1, \alpha_2 \in (0, 1)$ para $\alpha_1 < \alpha_2$. Esse mensura o quanto uma observação y é distante dos quantis estimados da distribuição condicional de Y dado $\mathbf{x}$.

- **[Weighted]** (Regressão ponderada): $s(\mathbf{x}, y) = \frac{|y - \hat{g}(\mathbf{x})|}{\hat{\rho}(\mathbf{x})}$, onde $\hat{\rho}(\mathbf{x})$ é qualquer estimador do desvio absoluto médio condicional $|Y - \hat{g}(\mathbf{X})||\mathbf{X} = \mathbf{x}$ (Lei *et al.*, 2018). Esse escore é uma medida de erro relativo, ponderada pela variabilidade esperada dos dados. Em que um valor alto de $s(\mathbf{x}, y)$ indica que y está distante da predição $\hat{g}(\mathbf{x})$, considerando a variabilidade natural dos dados.

- **[HPD-split]** (Divisão de Maior Densidade Preditiva, do inglês *Highest Predictive Density-split*): $s(\mathbf{x}, y) = -\int_{y' : \hat{f}(y'|\mathbf{x}) \leq \hat{f}(y|\mathbf{x})} \hat{f}(y'|\mathbf{x}) dy'$ (Izbicki *et al.*, 2021). Um valor alto de $s(\mathbf{x}, y)$ indica que y está em uma região de baixa densidade da distribuição condicional, ou seja, é uma observação atípica.

Com um *score* $s(\mathbf{x}, y)$, a região de predição conforme para uma nova predição de Y_{n+1}

é dada pela seguinte expressão

$$R(\mathbf{x}_{n+1}) = \{y : s(\mathbf{x}_{n+1}, y) \leq t\}, \tag{3.3}$$

o corte $t \in \mathbb{R}$ é estimado via calibração com D_2 pela técnica de *split conformal prediction (split-CP)*. Essa consiste em calcular $U_i = s(\mathbf{X}_i, Y_i)$ para cada amostra $(\mathbf{X}_i, Y_i) \in D_2$. O corte é dado por

$$t_{(1-\alpha)} \coloneqq U_{[(\frac{n+1}{n})(1-\alpha)]},$$

em que a normalização pelo fator $\frac{n+1}{n}$ é aplicada conforme (Lei *et al.*, 2018, Capítulo 2) para garantir a propriedade de cobertura finita. Nesse contexto, $U_{(1)}, \ldots, U_{(n)}$ representam as estatísticas de ordem de U. O Teorema 3.4 garante que as regiões conforme possuem cobertura marginal válida.

Teorema 3.4 Lei et al. *(2018) Se os dados são i.i.d, a região de predição*

$$R(\mathbf{X}_{n+1}) = \{y : s(\mathbf{x}_{n+1}, y) \leq U_{[(\frac{n+1}{n})(1-\alpha)]}\},$$

é marginalmente uma região de predição válida com

$$\mathbb{P}(Y_{n+1} \in R(\mathbf{X}_{n+1})) \geq 1 - \alpha.$$

O problema dos *scores* mencionados anteriormente é que, embora sejam eficazes na captura da incerteza aleatória, mostram-se insuficientes para quantificar adequadamente a incerteza epistêmica. Neste trabalho, utilizamos esses *scores* de não conformidade como ponto de partida, propondo adaptações que os tornem mais apropriados para a quantificação da incerteza epistêmica. A adaptação será realizada por meio da estimação de quantidades que capturam melhor esse tipo de incerteza, empregando modelos de Redes de Densidade de Mistura. Embora existam abordagens baseadas em Redes Neurais Bayesianas que exploram a incerteza epistêmica, elas tendem a ser computacionalmente mais intensivas. Assim, optamos por uma alternativa que ofereça um menor custo computacional, mantendo, no entanto, um nível elevado de robustez na modelagem das incertezas.

3.4 Integração da incerteza epistêmica na regressão quantílica conformalizada

Um desafio persistente na teoria de regiões de predição conforme é fornecer intervalos de predição válidos condicionalmente às covariáveis do ponto de teste. Isso ocorre apesar da garantia de cobertura marginal demonstrada no Teorema 3.4.

Essa garantia não é assegurada pela predição conforme sob a única suposição de permutabilidade. Por sua vez, Barber *et al.* (2020) discute as limitações de abordagens livres de distribuição para garantir cobertura condicional em cenários não triviais, especialmente quando as covariáveis seguem uma distribuição contínua.

Apesar desse perspectiva pessimista, ocorreram avanços empíricos no desenvolvimento de metodologias que aprimorassem a cobertura condicional. Um exemplo notável é a Regressão Quantílica Conformalizada (do inglês, *Conformalized Quantile Regression (CQR)*, descrita em Romano *et al.* (2019)). A técnica combina os conceitos da regressão quantílica com métodos de predição conforme, permitindo a construção de intervalos preditivos que se ajustam às covariáveis. Sejam $q_{Y|X}(x;\alpha_0)$, $q_{Y|X}(x;\alpha_1)$ os quantis condicional de $Y|X=x$, em que $\alpha_0 = \frac{\alpha}{2}$ e $\alpha_1 = 1 - \frac{\alpha}{2}$, por construção,

$$\mathbb{P}(Y \in [q_{Y|X}(X;\alpha_0), q_{Y|X}(X;\alpha_1)]|X=x) \geq 1-\alpha,$$

os quantis condicionais podem ser estimados com boa acurácia usando técnicas de aprendizado de máquina, como *Quantile Regression Forests - QRF*, *Gradient Boosting for Quantile Regression*, *Deep Quantile Regression* e MDNs. A partir disso, podemos construir regiões de predição, da forma

$$R(X_{n+1}) = \{y : [q_{Y|X}(X_{n+1};\alpha_0) - t, q_{Y|X}(X_{n+1};\alpha_1) + t]\}. \tag{3.5}$$

A limitação do uso apenas dos quantis básicos e de um corte fixo t é que suas estimativas são fortemente influenciadas pelo volume de dados. Em várias situações, os intervalos preditivos se assemelham muito ao oráculo em regiões com maior quantidades de dados de treinamento. O problema surge em locais com alta incerteza epistêmica, podendo resultar em estimativas imprecisas e intervalos preditivos inadequados.

Um forma de melhorar a cobertura condicional dos intervalos preditivos é quantificar a incerteza epistêmica. Assim, Rossellini *et al.* (2024) propõem uma nova família de

métodos para conformalizar a regressão quantílica, incorporando essa incerteza. Eles chamam esses métodos de UACQR-S (UncertaintyAware CQR via Scaling) e UACQR-P (UncertaintyAware CQR via Percentiles).

O primeiro método, UACQR-S, contrapõe a abordagem dos métodos convencionais de predição conforme que fixam um corte $t \in \mathbb{R}$ para todos os valores das covariáveis. Neste caso, os quantis básicos são ajustados proporcionalmente a um fator de escala. Definindo:

$$\hat{q}_0(x,t) = \hat{q}_{Y|X}(x,\alpha_0) - t \cdot \hat{s}_0(x),$$

$$\hat{q}_1(x,t) = \hat{q}_{Y|X}(x,\alpha_1) + t \cdot \hat{s}_1(x),$$

onde:

- $\hat{q}_{Y|X}(x,a)$ é um estimador inicial do quantil condicional para $a \in \{\alpha_0, \alpha_1\}$;
- $\hat{s}_0(x)$ e $\hat{s}_1(x)$ são estimadores dos desvios padrão de $\hat{q}_{Y|X}(x,\alpha_0)$ e $\hat{q}_{Y|X}(x,\alpha_1)$, respectivamente.

Por exemplo, podemos usar B amostras *bootstrap* dos dados de treinamento para obter as estimativas $\hat{q}^b_{Y|X}(x,a)$. Em seguida, $\hat{q}_{Y|X}(x,a)$ é calculado como a média das B estimativas *bootstrap*, enquanto $\hat{s}_0(x)$ e $\hat{s}_1(x)$ são os desvios padrão amostrais dessas estimativas.

Aplicando na construção geral (3.5), obtemos o intervalo de predição

$$\hat{R}_n(X_{n+1}) = \{y : y \in \left[\hat{q}_{Y|X}(X_{n+1},\alpha_0) - \hat{t} \cdot \hat{s}_0(X_{n+1}), \hat{q}_{Y|X}(X_{n+1},\alpha_1) + \hat{t} \cdot \hat{s}_1(X_{n+1})\right]\},$$

em que $\hat{t}$ é calculado usando um conjunto de calibração.

