



Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Estatística

Dissertação de Mestrado

Regressão Quantílica Multitarefa via Redes Neurais Profundas

por

Estevão Sérgio Zeco

Brasília, 01 de Outubro de 2024

Regressão Quantílica Multitarefa via Redes Neurais Profundas

por

Estevão Sérgio Zeco

Dissertação apresentada ao Departamento de Estatística da Universidade de Brasília, como requisito parcial para obtenção do título de Mestre em Estatística.

Orientadora: Profa. Dra. Thaís C. V. Rodrigues

Brasília, 01 de Outubro de 2024

Dissertação submetida ao Programa de Pós-Graduação em Estatística do Departamento de Estatística da Universidade de Brasília como parte dos requisitos para a obtenção do título de Mestre em Estatística.

Texto aprovado por:

Profa. Dra. Thaís Carvalho V. Rodrigues
Orientadora, EST/UnB

Prof. Dr. José Augusto Fiorucci
Membro interno, EST/UnB

Profa. Dra. Ana Carolina C. Nascimento
Membro externo, DET/UFV

Todos os modelos estão errados, mas alguns são úteis.

(George E. P. Box)

Dedicatória

Dedico este trabalho a mim mesmo, por todo o esforço, dedicação e perseverança que me trouxeram até aqui. Realizar este sonho de me tornar mestre em Estatística em uma universidade no Brasil é uma conquista pessoal que representa anos de estudo, sacrifícios e superação de desafios. Este é um marco importante na minha trajetória, que reflete não apenas o conhecimento adquirido, mas também o crescimento pessoal e profissional.

Dedico também esta dissertação aos meus pais, que sempre acreditaram em mim e me apoiaram em cada etapa do caminho. Sem o amor, encorajamento e ensinamentos deles, nada disso seria possível. Eles são a base sólida sobre a qual construí minhas conquistas, e é com grande orgulho que compartilho este sucesso com eles.

Finalmente, dedico este trabalho a todos que, de alguma forma, contribuíram para a minha formação, seja através de conselhos, apoio emocional ou colaboração acadêmica. Cada pessoa que cruzou meu caminho durante essa jornada deixou uma marca indelével que me ajudou a chegar aonde estou hoje. Este título não é apenas meu, mas também de todos que fizeram parte dessa caminhada.

Agradecimentos

Primeiramente, agradeço a Deus pela vida, proteção e pelas inúmeras bênçãos derramadas sobre mim, que tornaram possível mais essa conquista, fruto de Sua graça infinita.

Aos meus pais, Sérgio Fernando Zeco e Arlinda Dalilo Comé, expresso minha eterna gratidão pelo amor incondicional, pelo apoio constante e pelos ensinamentos que moldaram quem sou hoje. Suas palavras de encorajamento, força e confiança foram a base de cada passo que dei ao longo dessa jornada. Parte deste sucesso pertence a vocês, e sou profundamente grato por tudo que fizeram por mim.

Aos meus irmãos, Ernesto Zeco, Ragildo Zeco, Rasmina Zeco e Sidónia Zeco, sou imensamente grato pelo apoio incondicional, por acreditarem em mim e por estarem sempre ao meu lado, mesmo nos momentos mais desafiadores. Vocês foram um alicerce essencial nesta caminhada.

À minha namorada, Isabel Savanguana, agradeço pelo carinho, apoio e companheirismo. Apesar da distância que nos separa, você sempre esteve presente de coração e alma, me fortalecendo nos momentos em que mais precisei. Sua compreensão e amor tornaram essa jornada

mais leve.

Ao Prof. Dr. Miranda Muaualo e ao Prof. Dr. Raúl Chambote, minha sincera gratidão por acreditarem em mim e por me apoiarem de maneira incondicional ao longo desta trajetória. Um agradecimento especial ao Prof. Dr. Frederico Almeida, que acompanhou este processo desde o início até o fim, sendo mais que um professor, um verdadeiro amigo e irmão.

À minha orientadora, Profa. Dra. Thaís Rodrigues, sou profundamente grato pela paciência, disponibilidade e pelos valiosos contributos que enriqueceram cada etapa deste trabalho. Sua orientação foi crucial para que este projeto se concretizasse com sucesso.

Agradeço também à banca avaliadora de qualificação, composta pelos Profs. Drs. Helton dos Santos e José Augusto Fiorucci, do Departamento de Estatística da Universidade de Brasília, pelas contribuições significativas que ajudaram a moldar este trabalho. Desde já, agradeço a Profa. Dra. Ana Carolina C. Nascimento da Universidade Federal de Viçosa, por aceitar o convite para compor minha banca de defesa final, ao lado dos professores já mencionados.

Sou extremamente grato a todos os professores do Programa de Pós-Graduação em Estatística da Universidade de Brasília (PPGEST/UnB) que, com dedicação e atenção, contribuíram para minha formação, especialmente à Profa. Dra. Cira Otiniano, ao Prof. Dr. Guilherme Rodrigues e ao Prof. Dr. Raul Matsushita. Estendo meus agradecimentos à equipe da secretaria do PPGEST/UnB, que sempre demonstrou um compromisso exemplar durante todo esse processo.

Por fim, meu muito obrigado aos meus colegas e companheiros de jornada, Carlo Aleksandr, João Victor de Andrade, Leonardo da Cruz, Mariana Fehr, Melquisadec de Souza Oliveira, Moisés Lima, Ofélia Mula, Rebeca Klamerick, Reinaldo e Tiago Chandiona. A convivência com vocês tornou essa caminhada mais leve e dinâmica, e sou grato pela troca de conhecimentos e pela amizade que levarei para a vida toda.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Resumo

A regressão quantílica via redes neurais tem se consolidado como uma abordagem poderosa para modelar relações não lineares entre variáveis, permitindo a estimação individual de quantis condicionais. No entanto, essa técnica enfrenta desafios quando é necessário estimar múltiplos quantis simultaneamente. Neste trabalho, propomos uma nova arquitetura para Regressão Quantílica Multitarefa via Redes Neurais Profundas, visando aprimorar tanto a precisão quanto a eficiência computacional na estimação de quantis. Baseada na arquitetura de Kuleshov e Deshpande (2022), essa nova arquitetura, denominada RQRN1E, inova ao unificar dois estágios da arquitetura original em um único estágio e ao introduzir o logito do τ diretamente na penúltima camada intermediária da rede. Nos três conjuntos de dados avaliados, os resultados evidenciaram que o modelo RQRN1E superou os modelos concorrentes, apresentando consistentemente a menor perda quantílica e a melhor cobertura para os quantis, o que reflete uma melhor precisão e calibração na estimativa dos quantis. Além disso, o RQRN1E destacou-se pela eficiência computacional, com menor tempo de processamento e rápida convergência, sem comprometer a qualidade das previsões.

Palavras-chave: Calibração, Multitarefa, Redes Neurais, Regressão Gaussiana, Regressão Quantílica, Perda Quantílica.

Abstract

Quantile regression via neural networks has emerged as a powerful approach for modeling nonlinear relationships between variables, allowing for the individual estimation of conditional quantiles. However, this technique faces challenges when estimating multiple quantiles simultaneously. In this work, we propose a new architecture for Multitask Quantile Regression via Deep Neural Networks, aiming to improve both accuracy and computational efficiency in quantile estimation. Based on the architecture of Kuleshov e Deshpande (2022), this new architecture, called RQRN1E, introduces innovations by unifying the two stages of the original architecture into a single stage and incorporating the logit of τ directly into the penultimate intermediate layer of the network. In the three datasets evaluated, the results showed that the RQRN1E model outperformed competing models, consistently achieving the lowest quantile loss and the best coverage for the quantiles, reflecting better accuracy and calibration in quantile estimation. Furthermore, RQRN1E excelled in computational efficiency, with reduced processing time and faster convergence, without compromising prediction quality.

Keywords: Calibration, Gaussian Regression, Multitask, Neural Networks, Quantile Loss, Quantile Regression.

Sumário

1	Introdução	1
2	Referencial Teórico	5
2.1	Regressão Quantílica	5
2.2	Redes Neurais Artificiais	10
2.3	Regressão Quantílica via Redes Neurais	19
3	Metodologia	28
3.1	Nova Arquitetura	28
3.2	Conjunto de Dados	31
3.2.1	Diamonds	31
3.2.2	Boston	31
3.2.3	Yacht Hydrodynamics	32
3.3	Métodos	32
3.3.1	Pré-processamento dos Dados	32
3.3.2	Padronização dos Dados	33
3.3.3	Divisão dos Conjuntos de Dados	33
3.3.4	Modelos Treinados	33
3.3.5	Métricas de Desempenho	34

4	Resultados	38
4.1	Aplicação no Banco Diamonds	38
4.2	Aplicação no Banco Boston	43
4.3	Aplicação no Banco Yacht Hydrodynamics	46
5	Conclusões	50
	Referências Bibliográficas	51

Lista de Tabelas

4.1	Desempenho dos modelos nos conjuntos de treino e validação no banco <i>Diamonds</i>	40
4.2	Desempenho dos modelos no conjunto de teste para o banco <i>Diamonds</i>	42
4.3	Desempenho dos modelos nos conjuntos de treino e validação no banco <i>Boston</i>	45
4.4	Desempenho dos modelos no conjunto de teste no banco <i>Boston</i>	45
4.5	Desempenho dos modelos nos conjuntos de treino e validação no banco <i>Yacht Hydrodynamics</i>	48
4.6	Desempenho dos modelos no conjunto de teste no banco <i>Yacht Hydrodynamics</i>	49

Lista de Figuras

2.1	Representação gráfica da função de perda pinball.	7
2.2	Representação do Neurônio Artificial segundo McCulloch e Pitts (1943).	12
2.3	Rede Neural Simples e Rede Neural Profunda.	14
2.4	Processo de estimação de pesos em Redes Neurais.	15
2.5	Regressão Linear Gaussiana via Rede Neural Simples.	20
2.6	Regressão Quantílica Linear via Rede Neural Simples.	21
2.7	Regressão Quantílica não Linear via Rede Neural Profunda.	22
2.8	Regressão Quantílica via Rede Neural para múltiplos quantis, proposta por Rodrigues e Pereira (2020).	23
2.9	Regressão Quantílica via Rede Neural, proposta por Si, Kuleshov e Bishop (2022).	24
2.10	Regressão Quantílica via Redes Neurais, proposta por Kuleshov e Deshpande (2022).	25
3.1	Nova arquitetura da Regressão Quantílica Multitarefa via Redes Neurais Profundas.	29
4.1	Distribuição da variável preço dos diamantes em dólares.	39
4.2	Distribuição do preço mediano das casas do banco <i>Boston</i>	44
4.3	Distribuição da variável resistência no banco <i>Yacht Hydrodynamics</i>	47