O segundo método, UACQR-P, utiliza técnicas de reamostragem para estimar diretamente os quantis das estimativas dos quantis condicional base $\hat{q}_{Y|X}(x,\alpha_0)$ e $\hat{q}_{Y|X}(x,\alpha_1)$. Seja $\hat{q}^{(b)}_{Y|X}(x,a)$ a estatística de ordem b para cada a, com as estimativas ordenadas: $\hat{q}^{(1)}_{Y|X}(x,a) \leq \cdots \leq \hat{q}^{(B)}_{Y|X}(x,a)$. Também, assumindo que $\hat{q}^{(0)}_{Y|X}(x,a) = -\infty$ e $\hat{q}^{(B+1)}_{Y|X}(x,a) = +\infty$. Assim, definindo

$$\hat{q}_0(x,t) = \hat{q}^{(B+1-t)}_{Y|X}(x,\alpha_0), \quad \hat{q}_1(x,t) = \hat{q}^{(t)}_{Y|X}(x,\alpha_1),$$

em que $t \in \mathcal{T} = \{0,1,\ldots,B,B+1\}$, realizando a aplicação na construção geral (3.5),

obtêm-se o seguinte intervalo de predição

$$\hat{R}_n(X_{n+1}) = \{y : y \in \left[\hat{q}_{Y|X}^{(B+1-\hat{t})}(X_{n+1}, \alpha_0), \hat{q}_{Y|X}^{(\hat{t})}(X_{n+1}, \alpha_1)\right]\},$$

em que $\hat{t}$ é calculado usando um conjunto de calibração.

3.5 *Dropout* como uma Aproximação Bayesiana: Representando a Incerteza do Modelo em Aprendizado Profundo

A teoria de Inferência Bayesiana fornece ferramentas matematicamente fundamentadas para estudar a incerteza do modelo, mas estas no geral são computacionalmente intensivas. Essa limitação culmina em inúmeros avanços de pesquisas pela busca de boas aproximações Bayesianas. Uma das técnicas que se destacou foi o uso do *dropout* como uma aproximação Bayesiana, proposta por Gal e Ghahramani (2016). Essa consiste em utilizar uma rede neural com profundidade e não linearidade arbitrárias antes de cada camada de pesos, como uma aproximação variacional matematicamente equivalente do processo Gaussiano profundo probabilístico Damianou e Lawrence (2013). Este resultado nos permite interpretar o procedimento de treinamento com *dropout* como uma inferência Bayesiana aproximada.

3.5.1 Formulação Matemática

Considere uma rede neural com L camadas e uma função de perda $\mathcal{L}$. Denotamos por $\mathbf{W}_l$ as matrizes de pesos da camada l e seja $D = \{(\mathbf{X}_1, Y_1), (\mathbf{X}_2, Y_2), \ldots, (\mathbf{X}_n, Y_n)\}$ o conjunto de treinamento. A distribuição a posteriori dos pesos $\mathbf{W}|D$ é dada pela expressão

$$\mathbb{P}(\mathbf{W}|D) = \frac{f(y|\mathbf{X}, \mathbf{W})\mathbb{P}(\mathbf{W})}{f(y|\mathbf{X})},$$

no entanto, ela é intratável. O termo $\mathbb{P}(\mathbf{W}|D)$ é a distribuição a posteriori dos pesos da rede $\mathbf{W}$, $f(y|\mathbf{X}, \mathbf{W})$ é a verossimilhança, $\mathbb{P}(\mathbf{W})$ é a distribuição a priori dos pesos $\mathbf{W}$ e $f(y|\mathbf{X})$ é a evidência, que é a probabilidade marginal das variáveis respostas Y dado os dados de entrada $\mathbf{X}$. Assim, usamos $q(\mathbf{W})$, uma distribuição sobre matrizes cujas

colunas são configuradas aleatoriamente para zero, para aproximar a posteriori intratável. Definimos $q(\mathbf{W})$ como:

$$q(\mathbf{W}) = \prod_{l=1}^{L} \prod_{i=1}^{K_{l-1}} \mathcal{N}(W_{l,i} \mid M_{l,i}, \sigma^2 I),$$

onde $M_{l,i}$ são os parâmetros variacionais e σ^2 é a variância. O objetivo é minimizar a divergência de Kullback-Leibler entre a distribuição posterior aproximada e a verdadeira posterior. Isso é equivalente a minimizar a seguinte função de perda:

$$\mathcal{L}_{\text{dropout}} = \sum_{i=1}^{n} \mathbb{E}_{q(\mathbf{W})}[\mathcal{L}(y_i, g_{\mathbf{W}}(\mathbf{x}_i))] + \sum_{l=1}^{L} \frac{1}{2} \left(\|M_l\|^2 + \sigma^2 \sum_{i=1}^{K_{l-1}} \|W_{l,i}\|^2 \right),$$

sendo $g_{\mathbf{W}}(\mathbf{x}_i)$ a saída final do modelo para a i-ésima observação em função de $\mathbf{W}$. Esta função de perda pode ser otimizada usando métodos descida de gradiente estocástico já abordados.

3.5.2 Obtenção da Incerteza do Modelo

A derivação desses resultados mostram que a incerteza do modelo pode ser obtida a partir do *dropout*. Assim, temos que a função de distribuição acumulada preditiva $F(y|\mathbf{x}, D)$ é aproximada por:

$$q(y|\mathbf{x}) = \int f(y|\mathbf{x}, \mathbf{W}) q(\mathbf{W}) d\mathbf{W}.$$

Esta integral nos permite aproximar os momentos da distribuição preditiva pela integração de Monte Carlo, amostrando da distribuição variacional $q(\mathbf{W})$.

$$\mathbb{E}_{q(y|\mathbf{x})}[Y|\mathbf{x}, D] \approx \frac{1}{T} \sum_{t=1}^{T} g_{\mathbf{W}^{(t)}}(\mathbf{x}_i),$$

em que $\mathbf{W}^{(t)}$ é uma das T amostras de $q(\mathbf{W})$. Particularmente, nos referimos a essas estimativas Monte Carlo como MC *dropout*. Essas nos permite obter estimativas de incerteza para nossas predições.

Capítulo 4

Incerteza Epistêmica em Inferência Conforme: Uma Abordagem Geral

Neste capítulo, apresentamos o **EPICSCORE** (*Epistemic Conformal Score*, em português *Escore Conforme Epistêmico*), um novo escore de não conformidade desenvolvido para incorporar explicitamente a incerteza epistêmica à metodologia de predição conforme. Diferente de abordagens tradicionais, que frequentemente consideram apenas a incerteza aleatória, nossa proposta utiliza um escore inicial $s(\mathbf{x}, y)$ como base para definir um novo escore que integra tanto a incerteza aleatória quanto a epistêmica, oferecendo uma quantificação de incerteza mais robusta e abrangente. Para tal, definimos um modelo Bayesiano para predizer $s(\mathbf{X}, Y)$ dado $\mathbf{X}$. O modelo é construído da seguinte forma. Seja $\mathcal{F} = \{f(s|\mathbf{x}, \theta) : \theta \in \Theta\}$ uma família de distribuições que modelam a incerteza aleatória de $s(\mathbf{X}, Y)$ dado $\mathbf{X}$. O modelo Bayesiano propõe uma distribuição a priori sobre Θ (ou, equivalentemente, sobre $\mathcal{F}$), por simplicidade assumimos ter densidade $f(\theta)$. Essa priori captura a incerteza epistêmica no processo de geração dos dados. A ideia é utilizar a Rede Neural de Mistura com *Monte Carlo dropout* como aproximação para o processo priori como abordado na Seção 3.5.

No nosso processo de atualização da distribuição a priori $f(\theta)$, utilizamos um conjunto de calibração $\mathcal{D}_{\text{cal}}$. Esse conjunto é dividido em dois subconjuntos disjuntos:$\mathcal{D}_{\text{cal},1}$ e $\mathcal{D}_{\text{cal},2}$. O primeiro subconjunto, $\mathcal{D}_{\text{cal},1}$, é obtido por meio de uma transformação no banco de dados.

$$D = \{(\mathbf{X}, S) : (\mathbf{X}, Y) \in \mathcal{D}_{\text{cal},1},\ S = s(\mathbf{X}, Y)\}.$$

Com o banco de dados transformados, calculamos a distribuição a posteriori $f(\theta|D)$, que reflete a atualização da incerteza epistêmica após o processo de geração dos dados observados. Na perspectiva dos modelos Bayesianos, assumimos que dado θ, os pontos dos dados $(\mathbf{X}, S)$ são independentes e possuem a mesma distribuição condicional $f(s|\mathbf{x}, \theta)$. Assim, a distribuição cumulativa preditiva é dada por

$$F(s|\mathbf{x}, D) = \int F(s|\mathbf{x}, \theta) f(\theta|D) d\theta,$$

em que $f(\theta|D)$ é a distribuição a posteriori e $F(s|\mathbf{x}, \theta)$ é função de distribuição acumulada (FDA) modelada por θ.

Por fim, definimos um novo escore de não conformidade, **EPICSCORE**, como

$$s'(\mathbf{x}, y) = F(s(\mathbf{x}, y)|\mathbf{x}, D). \tag{4.1}$$

A construção de s' incorpora naturalmente a incerteza epistêmica presente em s por meio da média da distribuição original do escore, $F(s|\mathbf{x}, \theta)$, ponderada pela distribuição a posteriori $f(\theta|D)$. Dessa forma, a incerteza sobre θ é dissipada para todo o modelo.

Com o s' calculado, as regiões de predição para as novas observações são obtidas utilizando o método de split-cp padronizado, em que s' desempenha o papel de escore de não conformidade. A prova de que o **EPICSCORE** é de fato um escore conforme é apresentada por Cabezas *et al.* (2025). Especificamente, o s' é avaliado nas amostras do segundo subconjunto, $\mathcal{D}_{\text{cal},2}$. O quantil de $(1 - \alpha)$ desses valores, denotado por $t_{1-\alpha}$, é utilizado na construção da região de predição

$$R_{\text{EPIC}}(\mathbf{x}_{n+1}) = \{y : s'(\mathbf{x}_{n+1}, y) \leq t_{1-\alpha}\},$$

pela definição de s', conseguimos expressar a região de predição em função do escore de não conformidade original s como

$$R_{\text{EPIC}}(\mathbf{x}_{n+1}) = \{y : s(\mathbf{x}_{n+1}, y) \leq F^{-1}(t_{1-\alpha}|\mathbf{x}, D)\}.$$

O método **EPICSCORE** é resumido pelo Algoritmo 1.

Algorithm 1: EPICSCORE

Input: Data $\mathcal{D} = \{(\mathbf{X}_i, Y_i)\}_{i=1}^{n}$, escore conforme $s(\mathbf{X}, Y)$, nível nominal α, observação de teste $\mathbf{X}_{n+1}$

Passo I: Ajustar escores conforme

1: Dividir os dados $\mathcal{D}$ em conjunto de treinamento $\mathcal{D}_{\text{treino}}$ e conjunto de calibração $\mathcal{D}_{\text{cal}}$.

2: Ajustar o escore conforme $s(\mathbf{X}, Y)$ em $\mathcal{D}_{\text{treino}}$.

Passo II: Calcular a função preditiva

1: Dividir os dados $\mathcal{D}_{\text{cal}}$ em conjunto de treino $\mathcal{D}_{\text{cal},1}$ e conjunto de calibração $\mathcal{D}_{\text{cal},2}$.

2: Calcular a função distribuição acumulada da preditiva $F(s|\mathbf{x}, D)$ usando$\mathcal{D}_{\text{cal},1}$.

3: Calcular **EPICSCORE** escore conforme $s'(\mathbf{x}, y)$ para todos os elementos de $\mathcal{D}_{\text{cal},2}$ (Eq. (4.1))

4: Calcular o quantil empírico de $(1-\alpha)$, $t_{1-\alpha}$, dos escores conforme.

Passo III: Calcular o conjunto de predição

3: Calcular o conjunto $R_{\text{EPIC}}(\mathbf{X}_{n+1})$ como:

$$\begin{aligned} R_{\text{EPIC}}(\mathbf{X}_{n+1}) &= \{y : s'(\mathbf{X}_{n+1}, y) \leq t_{1-\alpha}\} \\ &= \{y : s(\mathbf{X}_{n+1}, y) \leq F^{-1}(t_{1-\alpha}|\mathbf{X}_{n+1}, D)\} \end{aligned}$$

Capítulo 5

Avaliação das técnicas para quantificação de incerteza

Neste capítulo, apresentamos os experimentos realizados em dados simulados e em conjuntos de dados reais. O objetivo principal é avaliar o desempenho dos métodos atuais de construção de intervalos preditivos em comparação com a nossa nova abordagem. Denotamos por $\hat{R}(\cdot)$ um intervalo preditivo e as métricas são avaliadas em um conjunto de teste $(\mathbf{X}_1, Y_1), (\mathbf{X}_2, Y_2), \ldots, (\mathbf{X}_m, Y_m)$.

Nos dados reais, algumas limitações surgem nas avaliações devido ao desconhecimento da verdadeira distribuição dos dados. Por isso, adotamos métricas de avaliação amplamente utilizadas na literatura de intervalos preditivos. Essas não necessitam do conhecimento da distribuição dos dados.

A primeira métrica é a **cobertura média dos intervalos**, dada por:

$$\text{Cobertura Média} = \frac{1}{m}\sum_{i=1}^{m} \mathbb{1}\{Y_i \in \hat{R}(X_i)\},$$

em que m é o tamanho do conjunto de dados, Y_i é a variável resposta e $\hat{R}(X_i)$ representa o intervalo preditivo associado à covariável X_i.

A segunda métrica é o **perda média de pontuação do intervalo** (*Average Interval Score Loss - AISL*), expressa por:

$$\text{AISL} = \frac{1}{m}\sum_{i=1}^{m}\left[(\hat{R}_u(\mathbf{X}_i) - \hat{R}_l(\mathbf{X}_i)) + \frac{2}{\alpha}\cdot(\hat{R}_l(\mathbf{X}_i) - Y_i)_+ + \frac{2}{\alpha}\cdot(Y_i - \hat{R}_u(\mathbf{X}_i))_+\right],$$

em que $\hat{R}_l(X_i)$ e $\hat{R}_u(X_i)$ são os limites inferior e superior do intervalo preditivo, $1-\alpha$

é o nível de credibilidade, e $(z)_+ = \max(z, 0)$ denota a parte positiva de z. O AISL é uma métrica usada para avaliar a qualidade de intervalos preditivos, considerando tanto a largura do intervalo quanto a penalização por coberturas inadequadas (quando a verdade real não está dentro do intervalo).

A terceira métrica trata-se do **comprimento médio dos intervalos**:

$$\frac{1}{m}\sum_{i=1}^{m} \max \hat{R}(\mathbf{X}_i) - \min \hat{R}(\mathbf{X}_i),$$

que reflete a precisão dos intervalos de predição. A ideia é que valores altos dessa métrica indicam intervalos poucos informativos, enquanto valores baixos indica intervalos mais concisos e precisos.

A última métrica analisada consiste na **correlação de *Pearson* entre a cobertura e a amplitude do intervalo** (ρ) (Feldman *et al.*, 2021), que fornece indicativos de violação da cobertura condicional. A expressão é dada por

$$\rho = \left| \frac{\text{Cov}(\mathbf{C}, \mathbf{W})}{\sigma_{\mathbf{C}} \sigma_{\mathbf{W}}} \right|,$$

em que $\mathbf{C} = (C_1, C_2, \ldots, C_m)$ representa um vetor binário, cujos componentes $C_i = \mathbb{1}(Y_i \in \hat{R}(\mathbf{X}_i))$ indicam se o valor de Y_i está dentro do intervalo $\hat{R}(\mathbf{X}_i)$, $\mathbf{W} = (W_1, W_2, \ldots, W_m)$, com $W_i = \max \hat{R}(\mathbf{X}_i) - \min \hat{R}(\mathbf{X}_i)$, $\text{Cov}(\mathbf{C}, \mathbf{W})$ é a covariância entre $\mathbf{C}$ e $\mathbf{W}$, e $\sigma_{\mathbf{C}}$ e $\sigma_{\mathbf{W}}$ são os desvios padrão de $\mathbf{C}$ e $\mathbf{W}$, respectivamente. A proposta dessa métrica é que valores elevados da correlação entre a cobertura e a amplitude do intervalo sugerem violação da cobertura condicional, pois, a princípio, espera-se independência entre essas duas medidas (Feldman *et al.*, 2021). No entanto, $\rho = 0$ não garante cobertura condicional, ou seja, é uma métrica que fornece indicativos, mas não uma medida definitiva.

Essas métricas permitem avaliar a qualidade dos intervalos gerados, considerando tanto sua cobertura quanto sua largura e acurácia dos mesmos, além de fornecerem indicativos de que os intervalos de predição possuem cobertura condicional. A análise possibilitará a comparação dos métodos existentes com a abordagem proposta. O método de construção da região/intervalo de predição com o **EPICSCORE** com a MDN com Monte Carlo *dropout* [1].

[1] para simplificação, nos referiremos como EPIC-MDN

5.1 Estudo de simulação

Nos experimentos com dados simulados, a avaliação será realizada por meio da visualização dos intervalos preditivos no caso de uma única covariável e da análise da cobertura condicional. O exemplo abordado visa simular dados com alta incerteza epistêmica, consistindo em supor que $X \sim \text{Beta}(1,2;0,8)$, dessa forma temos poucos valores gerados para $X < 0,4$, incorporando a incerteza epistêmica. E supondo que $Y \sim \text{Normal}(\text{sen}(X^{-3}); X^4)$. Esse exemplo foi retirado do artigo da Rossellini *et al.* (2024). Os detalhes de implementação de todos os métodos avaliados são fornecidos no Apêndice A. Na construção dos intervalos de predição de 90%, utilizamos $\alpha = 0,1$. No treinamento foi usado uma amostra de tamanho $n = 100$ e $p = 1$ covariável.

O intervalo de predição oráculo com 90% de confiança é derivado da distribuição condicional de $Y|X = x$, que, conforme o exemplo segue uma distribuição $\text{Normal}(\text{sen}(X^{-3}); X^4)$. Assim, a expressão do intervalo oráculo é dada por:

$$IO[y; 90\%] = \text{sen}(x^{-3}) \pm z_{0,95} \cdot x^2,$$

em que $z_{0,95}$ é o quantil de 95% da distribuição $\text{Normal}(0;1)$ (aproximadamente $1,645$).