Abreviações e Siglas

DGE	Descida de Gradiente Estocástico
FQA	Fluxos Quantílicos Autorregressivos
MADECP	Mean Absolute Distance of Empirical Cumulative Probability
MAE	Mean Absolute Error
MSE	Mean Squared Error
MQO	Mínimos Quadrados Ordinários
ReLU	Unidade Linear Retificada
RFQ	Regressão de Fluxo Quantílico
RGRN	Regressão Gaussiana via Redes Neurais Profundas
RMSE	Root Mean Square Error
RNA	Rede Neural Artificial
RNC	Redes Neurais Convolucionais
RNP	Rede Neural Profundas
RNR	Redes Neurais Recorrentes
RQ	Regressão Quantílica
RQRN	Regressão Quantílica via Redes Neurais
RQRN1E	Regressão Quantílica Multitarefa via Redes Neurais Profundas em um Estágio
RQRN2E	Regressão Quantílica Multitarefa via Redes Neurais Profundas em dois Estágios

Lista de Símbolos e Notações

$\mathbb{1}(\cdot)$	função indicadora, que retorna 1 se a condição for verdadeira e 0 caso contrário
τ	nível do quantil a ser estimado na regressão quantílica
n	tamanho da amostra
m	número de covariáveis
d	número de quantis estimados
n_B	tamanho do lote (batch)
k	índice da observação y_k , varia de 1 a n
i	índice do quantil τ_i , varia de 1 a d
y_k	variável resposta referente a k -ésima observação
\hat{y}_k	valor previsto da variável resposta da k -ésima observação
\mathbf{x}_k	vetor de covariáveis para a k -ésima observação
$\hat{q}_k(\tau)$	valor estimado para o τ -ésimo quantil da k -ésima observação.

Capítulo 1

Introdução

A utilização de redes neurais para a estimação de diversos quantis de uma distribuição condicional da variável resposta Y dado X é uma técnica promissora de modelagem de dados. Nessa abordagem, as redes neurais são treinadas não apenas para prever a média condicional, mas também quantis individuais, proporcionando uma modelagem mais abrangente da distribuição condicional da variável dependente. Além disso, as redes neurais possibilitam incorporar maior flexibilidade na modelagem da relação entre a variável dependente e as variáveis independentes, consolidando-se como uma abordagem robusta e eficaz na análise de dados complexos.

Aplicações de regressão quantílica via redes neurais na modelagem de dados vem sendo cada vez mais utilizada em diversas áreas. Na área de finanças, destacam-se os trabalhos de Taylor (2000), Pradeepkumar e Ravi (2017) e Gan et al. (2018), enquanto na saúde, a pesquisa de Cannon (2011) é notável. No campo da educação, o estudo de Pan, Huang e Chiu (2016) ressalta a utilidade dessa abordagem. As ciências ambientais também se beneficiaram com a contribuição de Cannon (2018), e na engenharia elétrica, o trabalho de Zhang, Quan e Srinivasan (2019) representa um avanço significativo. Além dessas áreas, a regressão quantílica via redes neurais encontrou aplicações em outros campos, como evidenciado por Xu, Deng, Jiang et al. (2017), Srivastava et al. (2014), Jantre, Bhattacharya e Maiti (2021) e Hao et al. (2023), demonstrando sua ampla aplicabilidade e interesse crescente na comunidade científica.

Este trabalho propõe o estudo de uma nova arquitetura de Regressão Quantílica Multitarefa via Redes Neurais Profundas, inspirada nos avanços de Kuleshov e Deshpande (2022). A relevância dessa abordagem está na sua capacidade de estimar simultaneamente múltiplos quantis de forma flexível, proporcionando múltiplas saídas. Assim, as informações são compartilhadas entre todos os quantis e espera-se que o problema de cruzamento seja empiricamente resolvido (Rodrigues e Pereira, 2020). Esta abordagem é enriquecida pelo aprendizado multitarefa, que visa aprender diversas tarefas simultaneamente, aplicando uma representação compartilhada. O espaço de características aprendido para cada tarefa é utilizado para aprimorar o desempenho das demais, seguindo um paradigma comprovadamente bem-sucedido em diversas aplicações de aprendizado de máquina, como processamento de linguagem natural, reconhecimento de fala e visão computacional (Faustine et al., 2022). Ao prever não apenas a média condicional, mas também uma variedade de quantis, o modelo oferece visões valiosas sobre a variabilidade dos resultados, o que é crucial para contextos onde a análise da incerteza e uma compreensão detalhada dos diferentes percentis da distribuição são fundamentais. Desta forma, a Regressão Quantílica Multitarefa via Redes Neurais Profundas destaca-se como uma ferramenta importante para a modelagem robusta e abrangente de dados complexos, especialmente onde a variabilidade das respostas desempenha um papel fundamental na análise completa.

Nos últimos anos, alguns modelos de Regressão Quantílica Multitarefa via Redes Neurais Profundas foram desenvolvidos e aperfeiçoados. Um estudo relevante foi conduzido por Rodrigues e Pereira (2020) no artigo intitulado "Beyond Expectation: Deep Joint Mean and Quantile Regression for Spatio-Temporal Problems". Nesse trabalho, os autores propuseram uma abordagem de aprendizado profundo multi-saída e multi-quantil para modelar simultaneamente vários quantis condicionais, juntamente com a média condicional. Essa estratégia visa oferecer uma visão mais abrangente da densidade preditiva em problemas espaço-temporais. Os resultados empíricos demonstraram a eficácia da abordagem ao resolver o problema de cruzamento de quantis, proporcionando uma descrição mais detalhada da densidade preditiva. Isso resultou em previsões aprimoradas da média condicional devido à inclusão de informações adicionais e

um efeito de regularização induzido pelos quantis adicionados.

Outro estudo relevante foi conduzido por Si, Kuleshov e Bishop (2022) no artigo "Auto-regressive Quantile Flows for Predictive Uncertainty Estimation". Sua abordagem propôs os Fluxos Quantílicos Autorregressivos (FQA), uma classe flexível de modelos de fluxo normalizado treinados com um objetivo baseado em regras de pontuação apropriadas. Os autores introduziram a Regressão de Fluxo Quantílico (RFQ), que parametriza distribuições condicionais preditivas com fluxos, resultando em previsões probabilísticas aprimoradas em diversos contextos.

Por último, Kuleshov e Deshpande (2022), no artigo "Calibrated and Sharp Uncertainties in Deep Learning via Density Estimation", enfatizaram a importância da calibração na prática. A calibração se refere à precisão com que as probabilidades previstas pelo modelo refletem as probabilidades reais dos eventos. Os autores aperfeiçoaram o trabalho de Si, Kuleshov e Bishop (2022), buscando a estimação de baixa dimensão das distribuições condicionais preditivas. Os autores descrevem esse procedimento como fácil de treinar e fundamentado em recalibração, gerando modelos calibrados sem comprometer o desempenho geral. Os resultados indicam melhorias de desempenho empírico em modelos Bayesianos lineares e profundos, sugerindo que a calibração deve ser cada vez mais incorporada ao aprendizado de máquina.

Inicialmente, Si, Kuleshov e Bishop (2022) propuseram uma arquitetura baseada em Monte Carlo, mas posteriormente, Kuleshov e Deshpande (2022) identificaram sua ineficiência em dimensões moderadas a altas do espaço de preditores. Eles apresentaram uma abordagem de incertezas calibradas e nítidas em aprendizagem profunda, envolvendo dois estágios. No primeiro, a rede neural gera um conjunto de características ou estatísticas resumo pré-definidas pelo pesquisador, e no segundo estágio, a rede neural auxiliar recebe essa representação como entrada, assim como a informação sobre os níveis τ dos quantis produzindo previsões calibradas.

Apesar da proposta interessante de Kuleshov e Deshpande (2022), ela possui limitações, especialmente a necessidade de treinamento em dois estágios, conforme descrito anteriormente,

e a necessidade de especificação de medidas resumo.

Neste trabalho, estudamos uma nova arquitetura de aprendizado profundo multitarefa baseada nas inovações de Si, Kuleshov e Bishop (2022) e Kuleshov e Deshpande (2022). A nova arquitetura propõe a unificação do processo de ajuste em um único estágio com a inclusão do parâmetro τ (níveis dos quantis) em uma das camadas intermediárias finais, mantendo o modelo mais robusto para espaços de preditores de alta dimensão. Essa modificação busca simplificar o treinamento e aumentar a eficácia do modelo proposto, proporcionando uma alternativa mais eficiente e prática. Um trabalho correlato, envolvendo Redes Neurais com Grafos e alterações semelhantes no modelo, foi desenvolvido concomitantemente na dissertação de Mestrado de Amorim (2024).

Esta dissertação está estruturada da seguinte forma: o Capítulo 2 (Referencial Teórico) detalha o referencial teórico, iniciando com os modelos de regressão quantílica e de redes neurais de forma independente. Ainda no Capítulo 2, discute-se a Regressão Quantílica via Redes Neurais, incluindo sua utilidade, histórico, substituição da função de perda MSE pela perda pinball, e a estimação independente dos quantis. As contribuições de Rodrigues e Pereira (2020), Si, Kuleshov e Bishop (2022), e a inovação de estimação em dois estágios de Kuleshov e Deshpande (2022) são também detalhadas no Capítulo 2. O Capítulo 3 (Metodologia) apresenta a nova arquitetura, denominada Regressão Quantílica Multitarefa via Redes Neurais em um Estágio, focando na escolha e descrição dos conjuntos de dados, nas variáveis envolvidas e nos modelos testados. Além disso, aborda o processo de treinamento, ajuste de hiperparâmetros e as métricas de desempenho, como RMSE, MAE, MADECP, Perda Quantílica e Cobertura. No Capítulo 4 (Resultados), são apresentados os resultados obtidos, seguidos de uma análise comparativa entre os modelos utilizando as métricas definidas. Por fim, o Capítulo 5 (Conclusões) sintetiza os principais achados, destacando a eficácia do melhor modelo e sugerindo possíveis melhorias para trabalhos futuros.

Capítulo 2

Referencial Teórico

Nesse capítulo, será apresentada uma revisão detalhada dos principais conceitos e técnicas que fundamentam este estudo. Primeiramente, abordaremos a Regressão Quantílica (RQ), uma metodologia estatística fundamental para a estimação de quantis condicionais. Em seguida, exploraremos as Redes Neurais Artificiais (RNA), descrevendo os elementos básicos do neurônio artificial, seu funcionamento, e a evolução para as Redes Neurais Profundas (RNP). Além disso, discutiremos os processos de estimação dos pesos e a importância dos hiperparâmetros em uma rede neural. Por fim, será introduzida a Regressão Quantílica via Redes Neurais (RQRN), que combina as técnicas de regressão quantílica com a flexibilidade das redes neurais, proporcionando uma visão abrangente das metodologias que embasam a pesquisa.

2.1 Regressão Quantílica

Os modelos tradicionais de regressão, embora amplamente utilizados por apresentarem uma interpretação simples, enfrentam limitações quando os erros do modelo não seguem uma distribuição Normal. A abordagem comumente adotada, como a minimização dos quadrados dos erros (Mínimos Quadrados Ordinários - MQO), também é vulnerável a outliers, exigindo uma avaliação minuciosa de sua influência no ajuste do modelo (Rousseeuw e Leroy, [2003](#)).

A Regressão Quantílica, introduzida por Koenker e Bassett (1978), surge como uma alternativa robusta e informativa. Ao contrário da regressão linear tradicional, que resume as variáveis independentes às suas médias condicionais $E(Y|X)$, a Regressão Quantílica estima diferentes quantis $Q_\tau(Y|X)$ da distribuição condicional da variável resposta Y . Essa capacidade de avaliar o impacto das covariáveis em toda a distribuição, não apenas na média, oferece uma compreensão mais abrangente e realista do comportamento do modelo, sendo particularmente eficaz na presença de assimetria e outliers (Koenker, 2005).

O modelo linear de regressão quantílica, introduzido por Koenker e Bassett (1978), pode ser definido como:

$$Q_{Y_k}(\tau|X = \mathbf{x}_k) = \mathbf{x}_k^T \boldsymbol{\beta}(\tau), \quad (2.1)$$

onde $\boldsymbol{\beta}(\tau)$ é o vetor de parâmetros e \mathbf{x}_k é o vetor de preditores. A estimação dos quantis condicionais na Regressão Quantílica é conduzida através da minimização de uma função de perda fundamental conhecida como "Pinball". Esta função de perda, denotada por $\rho_\tau(r)$, utiliza os resíduos ponderados pelo nível do quantil desejado,

$$\rho_\tau(r) = \begin{cases} \tau r, & \text{se } r \geq 0, \\ (\tau - 1)r, & \text{se } r < 0. \end{cases} \quad (2.2)$$

Aqui, o τ representa o nível do quantil desejado, e r denota o resíduo. Os resíduos são calculados como a diferença entre o valor observado (y_k) e a previsão do quantil condicional $\mathbf{x}_k^T \boldsymbol{\beta}(\tau)$ para um dado quantil τ .

Essa função de perda Pinball possui uma propriedade única: ela penaliza de maneira assimétrica os desvios positivos e negativos.

- Se o resíduo for positivo ($r \geq 0$), a penalização é proporcional a τ vezes o resíduo;
- Se o resíduo for negativo ($r < 0$), a penalização é proporcional a $(\tau - 1)$ vezes o resíduo.

Dessa maneira, a minimização retorna uma estimativa do quantil $Q_\tau(Y|X)$, considerando a função de perda assimétrica associada ao quantil τ . A Figura 2.1 ilustra a função de perda.

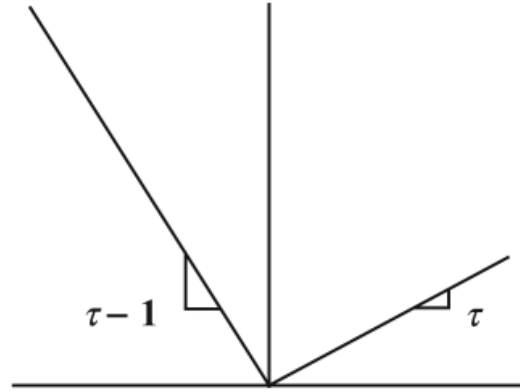


Figura 2.1: Representação gráfica da função de perda pinball.
Fonte: Rodrigues e Pereira (2020).

A perda total para o conjunto de dados é calculada encontrando primeiro a perda individual para cada observação e, em seguida, tomando a média dessas perdas:

$$J_\tau = \frac{1}{n} \sum_{k=1}^n \rho_\tau(r_k), \quad (2.3)$$

onde a estimativa do parâmetro $\hat{\beta}(\tau)$ para um dado quantil τ pode ser obtida resolvendo o seguinte problema de otimização por meio de programação linear (Koenker, 2005):

$$\hat{\beta}(\tau) = \arg \min_{\beta(\tau) \in \mathbb{R}^p} \left\{ \sum_{k=1}^n \rho_\tau(y_k - \hat{q}(\tau | \mathbf{x}_k)) \right\}, \quad (2.4)$$

onde $\hat{q}(\tau | x_k)$ é o quantil previsto para o quantil τ , dado o vetor de covariáveis \mathbf{x}_k na k -ésima observação.

O problema de minimização indicado em (2.4) deve ser repetido para cada quantil τ de forma independente. Assim, os quantis são estimados separadamente utilizando a função de perda pinball para ponderar os resíduos, resultando em uma modelagem adaptativa e informativa de diferentes partes da distribuição condicional.

Para a construção de intervalos de confiança para o parâmetro $\hat{\beta}$, serão apresentados dois métodos: (i) baseados em resultados assintóticos e (ii) *bootstrap*. Os métodos assintóticos permitem avaliar casos onde os erros sejam independentes e identicamente distribuídos ou independentes e não identicamente distribuídos, enquanto o método *bootstrap* é destacado por não requerer suposições sobre a distribuição dos erros (Efron e Tibshirani, 1994).

Método baseado em resultados assintóticos

O método baseado em resultados assintóticos para a construção de intervalos de confiança dos parâmetros da regressão quantílica foi estabelecido por (Koenker e Bassett, 1978). Estes métodos, baseiam-se na distribuição assintótica dos estimadores dos parâmetros, obtidos pela minimização da soma dos erros absolutos ponderados. Sob condições de regularidade, tem-se que a distribuição assintótica dos estimadores de parâmetros se aproxima de uma distribuição normal multivariada:

$$\sqrt{n} \begin{pmatrix} \hat{\beta}(\tau_1) - \beta(\tau_1) \\ \hat{\beta}(\tau_2) - \beta(\tau_2) \\ \vdots \\ \hat{\beta}(\tau_d) - \beta(\tau_d) \end{pmatrix} \xrightarrow{D} N_{d \times m}(0, V(\tau_1, \tau_2, \dots, \tau_d)), \quad (2.5)$$

onde $V(\tau_1, \tau_2, \dots, \tau_d)$ é a matriz de covariâncias.

Koenker e Bassett (1978) também introduziram expressões para a matriz de covariâncias assintóticas $V(\tau)$ para um quantil específico τ .

- No caso em que os erros são independentes e não identicamente distribuídos (inid):

$$V(\tau) = \tau(1 - \tau)(X'FX)^{-1}(X'X)(X'FX)^{-1}. \quad (2.6)$$

- Onde os erros são independentes e identicamente distribuídos (iid):

$$V(\tau) = \frac{\tau(1 - \tau)}{f^2(0)}(X'X)^{-1}. \quad (2.7)$$

A matriz F é diagonal, com elementos $f_i(0)$, representando a função de densidade dos erros. Com base nessas formulações para $V(\tau)$, é possível construir intervalos de confiança para os parâmetros do modelo $\beta(\tau)$.

Método *bootstrap*

O método *bootstrap* é uma técnica de reamostragem utilizada extensivamente na inferência estatística, particularmente valiosa para a estimativa de parâmetros em modelos de regressão, que inclui a regressão quantílica. Esta técnica foi proposta inicialmente por Efron e Tibshirani (1993), com o objetivo de estimar a matriz de covariâncias dos parâmetros dos modelos. Posteriormente, Koenker (2005) adaptou o método *bootstrap* especificamente para a regressão quantílica, ampliando assim sua aplicabilidade.

O procedimento *bootstrap* envolve a seleção aleatória de pares de observações, cada um com igual probabilidade, para formar novos conjuntos de dados. Esses novos conjuntos, que consistem em vetores de variável resposta e matrizes de planejamento atualizadas, são utilizados em cada uma das R reamostragens (Efron e Tibshirani, 1993). Através dessas reamostragens, calcula-se um vetor de parâmetros estimados para cada uma, permitindo a estimativa do erro padrão baseada na dispersão dessas R estimativas. Esse processo é fundamental para a construção de intervalos de confiança robustos para os parâmetros estimados (He e Hu, 2002).

De maneira mais específica, um intervalo de confiança para o parâmetro $\beta_j(\tau)$, com um coeficiente de confiança $\gamma = 1 - \alpha$, pode ser expresso como:

$$\hat{\beta}_j(\tau) \pm z_{\alpha/2} \cdot \text{E.P.} \left(\hat{\beta}_j(\tau) \right). \quad (2.8)$$

Nesta expressão, $z_{\alpha/2}$ representa o quantil de ordem $1 - \alpha/2$ da distribuição Normal padrão, e $\text{E.P.} \left(\hat{\beta}_j(\tau) \right)$ é o estimador do erro padrão do estimador do parâmetro $\beta_j(\tau)$, obtido através do procedimento *bootstrap*. Este método não apenas facilita a estimação de parâmetros em contextos complexos, mas também contribui significativamente para a avaliação da precisão

dessas estimativas (Efron e Tibshirani, 1993; Koenker, 2005).

2.2 Redes Neurais Artificiais

As Redes Neurais Artificiais (RNA) têm se destacado nos últimos anos como uma opção atrativa para lidar com problemas complexos em variados campos. Essa tendência é impulsionada pela crescente capacidade de processamento computacional, tornando as RNAs ferramentas valiosas para modelagem matemática e simulação de cenários futuros em diversas aplicações. Conforme Oleskovicz, Coury e Aggarwal (2003) indicam, o progresso na inteligência artificial revela que as RNAs se apresentam como alternativas robustas aos métodos convencionais, baseados em abordagens paramétricas, que são mais limitadas e rígidas.