Examinando a Figura 5.1, observa-se que, na região próxima de $X = 1$, o oráculo apresenta uma amplitude maior, enquanto os métodos CQR, UACQR-P e UACQR-S tenderam a subestimar a cobertura. Em contraste, o EPIC-MDN demonstra uma proximidade mais acentuada em relação ao oráculo. Essa região é caracterizada por uma alta incerteza aleatória, uma vez que a variância de $Y|X = x$ é significativamente maior para $x \in [0,6; 1]$, o que reforça a eficácia da nossa abordagem em capturar adequadamente essa incerteza. No entanto, para valores de x muito próximos de 1, o EPIC-MDN demonstra uma leve sobrecobertura. Esse comportamento pode ser atribuído principalmente ao uso de um conjunto de dados relativamente pequeno, com apenas 100 observações (divididas entre treinamento e calibração), o que impacta diretamente a eficiência do procedimento de *split-conformal prediction*.

Por outro lado, em regiões com maior incerteza epistêmica, especialmente para x próximos de 0, observa-se que, dentre os quatro métodos avaliados, apenas o UACQR-P e o EPIC-MDN produziram intervalos preditivos próximos ao oráculo, com uma ligeira vantagem para o UACQR-P. Esse resultado sugere que ambos os métodos são capazes de incorporar a incerteza epistêmica nos intervalos de predição de maneira eficiente.

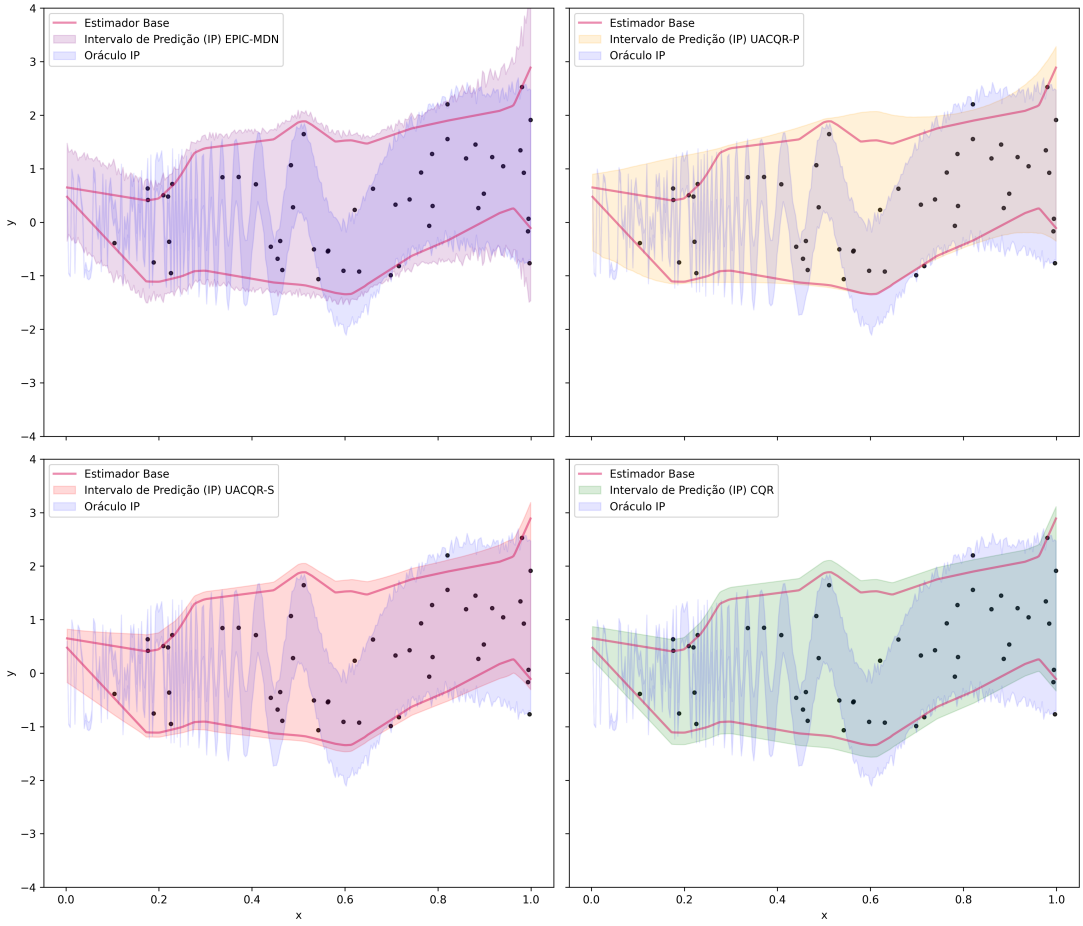

Figura 5.1: Intervalos de predição para métodos existentes e o EPIC-MDN em uma única extração aleatória do processo gerador de dados. Cada procedimento conformal é executado na mesma rede neural ajustada. Para uma melhor visualização fixamos o eixo y ao intervalo $[-4, 4]$. O intervalo de predição oráculo é obtido como $[q_{Y|X}(x; \frac{\alpha}{2}), q_{Y|X}(x; 1 - \frac{\alpha}{2})]$

De modo geral, embora o EPIC-MDN tenha sido afetado pelo tamanho reduzido da amostra, ele se mostra robusto na incorporação de ambos os tipos de incerteza (aleatória e epistêmica) em seus intervalos de predição, evidenciando seu potencial para aplicações em cenários com alta variabilidade e incerteza.

Pela Figura 5.2, observa-se um resultado interessante: embora a análise dos intervalos de predição indique leve superioridade do UACQR-P para valores de x próximos de 0, o EPIC-MDN apresenta cobertura condicional mais próxima do nível nominal de 90% na região de maior incerteza epistêmica. Para valores de $x \in [0,2; 0,8]$, todos os métodos avaliados apresentam um comportamento bem parecido, com uma cobertura condicional oscilando entre 0,8 e 1. Por fim, na região de maior incerteza aleatória, $x \in [0,8; 1]$, os três métodos existentes (CQR, UACQR-P e UACQR-S) mostraram-se mais próximos de 90%, enquanto o EPIC-MDN apresenta uma leve sobrecobertura nessa região, conforme

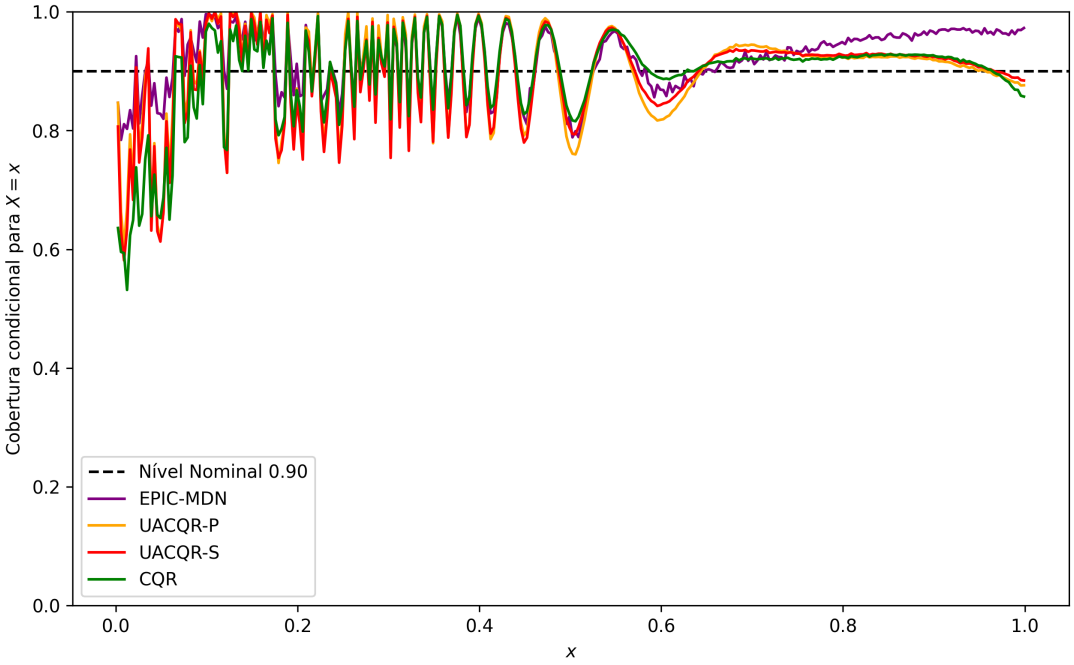

Figura 5.2: Cobertura condicional dos métodos existentes e do EPIC-MDN ao longo de 150 extrações aleatórias, com a mesma configuração da Figura 5.1.

já indicado na Figura 5.1.

5.2 Aplicação em base de dados reais

A fim de avaliar a performance dos métodos na construção de intervalos preditivos, selecionamos as base de dados reais descritas na Tabela 5.1. A ideia é explorar o desempenho em diferentes cenários, em número covariáveis e de observações.