Em contraste com os métodos tradicionais, que dependem de regras fixas para solucionar problemas, as RNAs oferecem uma abordagem mais dinâmica e adaptativa. Elas são capazes de aprender diretamente dos dados, identificando padrões e ajustando-se a novas informações sem a necessidade de programação explícita (Flood e Kartam, 1994). Por exemplo, Redes Neurais Convolucionais (RNC) são amplamente utilizadas no reconhecimento de imagens, superando técnicas convencionais ao aprender características relevantes automaticamente (Vogado et al., 2019). Da mesma forma, Bengio, Simard e Frasconi (1994) afirmam que as Redes Neurais Recorrentes são eficazes para análise de sequências de dados, como linguagem natural, demonstrando superioridade em tarefas de previsão e tradução quando comparadas a métodos que se apoiam em regras gramaticais estritas.

Além disso, os Autoencoders são outro exemplo de RNAs que se destacam, especialmente em tarefas de redução de dimensionalidade e detecção de anomalias, ao aprender representações compactas dos dados de maneira não supervisionada, algo que abordagens tradicionais, como a Análise de Componentes Principais, não conseguem fazer tão eficientemente (Şengönül et al., 2023). A origem das RNAs remonta a 1943, com o modelo matemático do neurônio biológico proposto por (McCulloch e Pitts, 1943). Desde então, essas redes evoluíram significativamente,

desempenhando um papel vital em abordagens inovadoras para modelagem de dados e tomada de decisões.

As RNAs são amplamente empregadas na solução de problemas complexos, especialmente quando o comportamento das variáveis não é facilmente descrito. Sua característica fundamental é a capacidade de aprendizado por meio de exemplos e a habilidade de generalizar informações, resultando na criação de modelos não-lineares (Spörl, Castro e Luchiari, 2011). A aplicabilidade das RNAs é notável em diversos domínios, como visão computacional, processamento de linguagem natural, reconhecimento de padrões e jogos, devido à sua capacidade única de aprender representações complexas e não-lineares (Haykin, 2001).

Neurônio Artificial: Elementos e Funcionamento

Um neurônio artificial constitui a unidade básica de processamento dentro de uma rede neural, desempenhando um papel vital na modelagem e interpretação de dados complexos. Inspirado na estrutura e funcionamento dos neurônios biológicos, o neurônio artificial foi primeiramente conceituado pelo modelo de McCulloch e Pitts (1943), ilustrado na Figura 2.2. Este modelo captura a essência do processamento de informações através de elementos-chave descritos a seguir:

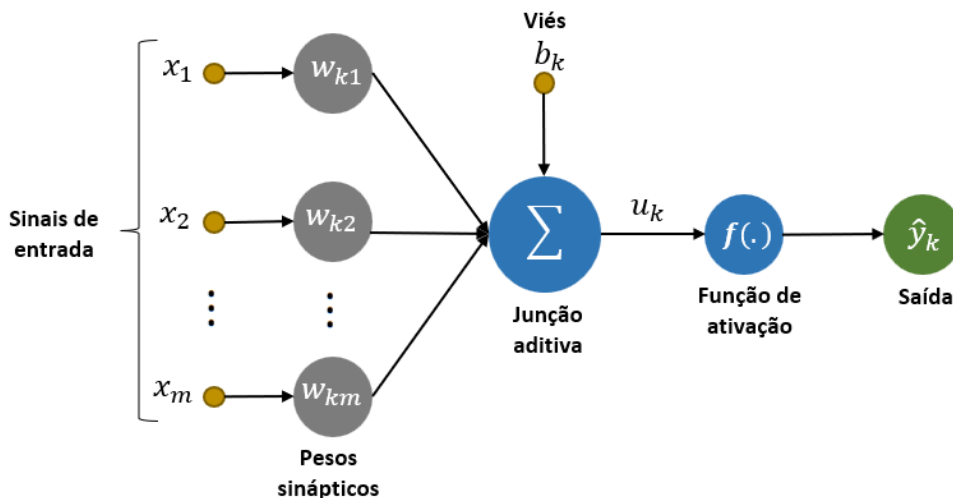


Figura 2.2: Representação do Neurônio Artificial segundo McCulloch e Pitts (1943).

Fonte: Elaboração própria.

- **Entradas e Pesos:** Cada neurônio k recebe múltiplas entradas (x_1, x_2, \dots, x_m) , onde cada entrada x_j é associada a um peso específico $(W_{k1}, W_{k2}, \dots, W_{km})$. Os pesos representam a força da conexão entre os neurônios e são ajustáveis durante o treinamento da rede.
- **Junção de Soma (Σ):** Este mecanismo agrega todas as entradas ponderadas, somando-as juntamente com um viés b_k , resultando na entrada líquida do neurônio. O viés permite ajustar o ponto de ativação do neurônio, oferecendo flexibilidade adicional ao modelo. Matematicamente, isso é expresso como:

$$u_k = \sum_{j=1}^m W_{kj}x_j + b_k. \quad (2.9)$$

- **Função de Ativação $f(\cdot)$:** A entrada líquida u_k é então processada por uma função de ativação $f(\cdot)$, que modula a saída do neurônio. A função de ativação introduz não-linearidade ao modelo, permitindo que a rede neural aprenda e modele relações complexas entre as entradas e as saídas. Exemplos comuns de funções de ativação incluem

a identidade, sigmoide (σ), a tangente hiperbólica (\tanh) e a unidade linear retificada (ReLU). Para mais detalhes, ver (Géron, 2019).

$$f(u_k) = f\left(\sum_{j=1}^m W_{kj}x_j + b_k\right). \quad (2.10)$$

Onde a saída do neurônio, \hat{y}_k , é dada pelo resultado dado em (2.10):

$$\hat{y}_k = f(u_k). \quad (2.11)$$

Juntos, esses componentes formam a base do funcionamento de um neurônio artificial, permitindo a implementação de redes neurais capazes de aprender e adaptar-se a uma vasta gama de tarefas computacionais, desde reconhecimento de padrões até decisões complexas baseadas em dados.

Redes Neurais Profundas

As Redes Neurais Profundas (RNP) representam um avanço significativo na modelagem de relações complexas e abstratas nos dados, estendendo-se além do conceito de um único neurônio para incorporar múltiplas camadas de neurônios (DSA, 2022). Essas camadas estão organizadas de tal forma que a saída de uma camada serve como entrada para a próxima, criando uma estrutura em cascata. As camadas situadas entre a entrada e a saída, conhecidas como camadas ocultas ou intermediárias, desempenham um papel crucial na captura de representações hierárquicas dos dados. Esta estrutura é exemplificada na comparação entre uma rede neural simples e uma rede neural profunda, conforme apresentado na Figura 2.3.

O campo da aprendizagem profunda, que se baseia na estrutura e funcionamento das redes neurais profundas, tem sido fundamental para impulsionar progressos notáveis em diversas áreas da tecnologia. Entre essas áreas, destacam-se a visão computacional, o reconhecimento de fala, o processamento de linguagem natural e o reconhecimento de áudio. O sucesso da

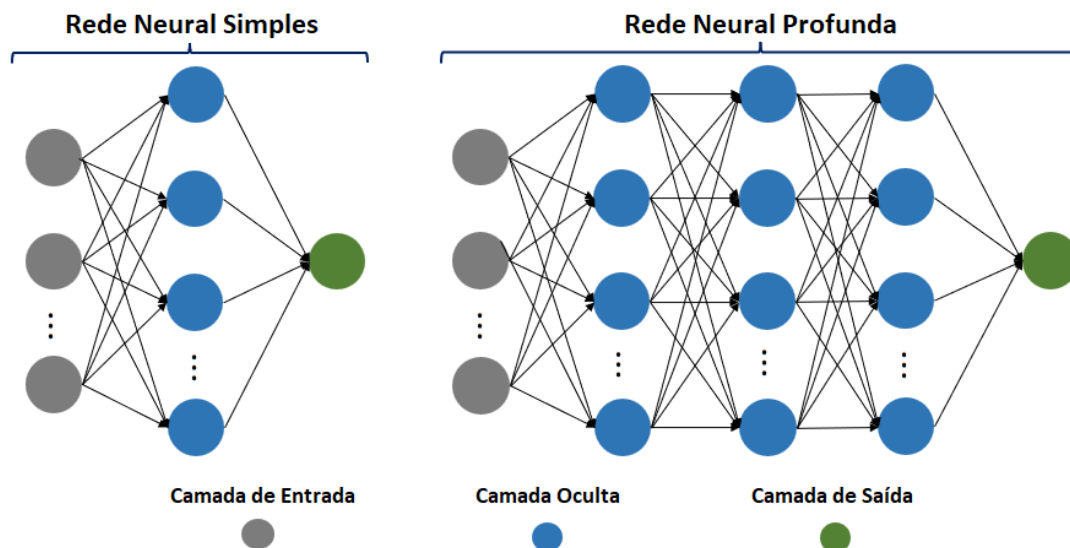


Figura 2.3: Rede Neural Simples e Rede Neural Profunda.

Fonte: Elaboração própria.

aprendizagem profunda advém da sua capacidade de imitar a maneira complexa e eficiente com que o cérebro humano processa informações (DSA, 2022).

Estimação dos Pesos

A estimativa dos pesos em uma rede neural é uma etapa crucial realizada através do algoritmo gradiente descendente com *backpropagation*, um método essencial no treinamento de modelos de aprendizado profundo. Durante o treinamento, o modelo utiliza uma função de perda para comparar suas previsões com os valores reais e quantificar o erro (DSA, 2022). O *backpropagation* propaga esse erro de volta pelas camadas da rede, permitindo o ajuste iterativo dos pesos para minimizar a função de perda. O gradiente descendente é um algoritmo de otimização, é crucial nesse processo, pois ajuda a encontrar o conjunto de pesos que minimiza a função de perda ao ajustar os pesos na direção oposta ao gradiente da função. Assim, *backpropagation* e gradiente descendente trabalham em conjunto para aprimorar as previsões do modelo, reduzindo progressivamente o erro durante o treinamento (Lawrence, 1997). O mesmo é ilustrado na Figura 2.4.

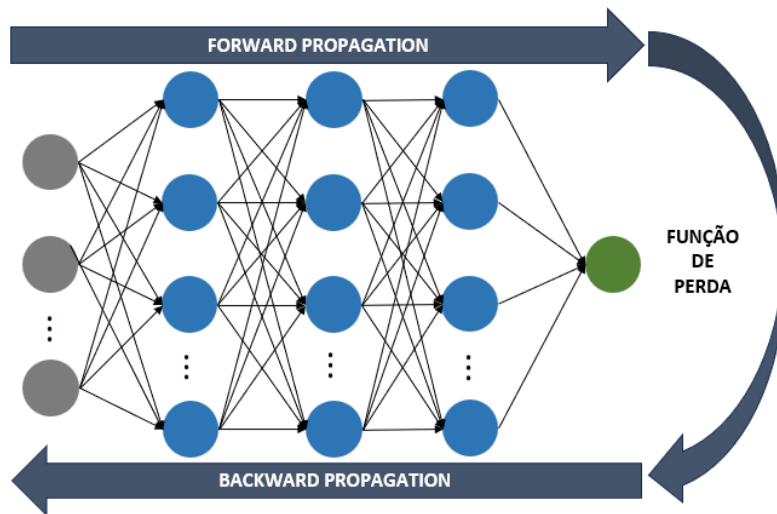


Figura 2.4: Processo de estimação de pesos em Redes Neurais.

Fonte: Elaboração própria.

- **Forward Propagation**

Nesta fase, as entradas são introduzidas na rede neural, e a informação é propagada camada por camada até a camada de saída. A saída de cada neurônio é calculada usando a função de ativação, representada na Equação 2.11.

- **Função de Perda**

A diferença entre as previsões da rede e os valores reais é calculada através da função de perda. Essa escolha é crucial no desenvolvimento do modelo, pois guia o processo de aprendizado ao medir o quão bem o modelo se ajusta aos dados. A seleção da função de perda depende do tipo de problema (regressão ou classificação), das características dos dados e dos objetivos do modelo (Goodfellow, Bengio e Courville, 2017). Abaixo, serão exploradas quatro funções de perda comuns e suas principais aplicações, proporcionando uma compreensão mais clara de suas utilidades e implicações.

A log-verossimilhança é frequentemente usada em modelos que parametrizam a distribuição dos dados. A maximização da log-verossimilhança (ou, equivalentemente, a minimização do seu valor negativo) permite estimar os parâmetros do modelo de forma que a probabilidade

observada dos dados sob o modelo seja maximizada (Tibshirani et al., 2017). A equação geral da log verossimilhança negativa para um modelo de probabilidade é:

$$J^{(1)} = - \sum_{k=1}^n \log P(\mathbf{y}_k | \mathbf{x}_k; \boldsymbol{\theta}), \quad (2.12)$$

onde $P(\mathbf{y}_k | \mathbf{x}_k; \boldsymbol{\theta})$ representa a probabilidade do valor real \mathbf{y}_k dado o vetor de características \mathbf{x}_k e os parâmetros do modelo $\boldsymbol{\theta}$.

Em tarefas de classificação, a entropia cruzada é a função de perda mais comum, usada para medir a diferença entre duas distribuições de probabilidade: a prevista pelo modelo $p(y|x)$ e a real observada nos dados $q(y|x)$ (Géron, 2019). Para classificação binária, a entropia cruzada é expressa por:

$$J^{(2)} = -\frac{1}{n} \sum_{k=1}^n [y_k \log(\hat{y}_k) + (1 - y_k) \log(1 - \hat{y}_k)], \quad (2.13)$$

onde y_k é o valor real (0 ou 1) e \hat{y}_k é a probabilidade prevista pelo modelo de a amostra pertencer à classe 1.

Quando o foco está na regressão para a média, o Mean Squared Error (MSE) é frequentemente escolhido como função de perda devido à sua eficácia em medir a média dos quadrados das diferenças entre os valores previstos e os valores reais.

$$MSE = J_1 = \frac{1}{n} \sum_{k=1}^n (y_k - \hat{y}_k)^2. \quad (2.14)$$

Essa função de perda é amplamente empregada para medir a diferença entre os valores observados (y_k) e os valores previstos (\hat{y}_k), tornando-a uma escolha robusta para muitos problemas de regressão (Hastie, Tibshirani e Friedman, 2009). Vale ressaltar que utilizar a função de perda da log verossimilhança assumindo Normalidade é equivalente a utilizar a perda MSE.

Por último, a regressão quantílica, que tem como objetivo prever um quantil específico da distribuição condicional de uma variável dependente, utiliza a função de perda Pinball:

$$J_\tau = \frac{1}{n} \sum_{k=1}^n \rho_\tau (y_k - \hat{q}(\tau|x_k)). \quad (2.15)$$

Esta função é projetada para penalizar os resíduos de forma assimétrica, dependendo do quantil alvo τ , tornando-a especialmente útil para modelos que precisam estimar valores em uma distribuição condicional que não é simétrica (Koenker, 2005).

A compreensão dos objetivos do modelo e das características da variável resposta é fundamental para a escolha adequada da função de perda, que, por sua vez, impacta diretamente o desempenho do modelo de aprendizado de máquina (Goodfellow, Bengio e Courville, 2017). Cada função de perda é projetada para diferentes tipos de problemas. Assim, uma escolha criteriosa da função de perda não apenas orienta o aprendizado do modelo, mas também influencia sua capacidade de generalização em dados novos, sendo crucial para o sucesso da modelagem.

- ***Backward Propagation***

Esta etapa calcula os gradientes da função de perda em relação aos pesos, retrocedendo camada por camada desde a saída até as camadas de entrada. Utiliza a regra da cadeia do cálculo para obter os gradientes das funções encadeadas.

Os pesos são ajustados na direção oposta ao gradiente, utilizando a regra de atualização:

$$W_{kj} = W_{kj} - \alpha \frac{\partial J(y, \hat{y})}{\partial W_{kj}}, \quad (2.16)$$

onde α é a taxa de aprendizado, detalhada na seção a seguir.

Este ciclo de propagação para frente e retropropagação do erro é repetido ao longo de várias iterações (épocas) até que a rede neural aprenda a realizar previsões precisas, ajustando seus pesos de maneira apropriada (Goh, 1995).

Hiperparâmetros de uma Rede Neural

Os hiperparâmetros das redes neurais desempenham um papel crucial na configuração e no desempenho dos modelos de aprendizado profundo. Eles são ajustados antes do treinamento e permanecem constantes ao longo do processo, diferentemente dos parâmetros (pesos) do modelo, que são aprendidos automaticamente. Os hiperparâmetros podem ser otimizados na etapa de validação, com o conjunto de dados de treinamento. Vamos explorar os tipos mais comuns de hiperparâmetros e suas definições no contexto das redes neurais (Géron, 2019; Goodfellow, Bengio e Courville, 2017; Srivastava et al., 2014):

- **Número de Camadas:** Determina a profundidade da rede neural. Redes mais profundas podem aprender padrões mais complexos, mas também são mais suscetíveis ao *overfitting* e exigem mais dados para treinamento eficaz. A chave é encontrar um equilíbrio que ofereça um bom desempenho sem tornar o modelo desnecessariamente complexo.
- **Número de Neurônios:** Define a largura das camadas. O número de neurônios em cada camada influencia a capacidade da rede de capturar nuances nos dados. Muitos neurônios podem aumentar o risco de *overfitting*, enquanto poucos podem resultar em *underfitting*. A escolha ideal depende da complexidade dos dados e da tarefa.
- **Taxa de aprendizado:** Determina o tamanho dos passos no ajuste dos parâmetros (ou pesos) da rede durante o treinamento (ver Equação 2.16), influenciando a velocidade e a eficiência com que a rede neural converge para uma solução.
- **Função de ativação:** Essencial para limitar a saída do neurônio a um intervalo manejável, geralmente entre 0 e 1 ou -1 e 1. Essa restrição evita problemas como a explosão de gradientes e contribui para a estabilidade e eficiência do treinamento. Além de facilitar a captura de padrões complexos, a função de ativação permite que a rede aprenda tanto relações lineares quanto interações não lineares entre variáveis de entrada, impactando diretamente seu desempenho (ver Equação 2.10).

- **Tamanho do Lote (*Batch Size*):** Determina quantas amostras são processadas antes da atualização dos parâmetros da rede. As abordagens variam desde o modo batch, onde o lote engloba todo o conjunto de dados, até o modo estocástico, com lotes de tamanho um. O modo minibatch é um meio-termo, com lotes maiores que um, mas menores que o total do conjunto.
- **Número de épocas:** Indica o número de vezes que o algoritmo de treinamento trabalhará através do conjunto completo de dados, influenciando diretamente no ajuste dos pesos da rede.
- **Early Stopping:** Técnica para evitar *overfitting* interrompendo o treinamento quando a performance em um conjunto de validação para de melhorar. Depende da *paciência*, que define o número de épocas sem melhoria antes da parada.
- **Dropout:** Técnica para reduzir *overfitting* desativando aleatoriamente uma fração de neurônios durante o treinamento, controlada pela *taxa de dropout*. Isso força o modelo a aprender representações mais robustas, melhorando sua capacidade de generalização em dados novos.

2.3 Regressão Quantílica via Redes Neurais

A Regressão Quantílica via Redes Neurais (RQRN) é uma metodologia avançada que combina os pontos fortes da regressão quantílica com a capacidade de modelagem de padrões complexos das redes neurais. Esta abordagem é fundamental em diversos campos, desde a previsão financeira até a pesquisa médica, por sua habilidade única de fornecer uma visão detalhada da estrutura condicional de uma variável dependente Y em relação a um conjunto de variáveis independentes X , através de diferentes quantis da distribuição de Y .

Em sua essência, a RQRN estende a regressão quantílica para capturar relações não lineares e interações complexas entre as variáveis. A estrutura típica de uma RQRN inclui uma camada

de entrada que corresponde aos preditores X , seguida por uma ou várias camadas ocultas que transformam as entradas através de pesos ajustáveis, vieses, e funções de ativação não lineares, culminando em uma camada de saída que fornece a previsão para um quantil específico τ de Y (Cannon, 2011).