Considerando o interesse em avaliar a versatilidade do método em diferentes contextos de aplicação, investigamos o impacto da escolha do escore utilizado na construção dos intervalos de predição. Para isso, comparamos dois tipos de escores conformes, definidos na Seção 3.3: o escore de divisão baseado em regressão clássica (residual absoluto) e o escore derivado da regressão quantílica conformalizada. Essa análise visa compreender como diferentes estratégias de conformalização afetam a qualidade dos intervalos preditivos gerados pelo EPIC-MDN, especialmente em termos de cobertura e comprimento médio, sob distintas características de base de dados.

Tabela 5.1: Descrição das bases de dados selecionadas, mostrando o número de covariáveis p, amostras n, endereço para acesso e uma breve descrição da variável resposta e das principais características.

Base de dados	n	p	Endereço	Descrição
Airfoil	1503	5	Airfoil (UCI)	Variável resposta: nível de pressão sonora (dB). As *features* (covariáveis) incluem frequência, ângulo de ataque, comprimento da corda, velocidade do fluxo e espessura da camada de deslocamento.
Concrete	1030	8	Concrete (UCI)	Variável resposta: resistência à compressão do concreto (MPa). As *features* descrevem proporções de materiais como cimento, água, escória e idade da amostra.
Cycle	9568	4	Cycle (UCI)	Variável resposta: energia elétrica líquida gerada (MW). As *features* incluem temperatura ambiente, pressão, umidade relativa e vácuo de exaustão.
Electric	10000	12	Electric (UCI)	Variável resposta: estabilidade do sistema elétrico (binária). As variáveis explicativas são medições elétricas simuladas, como tensão, corrente e frequência.
Meps19	15781	141	Meps19 (AHRQ site)	Variável resposta: custos e uso de serviços de saúde. As *features* incluem idade, renda, condições de saúde, tipo de seguro, entre outras informações demográficas e clínicas.
Star	2161	48	Star (Harvard Dataverse)	Variável resposta: desempenho escolar. As *features* consideram tamanho de turma, características do professor e atributos dos estudantes.
WineRed	1599	11	Wine red (UCI)	Variável resposta: qualidade sensorial do vinho tinto (nota de 0 a 10). As variáveis independentes incluem acidez, teor alcoólico, pH, sulfatos, entre outras.
WineWhite	4898	11	Wine white (UCI)	Igual ao conjunto WineRed, mas aplicado a vinhos brancos, com mesmas *features* químicas e mesma variável de interesse.

5.2.1 Avaliação dos intervalos preditivos com divisão de regressão

Para os escores conformais baseados em regressão, utilizamos uma rede neural otimizada com uma função de perda de Erro Quadrático Médio penalizado. Descrições detalhadas das arquiteturas e hiper-parâmetros empregados podem ser encontradas no Apêndice A.2.

Comparamos nosso método EPIC-MDN com as seguintes abordagens de referência:

- **Divisão de regressão**(*Regression-split*): método conformal baseado em resíduos de um modelo de regressão, conforme descrito Seção 3.3.
- **Regressão Ponderada** (*Weighted*): escore conforme apresentado na Seção 3.3.

O Desvio Absoluto Médio (DAM) é modelado regredindo os resíduos absolutos do conjunto de treinamento em $\mathbf{X}$, utilizando a mesma arquitetura do preditor base.

- **Regressão Conforme Mondrian** (Boström e Johansson, 2020): técnica que melhora a cobertura condicional por meio de particionamento adaptativo do espaço de características e que usa um esquema de discretização baseado em variância condicional. A variância é estimada ajustando uma Floresta Aleatória a $(\mathbf{X}, Y)$.

Tabela 5.2: Valores médios de cobertura para regressão em diferentes métodos e conjuntos de dados. Os valores representam a média de 50 execuções, com duas vezes o desvio padrão entre parênteses. Como esperado para métodos conformais, todas as abordagens mantêm cobertura marginal próxima do nível nominal de 0,9.

Dataset	EPIC-MDN	Mondrian	Reg-split	Weighted
airfoil	0,897 (0,009)	0,906 (0,006)	0,897 (0,007)	0,9 (0,007)
concrete	0,907 (0,009)	0,929 (0,006)	0,901 (0,008)	0,896 (0,008)
cycle	0,896 (0,004)	0,905 (0,003)	0,898 (0,003)	0,9 (0,002)
electric	0,896 (0,003)	0,905 (0,002)	0,899 (0,003)	0,901 (0,003)
meps19	0,897 (0,003)	0,902 (0,006)	0,9 (0,002)	0,9 (0,002)
star	0,906 (0,006)	0,913 (0,005)	0,903 (0,004)	0,9 (0,004)
winered	0,895 (0,008)	0,91 (0,005)	0,903 (0,006)	0,895 (0,005)
winewhite	0,901 (0,004)	0,911 (0,003)	0,9 (0,004)	0,899 (0,003)

A Tabela 5.2 mostra que nossa abordagem e as demais constroem intervalos de predição conformes, ou seja, há evidências de cobertura marginalmente válida.

Tabela 5.3: Valores AISL de regressão para cada método e conjunto de dados. Os valores reportados representam a média de 50 execuções, com duas vezes o desvio padrão entre parênteses. Valores em negrito destacam o método com desempenho superior dentro de um intervalo de confiança de 95%.

Dataset	EPIC-MDN	Mondrian	Reg-split	Weighted
airfoil	**19,823 (0,675)**	21,532 (0,919)	21,201 (0,98)	**20,276 (0,819)**
concrete	**51,648 (2,185)**	61,915 (2,815)	**54,902 (2,634)**	58,399 (3,165)
cycle	**19,436 (0,213)**	**19,403 (0,226)**	**19,73 (0,208)**	**19,49 (0,207)**
electric	**0,048 (<0,001)**	0,05 (<0,001)	0,05 (0,001)	**0,048 (<0,001)**
meps19	**75,061 (1,807)**	79,192 (1,821)	109,83 (2,695)	92,433 (3,259)
star $\times(10^1)$	**106,368 (1,173)**	109,346 (1,119)	**105,250 (1,038)**	129,492 (1,657)
winered	**3,101 (0,062)**	3,262 (0,069)	**3,214 (0,063)**	3,415 (0,067)
winewhite	**3,129 (0,029)**	**3,087 (0,023)**	3,181 (0,028)	3,189 (0,033)

Observando a Tabela 5.3, nota-se que, sob a métrica *AISL*, a abordagem EPIC-MDN apresenta desempenho majoritariamente superior na geração de intervalos preditivos em

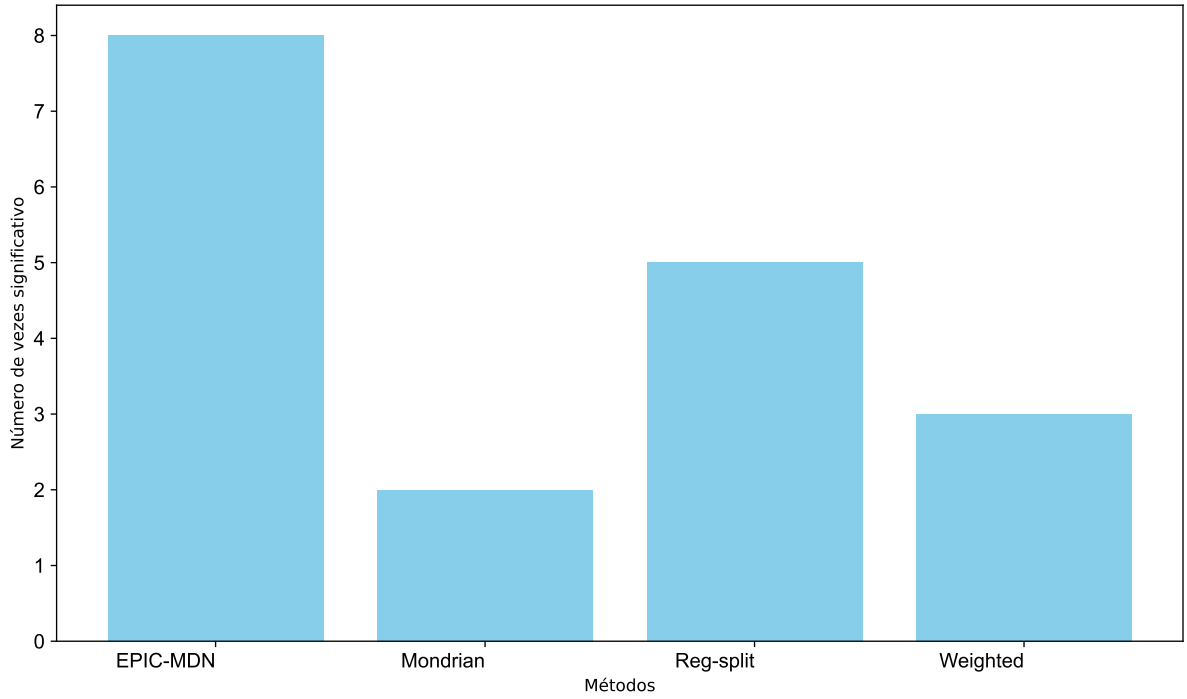

Figura 5.3: Número de vezes que cada método foi significativamente melhor que os demais segundo o critério AISL.

comparação com as demais abordagens conformes. A Figura 5.3 reforça a superioridade de nossa abordagem, pois evidencia que ela foi significativamente melhor em oito bancos de dados distintos. Esses resultados demonstram não apenas sua capacidade de desempenho consistente na geração de intervalos de predição, mas também a eficácia da arquitetura baseada em densidade de mistura na captura de incertezas preditivas.