Apesar das redes neurais serem utilizadas buscando a modelagem de relações complexas e não lineares nos dados, os modelos mais simples de regressão linear gaussiana e de regressão quantílica também podem ser ajustados utilizando uma arquitetura básica de uma rede neural. A Figura 2.5 ilustra a abordagem da regressão linear gaussiana realizada por meio de uma rede neural simples, focando na estimação da média.

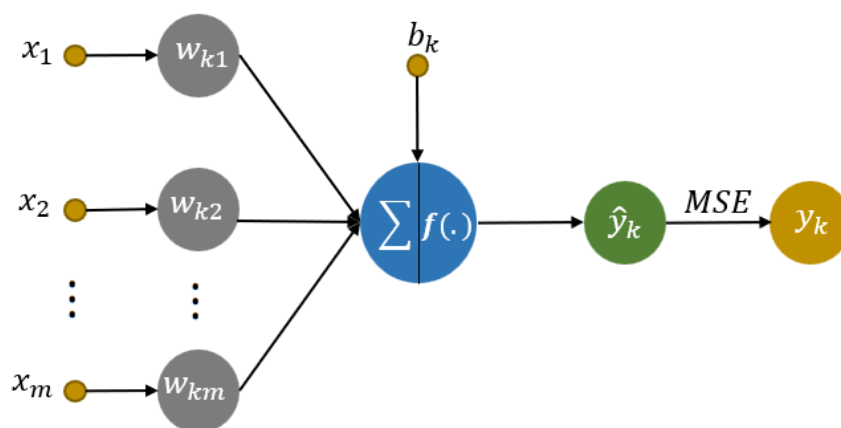


Figura 2.5: Regressão Linear Gaussiana via Rede Neural Simples.

Fonte: Elaboração própria.

Na Figura 2.5, para realização da regressão Gaussiana, foi usada uma rede com uma camada intermediária, uma função de ativação identidade e função de perda MSE. A função de ativação identidade, toma as entradas aplicadas a ela, multiplica-as pelos pesos atribuídos a cada neurônio na rede e produz uma saída que é diretamente proporcional à entrada (Gharat, 2019). Com isso, a fórmula para o valor estimado de \hat{y}_k pode ser apresentada como segue em (2.17):

$$\hat{y}_k = \sum_{j=1}^m W_{kj}x_j + b_k. \quad (2.17)$$

Neste contexto, a função de perda utilizada é dada pelo MSE, conforme a Equação 2.14. Assim, a rede busca minimizar o MSE com previsões lineares para variável resposta Y , ou seja, tem solução equivalente a uma regressão linear Gaussiana.

Ao abordarmos a Regressão Quantílica via Redes Neurais, precisamos alterar a função de perda utilizada, substituindo o MSE (2.14) pela função de perda Pinball Equação (2.4). Esta modificação é evidenciada na Figura 2.6, que apresenta uma regressão quantílica linear por meio de uma rede neural simples, focada em um quantil específico τ , empregando-se a função de ativação identidade.

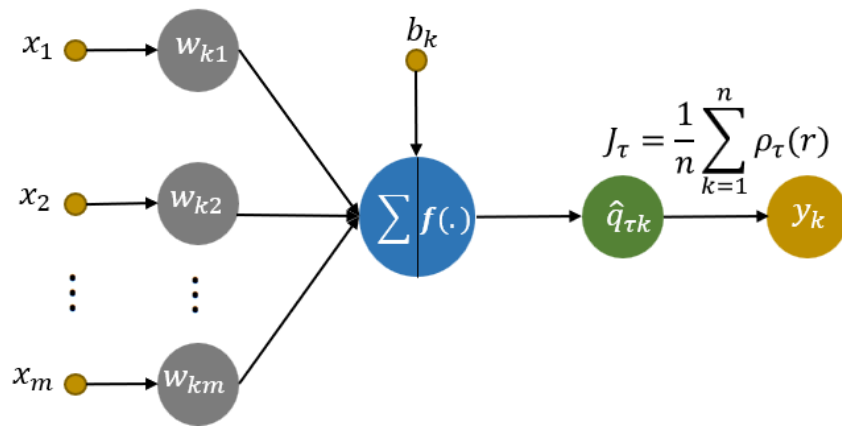


Figura 2.6: Regressão Quantílica Linear via Rede Neural Simples.

Fonte: Elaboração própria.

A evolução para estruturas mais complexas ou não lineares é explorada através do Aprendizado Profundo e funções de ativação não lineares, como representado na Figura 2.7. Aqui, é apresentada a Regressão Quantílica não linear via Rede Neural Profunda (RQRN), que emprega a mesma função de perda Pinball indicada anteriormente, mas com a adição de múltiplas camadas e funções de ativação diferentes da função identidade. Esta abordagem permite modelar relações mais complexas e capturar dinâmicas não lineares entre as variáveis. Entretanto, há a limitação da escolha de um quantil específico τ , ou seja, para a estimação de diversos quantis, redes neurais separadas precisam ser ajustadas.

Uma vantagem distintiva da RQRN é a sua habilidade para estimar de forma independente

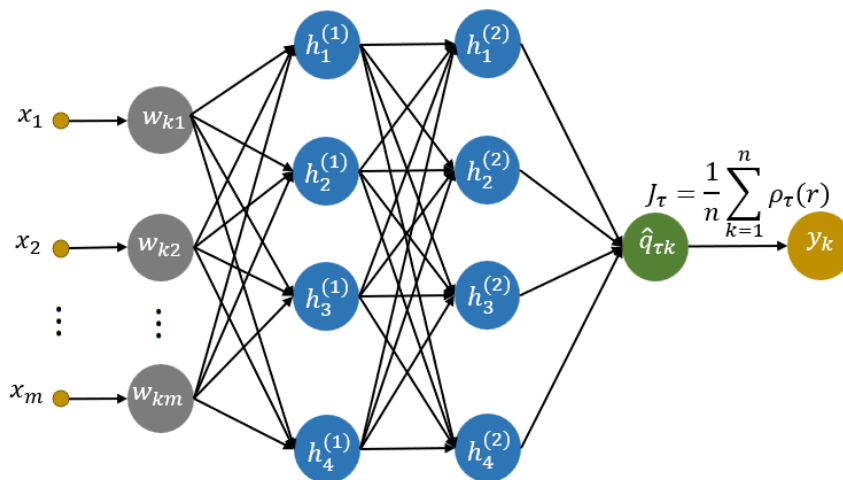


Figura 2.7: Regressão Quantílica não Linear via Rede Neural Profunda.
Fonte: Elaboração própria.

múltiplos quantis da distribuição condicional de Y , sem a necessidade de suposições prévias sobre a forma dessa distribuição (Moon et al., 2021). Isso é fundamental em cenários onde a distribuição de Y é desconhecida, assimétrica ou caracterizada por caudas pesadas. Diferentemente dos métodos tradicionais, que geralmente pressupõem uma relação linear entre as variáveis independentes X e a variável dependente Y , a RQRN é um método fundamental para entender a complexidade e a não linearidade dessas relações.

Neste contexto, Rodrigues e Pereira (2020) trouxe avanços na RQRN, que consistiram na modelagem não apenas a média, mas também múltiplos quantis de uma distribuição de forma simultânea. Esta metodologia se destaca ao oferecer uma visão mais completa da distribuição preditiva de uma variável dependente Y , condicionada a um conjunto de variáveis independentes X . Através de uma arquitetura multiquantil de saída múltipla, a proposta permite calcular $1 + d$ saídas, uma para a média e d saídas são destinadas aos quantis específicos $(\tau_1, \tau_2, \dots, \tau_d)$ de interesse. Esta capacidade de estimar simultaneamente a média e múltiplos quantis é uma evolução significativa em relação às abordagens apresentadas anteriormente, que normalmente se concentravam em um único aspecto da distribuição. A Figura 2.8 ilustra a metodologia proposta por Rodrigues e Pereira (2020) considerando a estimação simultânea de d quantis.

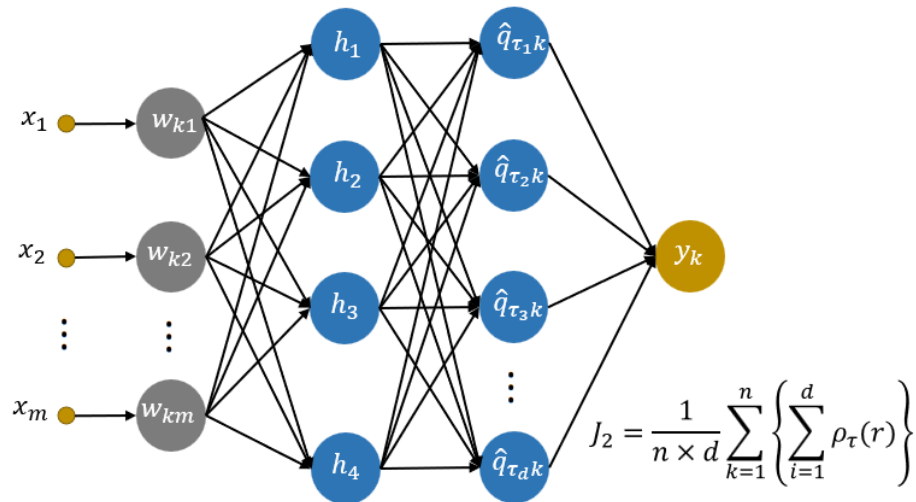


Figura 2.8: Regressão Quantílica via Rede Neural para múltiplos quantis, proposta por Rodrigues e Pereira (2020).

Fonte: Elaboração própria.

Para implementar essa abordagem, os autores utilizam uma função de perda (J_2) inspirada na Equação (2.15), composta pela média das funções de perda Pinball individuais para cada quantil τ_d de interesse.

Esta proposta de Rodrigues e Pereira (2020) expressa a concepção de multitarefa, destacando-a como uma abordagem interessante na execução simultânea de diversas atividades. Também representa um avanço significativo na aplicação de redes neurais para análises estatísticas, permitindo uma compreensão das relações entre variáveis. Ao estimar simultaneamente vários quantis, a abordagem fornece uma representação detalhada da distribuição condicional da variável dependente, superando limitações de modelos que focam em um único aspecto. O modelo proporciona ainda um compartilhamento de informações entre os quantis avaliados e espera-se que pelo menos empiricamente, o desafio associado ao cruzamento de quantis seja resolvido. Ainda assim, é importante reconhecer as limitações, como a necessidade de definir os quantis de interesse a priori.

Nesse contexto, a Teoria de Regressão Quantílica Multitarefa via Redes Neurais Profundas foi marcada por mais uma inovação proposta por Si, Kuleshov e Bishop (2022), ilustrada na

Figura 2.9. Esta metodologia traz avanços em comparação com a proposta de Rodrigues e Pereira (2020), especialmente na maneira como os quantis são selecionados e tratados.

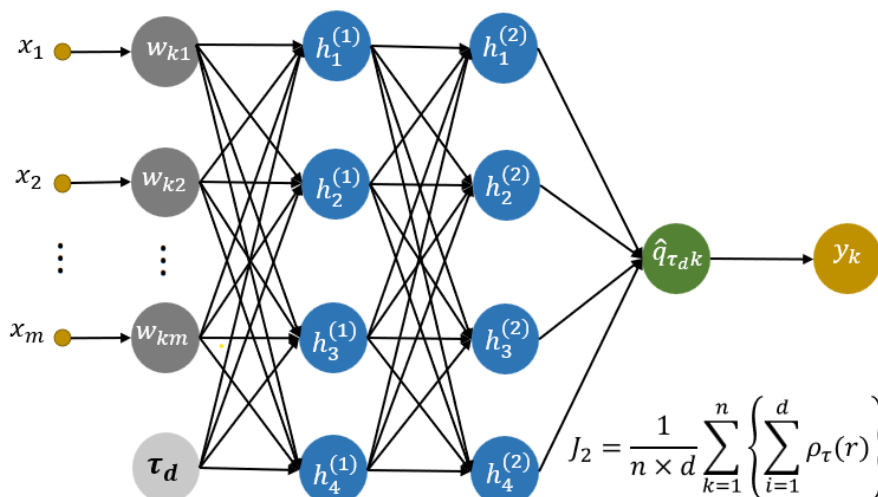


Figura 2.9: Regressão Quantílica via Rede Neural, proposta por Si, Kuleshov e Bishop (2022).
Fonte: Elaboração própria.

Em contraste com a abordagem de Rodrigues e Pereira (2020), que se baseava em quantis pré-definidos, limitando assim a flexibilidade do modelo, a proposta de Si, Kuleshov e Bishop (2022) introduz uma inovação ao considerar o quantil τ como uma entrada variável da rede neural, conforme ilustrado na Figura 2.9. Isso é realizado de maneira dinâmica, onde τ é amostrado de uma distribuição uniforme $\tau \sim U(0, 1)$. Essa abordagem se inspira nos métodos de Monte Carlo, conhecidos por sua capacidade de explorar espaços de amostra através da seleção aleatória de pontos. Para cada observação k no modelo, são amostrados múltiplos valores de τ , especificamente $\tau_1, \tau_2, \dots, \tau_d$, definindo assim um conjunto de d (número de τ) saídas. Este avanço metodológico permite ao modelo superar limitações anteriores, possibilitando a estimação direta para qualquer quantil de interesse, sem a necessidade de definir previamente os quantis a serem analisados.

Por último, Kuleshov e Deshpande (2022) introduziram outra inovação, visando superar a limitação de eficiência do método proposto inicialmente por Si, Kuleshov e Bishop (2022) em espaços preditores de dimensões moderadas a altas. Para alcançar esse objetivo, eles de-

envolveram uma abordagem que se vale de um procedimento de estimação em dois estágios, utilizando duas redes neurais profundas, conforme ilustrado na Figura 2.10.

O primeiro estágio envolve treinar um modelo de redes neurais que considera um conjunto de características iniciais como entrada (as covariáveis X) e produz um conjunto de características derivadas ou estatísticas resumo ϕ , pré-definidas pelo pesquisador. Esse passo visa principalmente à redução da dimensionalidade dos dados de entrada, facilitando assim a subseqüente estimação quantílica. Essa abordagem é essencial para aumentar a eficiência do modelo, otimizando tanto a velocidade de processamento quanto a precisão nas estimações quantílicas. Destaca-se, assim, por tornar o método mais ágil e preciso na execução de análises quantílicas, através da otimização na gestão da complexidade dos dados e no aprimoramento dos resultados analíticos.

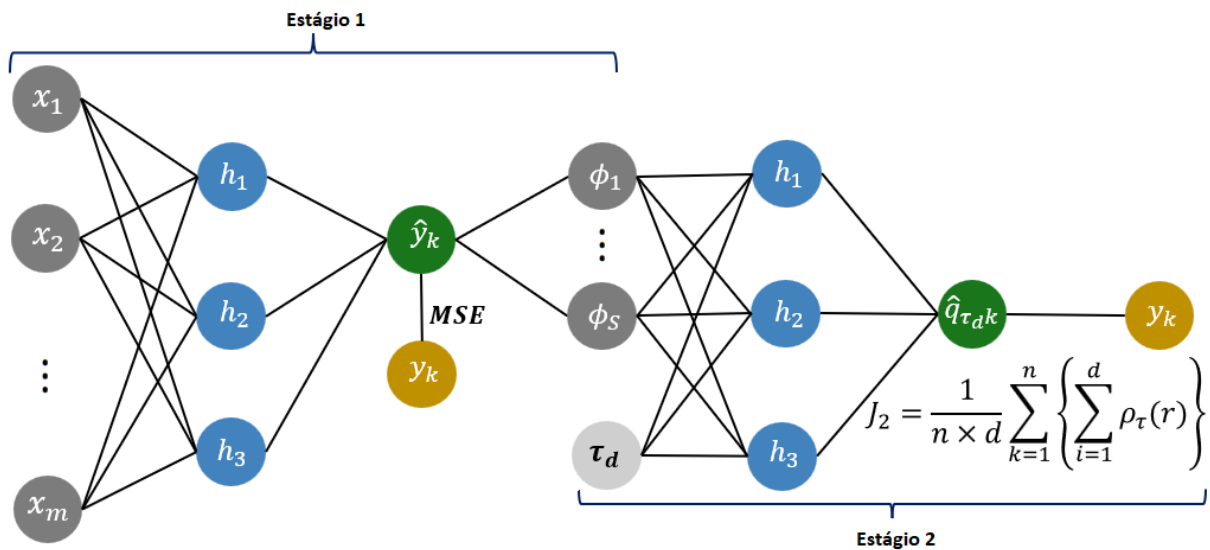


Figura 2.10: Regressão Quantílica via Redes Neurais, proposta por Kuleshov e Deshpande (2022).

Fonte: Elaboração própria.

No segundo estágio do algoritmo 1, utiliza-se a saída da primeira rede neural em uma segunda rede. Aqui, introduz-se um conjunto de dados separado, conhecido como conjunto de calibração. Este modelo auxiliar (segunda rede), frequentemente referido como recalibrador, serve para refinar as previsões do modelo inicial, levando em consideração a performance deste

nas amostras de calibração e ajustando as previsões para que melhor correspondam à realidade, ou seja, efetivamente recalibrando o modelo. O modelo de recalibração é então treinado para processar as características derivadas juntamente com valores aleatórios de espaço $U(0, 1)$ com o intuito de produzir previsões calibradas. Esses valores amostrados de τ , que indicam os quantis alvo, são introduzidos somente durante a fase de treinamento do recalibrador, que também recebe um vetor de baixa dimensão representando aspectos da distribuição condicional prevista pelo primeiro modelo (estatísticas resumos), podendo ser os quantis, variância, etc. Esse processo permite ao recalibrador entender como a representação de baixa dimensão pode ser utilizada para ajustar as previsões do primeiro modelo de forma eficaz.

Essa abordagem de Kuleshov e Deshpande (2022), Regressão Quantílica Multitarefa via Redes Neurais Profundas em dois Estágios (RQRN2E), apresenta claras vantagens em termos de robustez e versatilidade, conforme a Figura 2.10. Por outro lado, a versatilidade é evidenciada pela capacidade do modelo de se adaptar a diferentes contextos e tipos de dados, proporcionando uma ferramenta poderosa para uma ampla gama de aplicações analíticas. Além disso, o modelo não só mantém as vantagens do método original introduzido por Si, Kuleshov e Bishop (2022), permitindo a estimação direta de qualquer quantil de interesse, mas também aprimora significativamente a precisão e a eficiência da análise.

O algoritmo 1 descreve um procedimento em dois estágios, onde primeiramente um modelo base é treinado e suas previsões são utilizadas para gerar características resumo. Em seguida, um modelo de recalibração é treinado usando essas características, juntamente com valores amostrados, para ajustar as previsões iniciais. O objetivo é garantir que a distribuição das previsões recalibradas esteja bem calibrada.

Algorithm 1 RQRN2E (Kuleshov e Deshpande, 2022)

Estágio 1: Treinamento do Modelo Base

Entradas

- 1: • Dados de treinamento: variável resposta $\mathbf{Y}_{(n \times 1)}$ e matriz de covariáveis $\mathbf{X}_{(n \times m)}$.
- Hiperparâmetros da Rede Neural 1.

Treinamento em batches

- 2: **Para** $t = 1$ até T **faça**
- 3: Amostre um minibatch B_t de n_{B_t} pontos do conjunto de treinamento: $\mathbf{Y}_{(n_{B_t} \times 1)}^{B_t}$ e $\mathbf{X}_{(n_{B_t} \times m)}^{B_t}$.
- 4: Atualize os parâmetros θ da rede neural 1 via gradiente descendente estocástico (GDE) utilizando a função de perda escolhida para a Rede Neural 1.
- 5: **Fim do Para**

Saída

- 6: Estimativa dos parâmetros θ da Rede Neural 1.

Estágio 2: Treinamento do Modelo de Recalibração

Entradas

- 7: • Dados de recalibração: variável resposta $\mathbf{Y}_{(n \times 1)}$ e matriz de covariáveis $\mathbf{X}_{(n \times m)}$.
- Hiperparâmetros da Rede Neural 2.
- Número inteiro d representando a quantidade de quantis τ .
- Vetor com S estatísticas resumo $\phi = \phi_1^\theta(\mathbf{X}_{(n \times m)}), \phi_2^\theta(\mathbf{X}_{(n \times m)}), \dots, \phi_S^\theta(\mathbf{X}_{(n \times m)})$, onde θ indica a dependência com os parâmetros da Rede Neural 1.