Tabela 5.4: Valores de comprimento de intervalo para regressão em diferentes métodos e conjuntos de dados. Os valores representam a média de 50 execuções, com duas vezes o desvio padrão entre parênteses. Valores em negrito indicam o método com melhor desempenho dentro de um intervalo de confiança de 95%.

Dataset	**EPIC-MDN**	**Mondrian**	**Reg-split**	**Weighted**
airfoil	**15,089 (0,56)**	16,671 (0,632)	**15,325 (0,469)**	**15,693 (0,521)**
concrete	**40,475 (1,551)**	51,284 (1,904)	**39,943 (1,204)**	43,053 (1,629)
cycle	**14,712 (0,174)**	**15,015 (0,164)**	**14,855 (0,159)**	**14,833 (0,15)**
electric	**0,036 (<0,001)**	0,038 (<0,001)	0,037 (<0,001)	0,037 (<0,001)
meps19	32,093 (1,446)	38,904 (1,039)	**28,899 (0,544)**	29,555 (0,843)
star $\times(10^1)$	**86,499 (1,288)**	90,539 (0,762)	**85,230 (0,792)**	104,202 (1,306)
winered	**2,316 (0,054)**	2,576 (0,04)	**2,39 (0,037)**	2,541 (0,051)
winewhite	**2,356 (0,031)**	2,445 (0,013)	2,361 (0,014)	2,387 (0,015)

A análise conjunta das Tabelas 5.4 e 5.5 revela que, na maioria dos bancos de dados

Tabela 5.5: Valores de correlação de Pearson para regressão em diferentes métodos e conjuntos de dados. Os valores representam a média de 50 execuções, com duas vezes o desvio padrão entre parênteses. Valores em negrito indicam o método com melhor desempenho dentro de um intervalo de confiança de 95%. O método *Regression-split* produz intervalos com o mesmo tamanho, tornando impossível o cálculo da correlação entre o tamanho do intervalo e cobertura.

Dataset	EPIC-MDN	Mondrian	Weighted
airfoil	**0,054 (0,012)**	0,148 (0,017)	0,124 (0,016)
concrete	**0,054 (0,011)**	0,191 (0,02)	0,211 (0,022)
cycle	0,023 (0,005)	0,043 (0,006)	0,025 (0,005)
electric	**0,024 (0,005)**	0,047 (0,007)	0,029 (0,006)
meps19	**0,034 (0,008)**	**0,022 (0,006)**	0,053 (0,016)
star	0,073 (0,01)	0,156 (0,012)	0,335 (0,016)
winered	**0,042 (0,009)**	0,153 (0,018)	0,221 (0,019)
winewhite	**0,025 (0,005)**	0,055 (0,009)	0,092 (0,011)

analisados, nossa abordagem produziu intervalos preditivos mais estreitos com baixa correlação entre cobertura e amplitude intervalar. Esses resultados de menor comprimento dos intervalos e baixa correlação de *Pearson*, sugerem que os intervalos obtidos possuem cobertura condicionalmente válida. Nossos achados demonstram que a abordagem proposta, assim como o método de Rossellini *et al.* (2024) que incorpora incerteza epistêmica, proporciona uma melhoria significativa na qualidade dos intervalos preditivos.

5.2.2 Avaliação dos intervalos preditivos com regressão quantílica conformalizada

Tabela 5.6: Valores médios de cobertura para cada método e conjunto de dados. A média de 50 execuções é relatada com duas vezes o desvio padrão entre parênteses.

Dataset	EPIC-MDN	CQR	CQR-r	UACQR-P	UACQR-S
airfoil	0,896 (0,01)	0,901 (0,007)	0,901 (0,007)	0,907 (0,009)	0,9 (0,007)
concrete	0,898 (0,011)	0,897 (0,007)	0,897 (0,007)	0,914 (0,012)	0,895 (0,007)
cycle	0,898 (0,004)	0,901 (0,003)	0,902 (0,003)	0,901 (0,002)	0,9 (0,002)
electric	0,901 (0,004)	0,901 (0,002)	0,901 (0,002)	0,901 (0,002)	0,901 (0,002)
meps19	0,899 (0,002)	0,899 (0,002)	0,899 (0,002)	0,901 (0,002)	0,899 (0,002)
star	0,903 (0,006)	0,902 (0,004)	0,901 (0,004)	0,93 (0,013)	0,902 (0,004)
winered	0,904 (0,007)	0,897 (0,006)	0,897 (0,006)	0,903 (0,009)	0,897 (0,006)
winewhite	0,9 (0,004)	0,898 (0,003)	0,898 (0,003)	0,908 (0,009)	0,898 (0,003)

A Tabela 5.6 mostra que os métodos de fato constroem intervalos preditivos conformes,

pois a cobertura média dos intervalos são todas aproximadamente 0.9.

Tabela 5.7: Valores de AISL para cada método e conjunto de dados. A média de 50 execuções é relatada com duas vezes o desvio padrão entre parênteses. Os valores em negrito destacam o método com melhor desempenho comparando o intervalo de confiança de 95%.

Dataset	EPIC-MDN	CQR	CQR-r	UACQR-P	UACQR-S
airfoil	**18,819 (0,293)**	20,521 (0,234)	20,535 (0,236)	23,021 (0,337)	20,188 (0,3)
concrete	**44,434 (0,79)**	46,882 (0,681)	46,896 (0,683)	52,789 (1,097)	47,324 (1,349)
cycle	**34,097 (0,131)**	39,218 (0,134)	39,408 (0,136)	43,775 (0,181)	35,346 (0,197)
electric	**0,083 (<0,001)**	0,102 (<0,001)	0,102 (<0,001)	0,111 (0,001)	0,097 (0,001)
meps19	**64,282 (1,539)**	**64,239 (1,56)**	**64,239 (1,56)**	71,015 (1,763)	**63,737 (1,461)**
star	**988,389 (7,25)**	**977,69 (7,246)**	**977,905 (7,243)**	997,819 (6,467)	998,087 (9,679)
winered	**2,975 (0,05)**	**2,979 (0,069)**	**2,978 (0,069)**	**3,059 (0,069)**	**2,999 (0,063)**
winewhite	**3,218 (0,031)**	3,316 (0,036)	3,315 (0,036)	3,378 (0,038)	**3,2 (0,036)**

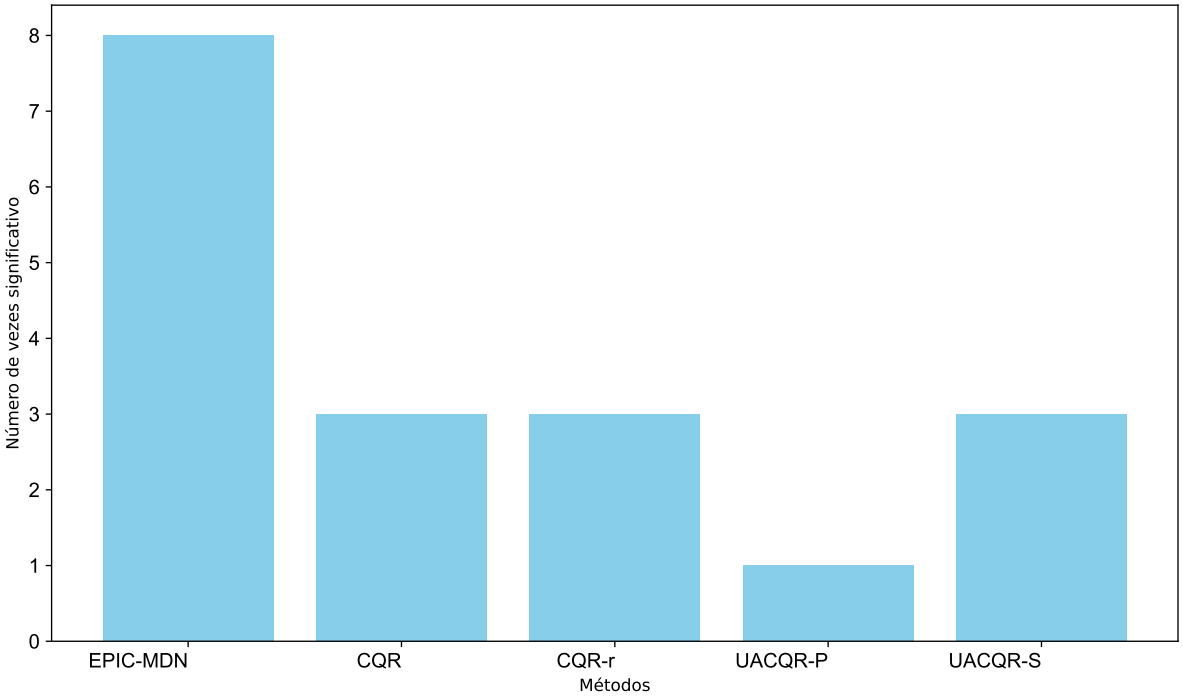

Figura 5.4: Número de vezes que cada método foi significativamente melhor que os demais segundo o critério AISL.

A análise da Tabela 5.7 e da Figura 5.4 revela que, de acordo com a métrica AISL, a abordagem que combina MDN com Monte Carlo *dropout* se mostrou significativamente melhor em oito bases de dados avaliadas.

A análise comparativa das Tabelas 5.8 e 5.9 revela que o método EPIC-MDN produz, em geral, intervalos preditivos com duas características fundamentais: amplitudes significativamente menores e baixa correlação entre amplitude e cobertura. Esses acha-

Tabela 5.8: Valores do tamanho do intervalo para cada método e conjunto de dados utilizando a regressão quantílica. A média de 50 execuções é relatada com duas vezes o desvio padrão entre parênteses. Os valores em negrito destacam o método com melhor desempenho comparando o intervalo de confiança de 95%.

Dataset	**EPIC-MDN**	**CQR**	**CQR-r**	**UACQR-P**	**UACQR-S**
airfoil	**16,02 (0,222)**	17,087 (0,127)	17,09 (0,124)	18,838 (0,367)	16,656 (0,386)
concrete	**37,614 (0,651)**	39,477 (0,353)	39,486 (0,36)	44,425 (1,307)	39,853 (1,536)
cycle	**30,714 (0,146)**	35,235 (0,095)	35,346 (0,093)	38,292 (0,195)	31,045 (0,207)
electric	**0,072 (0,001)**	0,09 (0,001)	0,09 (0,001)	0,097 (0,001)	0,084 (0,001)
meps19	29,16 (0,266)	28,948 (0,249)	28,949 (0,249)	**27,857 (0,314)**	32,763 (0,815)
star $\times(10^1)$	**82,050 (1,083)**	**81,359 (0,508)**	**81,396 (0,5117)**	82,521 (0,618)	83,253 (0,952)
winered	2,11 (0,031)	**1,906 (0,011)**	**1,902 (0,01)**	2,031 (0,025)	2,077 (0,042)
winewhite	2,253 (0,016)	**2,12 (0,006)**	**2,121 (0,006)**	**2,124 (0,012)**	2,222 (0,017)

Tabela 5.9: Valores de correlação de *Pearson* para regressão quantílica em diferentes métodos e conjuntos de dados. Os valores reportados representam a média de 50 execuções, com duas vezes o desvio padrão entre parênteses. Valores em negrito indicam o método com melhor desempenho dentro de um intervalo de confiança de 95%.

Dataset	**EPIC-MDN**	**CQR**	**CQR-r**	**UACQR-P**	**UACQR-S**
airfoil	**0,071 (0,016)**	0,125 (0,017)	0,129 (0,017)	0,132 (0,033)	0,108 (0,02)
concrete	**0,068 (0,013)**	**0,081 (0,017)**	**0,082 (0,017)**	0,121 (0,034)	**0,088 (0,02)**
cycle	**0,085 (0,01)**	0,27 (0,011)	0,292 (0,011)	0,255 (0,011)	0,192 (0,012)
electric	0,159 (0,008)	**0,075 (0,006)**	**0,071 (0,006)**	0,123 (0,01)	0,134 (0,008)
meps19	**0,069 (0,009)**	0,08 (0,006)	0,08 (0,006)	0,128 (0,006)	**0,051 (0,006)**
star	0,085 (0,012)	**0,046 (0,009)**	**0,047 (0,01)**	**0,048 (0,011)**	**0,041 (0,009)**
winered	**0,058 (0,011)**	0,113 (0,015)	0,114 (0,016)	0,097 (0,019)	**0,076 (0,016)**
winewhite	**0,072 (0,012)**	0,147 (0,011)	0,147 (0,011)	0,156 (0,014)	0,099 (0,011)

dos sugerem fortemente que os intervalos gerados pelo EPIC-MDN possuem cobertura condicional válida.

Os resultados demonstram consistentemente a eficácia do EPIC-MDN na construção de intervalos preditivos robustos. que mantém a confiabilidade, apesar da variação do escore conforme base, em diversos cenários com dados reais. A superioridade desse método pode ser atribuída principalmente à sua capacidade de incorporar a incerteza epistêmica de forma eficiente, graças à proposta de um novo escore que considera a incerteza epistêmica do escore primário.

Capítulo 6

Considerações finais

Os resultados deste trabalho comprovam a eficácia do método EPIC-MDN na construção de intervalos preditivos confiáveis, destacando-se em trÊs principais aspectos fundamentais. Primeiramente, a capacidade de incorporar tanto a incerteza aleatória quanto a epistêmica mostrou-se decisiva para o desempenho superior dos métodos, particularmente em cenários com limitação de dados e alta variabilidade. A proposta de combinar Redes de densidade de mistura com a técnica de *Monte Carlo dropout*, permitiu uma quantificação mais refinada das fontes de incerteza, superando algumas das abordagens presentes na literatura.

A avaliação abrangente em diversos conjuntos de dados reais destacou que o EPIC-MDN produz consistentemente intervalos mais precisos (com amplitude intervalar em relação a outros métodos) e indicativos de que os intervalos gerados possuem cobertura condicional. Essa vantagem foi particularmente marcante em comparação com métodos estabelecidos como CQR e suas variantes, conforme demonstrado pelas análises de correlação entre amplitude e cobertura.

A flexibilidade da abordagem constitui outro ponto forte, mantendo bom desempenho independente do escore conforme adotado. Essa característica abre possibilidade da aplicação em diversos contextos, desde que os princípios fundamentais da modelagem da incerteza aleatória e epistêmica da nossa abordagem sejam respeitados.

Algumas da limitações observadas do método EPIC-MDN consiste, especialmente na sensibilidade a tamanhos amostrais reduzidos, observada nos experimentos com dados simulados. Esse comportamento sugere que, apesar da robustez do método na incorporação da incerteza epistêmica, o seu desempenho mais eficiente depende de uma massa crítica de dados para uma calibração precisa dos intervalos.

Como perspectivas futuras, destacam-se a exploração do método EPIC-MDN em problemas de classificação e desenvolvimento de versões escaláveis para grandes volumes de dados. Além disso, avaliar aplicações em domínios onde a quantificação da incerteza é crítica, como diagnóstico médico e estudo do risco financeiro. A incorporação de componentes de misturas que se adaptem ao domínio do escore conformal, para aprimorar a aproximação Bayesiana e consequentemente, a quantificação de ambos os tipos de incerteza.

Este trabalho consolida uma nova abordagem para inferência preditiva, que combina a capacidade de aproximação flexível das redes de densidade de mistura com o rigor teórico da inferência conformal. Os resultados validam empiricamente a relevância da proposta, demonstrando sua capacidade de distinguir e quantificar de forma adequada a incerteza aleatória e a incerteza epistêmica, gerando intervalos preditivos mais precisos e confiáveis.

Referências Bibliográficas

Angelopoulos, A. N. e Bates, S. (2022). A gentle introduction to conformal prediction and distribution-free uncertainty quantification.

Angelopoulos, A. N. e Bates, S. (2023). Conformal prediction: A gentle introduction. *Foundations and Trends® in Machine Learning*, **16**(4), 494–591.

Barber, R. F., Candès, E. J., Ramdas, A. e Tibshirani, R. J. (2020). The limits of distribution-free conditional predictive inference.

Bishop, C. (1994). Mixture density networks. Workingpaper, Aston University.

Boström, H. e Johansson, U. (2020). Mondrian conformal regressors. Em *Conformal and Probabilistic Prediction and Applications*, páginas 114–133. PMLR.

Cabezas, L. M. C., Santos, V. S., Ramos, T. R. e Izbicki, R. (2025). Epistemic uncertainty in conformal scores: A unified approach.

Calonico, S., Cattaneo, M. e Farrell, M. (2018). On the effect of bias estimation on coverage accuracy in nonparametric inference. *Journal of the American Statistical Association*, **113**(522), 767–779.

Cheng, G. e Chen, Y.-C. (2019). Nonparametric inference via bootstrapping the debiased estimator. *Electronic Journal of Statistics*, **13**, 2194–2256.

Damianou, A. e Lawrence, N. D. (2013). Deep Gaussian processes. Em C. M. Carvalho e P. Ravikumar, editors, *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, volume 31 of *Proceedings of Machine Learning Research*, páginas 207–215, Scottsdale, Arizona, USA. PMLR.

Dorogush, A. V., Ershov, V. e Gulin, A. (2018). Catboost: gradient boosting with categorical features support.

Feldman, S., Bates, S. e Romano, Y. (2021). Improving conditional coverage via orthogonal quantile regression. *Advances in neural information processing systems*, **34**, 2060–2071.

Gal, Y. e Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. Em M. F. Balcan e K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, páginas 1050–1059, New York, New York, USA. PMLR.

Gal, Y., Hron, J. e Kendall, A. (2017). Concrete dropout.

Ioffe, S. e Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift.

Izbicki, R. (2025). *Machine Learning Beyond Point Predictions: Uncertainty Quantification*. First edition. ISBN 978-65-01-20272-3.

Izbicki, R. e Lee, A. B. (2017). Converting high-dimensional regression to high-dimensional conditional density estimation.

Izbicki, R., Shimizu, G. e Stern, R. B. (2021). Cd-split and hpd-split: efficient conformal regions in high dimensions.

Kingma, D. P. e Ba, J. (2017). Adam: A method for stochastic optimization.

Kiureghian, A. D. e Ditlevsen, O. (2009). Aleatory or epistemic? does it matter? *Structural Safety*, **31**(2), 105–112. Risk Acceptance and Risk Communication.

Kumar, S. K. (2017). On weight initialization in deep neural networks. *arXiv preprint arXiv:1704.08863*.

Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J. e Wasserman, L. (2018). Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, **113**(523), 1094–1111.

Mcculloch, W. e Pitts, W. (1943). A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, **5**, 127–147.

Nemani, V., Biggio, L., Huan, X., Hu, Z., Fink, O., Tran, A., Wang, Y., Zhang, X. e Hu, C. (2023). Uncertainty quantification in machine learning for engineering design and health prognostics: A tutorial. *Mechanical Systems and Signal Processing*, **205**.

Prechelt, L. (1996). Early stopping-but when? Em G. B. Orr e K.-R. Müller, editors, *Neural Networks: Tricks of the Trade*, volume 1524 of *Lecture Notes in Computer Science*, páginas 55–69. Springer. ISBN 3-540-65311-2.

Romano, Y., Patterson, E. e Candes, E. (2019). Conformalized quantile regression. Em H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, e R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Rossellini, R., Foygel Barber, R. e Willett, R. (2024). Integrating uncertainty awareness into conformalized quantile regression. Em S. Dasgupta, S. Mandt, e Y. Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, páginas 1540–1548. PMLR.

Rumelhart, D. E., Hinton, G. E. e Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, **323**(6088), 533–536.

Shafer, G. e Vovk, V. (2008). A tutorial on conformal prediction. *Journal of Machine Learning Research*, **9**(3), 371–421.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. e Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, **15**(56), 1929–1958.

Vovk, V., Gammerman, A. e Shafer, G. (2005). *Algorithmic Learning in a Random World*. Springer-Verlag, Berlin, Heidelberg. ISBN 0387001522.

Apêndice A

Destalhes de implementação

A.1 Simulação

Utilizamos uma rede neural como estimador base para a regressão quantílica, com uma camada oculta contendo 100 neurônios e função de ativação ReLU. A rede neural é treinada para minimizar a perda quantílica nos quantis-alvo de 0,05 e 0,95. Além disso, normalizamos as variáveis de entrada para melhorar a estabilidade do treinamento. Os parâmetros são inicializados com as configurações padrão do pacote PyTorch e a rede é treinada por 1000 épocas, com um tamanho de lote de 2 e uma taxa de aprendizado inicial de 0.001. Utilizamos o otimizador Adam (Kingma e Ba, 2017).

Para os métodos UACQR, empregamos uma rede neural com agregação baseada no desvio padrão das previsões bootstrap, utilizando $B = 999$ reamostragens. Aplicamos a heurística baseada em épocas para evitar retratamento do modelo. No caso do UACQR-S, CQR e CQR-r, o estimador base $\hat{q}_{Y|X}(x, a)$ corresponde à rede neural treinada completamente, $\hat{q}^{B}_{Y|X}(x, a)$.

Além disso, utilizamos o método EPIC-SCORE com uma Rede de Mistura de Densidade (MDN) baseada no *dropout* para estimar distribuições condicionais. O modelo é composto por duas camadas ocultas com 64 e 32 neurônios, respectivamente, e emprega a técnica Monte Carlo *dropout* para capturar incerteza epistêmica. O MDN utiliza 2 componentes de mistura gaussianos, com um fator de *dropout* de 0,5. O treinamento ocorre por até 2000 épocas, com um critério de parada antecipada baseado na paciência de 50 épocas. Normalizamos as saídas da rede e utilizamos um tamanho de lote de 5.

A.2 Dados reais

A.2.1 Estimador base

No contexto de regressão, implementamos uma Rede Neural com três camadas ocultas, contendo 64, 32 e 16 neurônios, respectivamente. Cada camada emprega uma função de ativação *ReLU*, normalização por lotes e taxas de *dropout* de 0.2, 0.1 e 0.05, nesta ordem, ambas as técnicas apresentadas na Seção 2.3. O modelo é treinado com a função de perda L1 suavizada (*smooth L1 loss*), que equilibra o erro absoluto médio e o erro quadrático médio, conferindo robustez a *outliers* enquanto mantém atualizações de gradiente estáveis.

Para otimização, utilizamos o algoritmo Adam, citado na Seção 2.2 com taxa de aprendizado inicial de 0.01. Um agendador de taxa de aprendizado (learning rate scheduler) reduz o valor por um fator de 0.5 caso não haja melhoria após 10 épocas, acelerando a convergência. Todos os pesos são inicializados com o método Xavier normal (Kumar, 2017). Destinamos 30% dos dados de treinamento para validação, com tamanho de lote fixado em 35. O treinamento executa no máximo 750 épocas, com parada antecipada (*early stopping*) acionada se não houver melhoria no conjunto de validação por 30 épocas consecutivas. Adicionalmente, aplicamos escalonamento de características (*feature scaling*) e normalização min-max da variável alvo para garantir estabilidade durante o treinamento.

Para o modelo de regressão quantílica *CatBoost* (Dorogush *et al.*, 2018), definimos o número de iterações (árvores) como 1.000 e a taxa de aprendizado como 0.001, habilitando a parada antecipada após 50 rodadas sem melhoria. Para mitigar ainda mais o sobreajuste, limitamos cada árvore a uma profundidade máxima de 6. Todos os outros parâmetros seguem as configurações padrão do *CatBoost*.

A.2.2 Técnica de split

Ao dividir o conjunto de calibração $\mathcal{D}_{\text{cal}}$ para ajustar o modelo preditivo e derivar os pontos de corte adaptativos do **EPICSCORE** em subconjuntos de dados disjuntos, priorizamos alocar a maior parte dos dados para o treinamento do modelo, uma vez que a derivação dos pontos de corte envolve principalmente uma computação de quantis mais simples. Especificamente, para conjuntos pequenos com $n \leq 3000$, reservamos 30% das amostras de calibração para o cálculo dos pontos de corte. Para conjuntos de dados maiores, essa alocação é limitada a 1000 amostras para manter a eficiência computacional.

Essa abordagem garante que o **EPICSCORE** alcance estimativas precisas da distribuição preditiva e pontos de corte bem calibrados.

A.2.3 EPIC-MDN

A Rede de Densidade de Mistura (Mixture Density Network, MDN) empregada consiste em 2 camadas ocultas, cada uma com 64 neurônios, ativações ReLU e normalização em lote (*batch normalization*) para melhorar a estabilidade do treinamento. A camada de saída prevê três parâmetros para cada componente da mistura $k = \{1, \ldots, K\}$: a probabilidade de pertencimento $\pi_k(\cdot)$, a média da mistura $\mu_k(\cdot)$ e a variância da mistura $\sigma_k^2(\cdot)$. Definimos o número de componentes como $K = 3$, resultando em 9 parâmetros de saída. A rede é treinada usando a função de perda de log-verossimilhança negativa, aplicando uma ativação softmax às probabilidades da mistura e uma ativação softplus às estimativas de variância.

A otimização é realizada usando o otimizador Adam com uma taxa de aprendizado de 0,001 e um agendador de taxa de aprendizado que decai em 0,99 a cada 5 épocas. Para modelar a incerteza epistêmica, incorporamos o MC *dropout*, aplicando uma taxa de *dropout* de 0,5 a cada camada oculta. Todos os pesos são inicializados usando as configurações padrão do PyTorch. Para monitorar o desempenho de generalização, reservamos 30% dos dados de treinamento exclusivamente para validação. O modelo é treinado por até 2.000 épocas, com parada antecipada acionada após 50 épocas sem melhoria no conjunto de validação para evitar sobreajuste. Além disso, aplicamos escalonamento de características e normalização de destino para melhorar a estabilidade numérica e a estimativa de parâmetros.

A seleção do tamanho do lote depende do conjunto de dados: 40 para conjuntos de dados pequenos ($n < 10000$), 125 para conjuntos de dados médios ($n < 50000$) e 250 para conjuntos de dados em grande escala, como o WEC. Para o experimento com imagens, introduzimos uma camada oculta adicional com 32 neurônios e mantemos uma taxa de *dropout* de 0,5, enquanto ajustamos o tamanho do lote para 135. Para calcular a Função de Distribuição Acumulada (FDA) preditiva $F(s(\mathbf{X}, Y)|\mathbf{X}, D)$ em $\mathbf{X} = \mathbf{x}$, geramos 500 amostras dos parâmetros da mistura usando passos diretos de MC *dropout*. Para cada conjunto de parâmetros da mistura amostrado, as amostras de pontuação são extraídas do Modelo de Mistura Gaussiana. A FDA preditiva final é obtida calculando a distribuição empírica de $s(\mathbf{X}, Y)$ sobre essas amostras de pontuação.