Treinamento em batches

- 8: **Para** $t = 1$ até T **faça**
- 9: Amostre um minibatch B_t com n_{B_t} pontos do conjunto de recalibração: $\mathbf{Y}_{(n_{B_t} \times 1)}^{B_t}$ e $\mathbf{X}_{(n_{B_t} \times m)}^{B_t}$.
- 10: Crie uma matriz de dimensão $(n_{B_t} \times d)$ com valores amostrados da $U(0, 1)$: $\mathbf{M}_{(n_{B_t} \times d)}^{B_t} = [\boldsymbol{\tau}_1, \boldsymbol{\tau}_2, \dots, \boldsymbol{\tau}_d]$.
- 11: **Para** $i = 1$ até d **faça**
- 12: Concatene as estatísticas resumo $\phi_{(n_{B_t} \times S)}^{B_t}$ com a coluna $\boldsymbol{\tau}_i$ para criar as covariáveis de entrada da Rede Neural 2.
- 13: Prever os quantis $\hat{q}(\tau_{(1,i)} | X_1^{B_t}), \hat{q}(\tau_{(2,i)} | X_2^{B_t}), \dots, \hat{q}(\tau_{(n_{B_t},i)} | X_{n_{B_t}}^{B_t})$.
- 14: **Fim do Para**
- 15: Calcule a função de perda $J_{B_t} = \frac{1}{n_{B_t} \times d} \sum_{k=1}^{n_{B_t}} \left\{ \sum_{i=1}^d [\rho_\tau(y_k^{B_t} - \hat{q}(\tau_{(k,i)} | X_k^{B_t}))] \right\}$.
- 16: Atualize os parâmetros γ da Rede Neural 2 via GDE.
- 17: **Fim do Para**

Saída

- 18: Estimativa dos parâmetros γ da Rede Neural 2.
-

Capítulo 3

Metodologia

Este capítulo detalha o desenvolvimento da abordagem proposta, incluindo a nova arquitetura e suas inovações. Também aborda os métodos utilizados, como o pré-processamento, padronização dos dados, divisão para treinamento e validação, além dos modelos treinados e métricas de avaliação de desempenho. Por fim, são descritos os três conjuntos de dados utilizados: Diamonds, Boston e Yacht Hydrodynamics.

3.1 Nova Arquitetura

Essa dissertação propõe uma nova arquitetura que aprimora a abordagem introduzida por Kuleshov e Deshpande (2022), ao explorar uma estratégia de modelagem que busca integrar e simplificar o processo de estimação em um processo unificado. Essa nova arquitetura, ilustrada na Figura 3.1, rompe com a abordagem de dois estágios ao incorporar diretamente o parâmetro τ , representativo dos quantis gerados pela $U(0, 1)$, em uma camada de dimensão reduzida, posicionada em uma das camadas intermediárias da rede. Ela elimina a necessidade de uma etapa separada de definição da medida resumo ϕ e subsequentemente do treinamento de duas redes distintas, características centrais do método precedente.

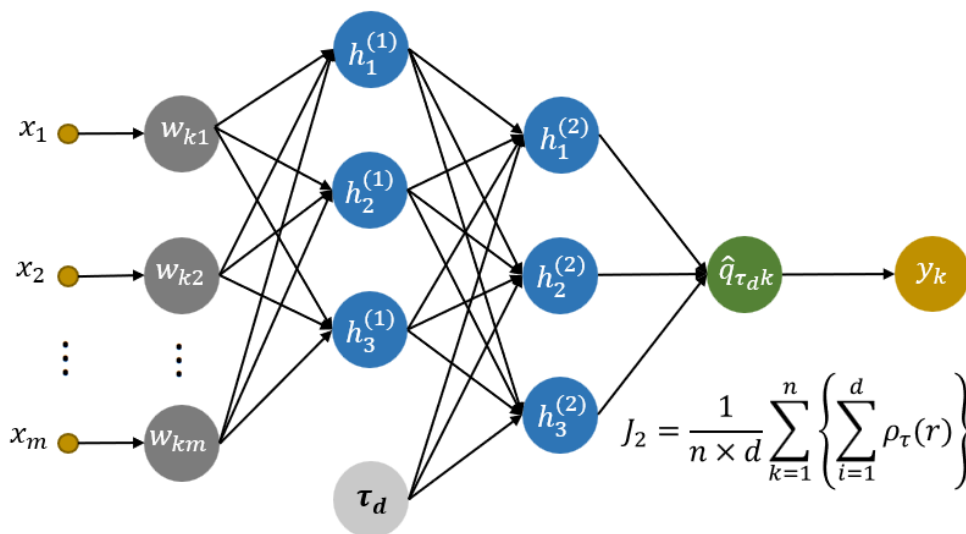


Figura 3.1: Nova arquitetura da Regressão Quantílica Multitarefa via Redes Neurais Profundas.
Fonte: Elaboração própria.

A inserção de τ diretamente na rede permite um ajuste mais integrado das previsões quantílicas. A função de perda Pinball, descrita na Equação 2.15, serviu de base para a criação da função de perda J_2 , que é utilizada na otimização, tornando o modelo sensível às variações de τ . Essa nova arquitetura oferece potenciais vantagens. Ao simplificar o modelo para um único estágio de ajuste que incorpora τ , há uma possível redução na complexidade computacional e operacional, o que pode tornar o processo de estimação quantílica mais eficiente. Além disso, essa abordagem unificada facilita a captura da relação entre os dados de entrada e os quantis de maneira mais direta, o que pode contribuir para uma maior precisão e confiabilidade nas previsões, dependendo do contexto de aplicação.

O algoritmo 2 proporciona um roteiro claro para o treinamento e a avaliação desse novo modelo, aqui denominado Regressão Quantílica Multitarefa via Redes Neurais Profundas em um Estágio (RQRN1E).

Algorithm 2 Nova Arquitetura- RQRNIE

Entradas

- 1: • Dados de treinamento: variável resposta $\mathbf{Y}_{(n \times 1)}$ e matriz de covariáveis $\mathbf{X}_{(n \times m)}$.
- Hiperparâmetros da rede neural.
- Número inteiro d representando a quantidade de quantis τ .
- Número inteiro c que indica a camada intermediária específica na qual a concatenação com o vetor de quantis τ será realizada.
- Número inteiro z que especifica a dimensão (ou tamanho) do vetor de saída resultante da camada intermediária c após a operação de concatenação.

Treinamento em batches

- 2: **Para** $t = 1$ até T **faça**
- 3: Amostre um mini-batch B_t com n_{B_t} pontos do conjunto de treinamento: $\mathbf{Y}_{(n_{B_t} \times 1)}$ e $\mathbf{X}_{(n_{B_t} \times m)}$.
- 4: Aplique as camadas da rede (até a camada intermediária c) para projetar $X_{(n_{B_t} \times m)}^{B_t}$ em $\phi_{(n_{B_t}, z)}$, reduzindo a dimensionalidade das covariáveis para posterior concatenação dos quantis.
- 5: Crie uma matriz com valores amostrados da $U(0, 1)$: $\mathbf{M}_{(n_{B_t} \times d)}^{B_t} = [\tau_1, \tau_2, \dots, \tau_d]$.
- 6: **Para** $i = 1$ até d **faça**
- 7: Concatene $\phi_{(n_{B_t}, z)}$ com τ_i para criar as covariáveis de entrada da camada $c+1$.
- 8: Prever os quantis $\hat{q}(\tau_{(1,i)} | X_1^{B_t}), \hat{q}(\tau_{(2,i)} | X_2^{B_t}), \dots, \hat{q}(\tau_{(n_{B_t}, i)} | X_{n_{B_t}}^{B_t})$.
- 9: **Fim do Para**
- 10: Calcule a função de perda $J_{B_t} = \frac{1}{n_{B_t} \times d} \sum_{k=1}^{n_{B_t}} \left\{ \sum_{i=1}^d [\rho_\tau (y_k^{B_t} - \hat{q}(\tau_{(k,i)} | X_k^{B_t}))] \right\}$.
- 11: Atualize os parâmetros θ da Rede Neural via GDE.
- 12: **Fim do Para**

Saída

- 13: Estimativa dos parâmetros θ da Rede Neural.
-

3.2 Conjunto de Dados

Para avaliar a performance da nova arquitetura de Regressão Quantílica Multitarefa via Redes Neurais Profundas, foram utilizados três conjuntos de dados distintos: *Diamonds*, *Boston* e *Yacht Hydrodynamics*. Esses conjuntos foram selecionados devido à sua utilização em estudos anteriores, permitindo uma comparação robusta com abordagens estabelecidas e oferecendo diversidade em termos de tamanho, tipo de variáveis e contexto de aplicação.

3.2.1 Diamonds

O conjunto de dados *Diamonds* foi extraído do pacote *'ggplot2'* no R e contém informações sobre 53.940 diamantes, cada um descrito por 10 variáveis. Destas, 9 foram consideradas como covariáveis (x_1, x_2, \dots, x_9) , incluindo características como o peso em quilates, cor, clareza, profundidade e comprimentos. A variável resposta (y) é o preço dos diamantes, em dólares americanos, uma variável contínua que reflete o valor de mercado dos diamantes. Este conjunto foi escolhido não apenas por sua popularidade em análises de regressão, mas também devido ao trabalho Torres et al. (2024), que utilizou essa base para recalibração de redes neurais. A variabilidade nas características dos diamantes e sua influência no preço fazem deste conjunto uma conveniente escolha para testar a capacidade do modelo em capturar múltiplos quantis da distribuição condicional do preço.

3.2.2 Boston

O conjunto de dados *Boston*, disponível no pacote *'MASS'* do R, fornece informações detalhadas sobre 506 áreas residenciais em Boston. Ele inclui 14 variáveis, das quais 13 foram consideradas como covariáveis, tais como taxa de criminalidade, proporção de residências antigas, distância ao centro de emprego, entre outras. A variável resposta (y) é o valor mediano das casas ocupadas pelos proprietários, medido em milhares de dólares. Este conjunto de dados é historicamente utilizado em estudos de regressão e modelagem preditiva, incluindo trabalhos

de Kuleshov e Deshpande (2022), servindo como um *benchmark* clássico para testar a eficácia de novos modelos. A combinação de variáveis socioeconômicas e demográficas torna este conjunto particularmente útil para avaliar a habilidade do modelo em lidar com relações complexas e não lineares entre as variáveis independentes e a variável dependente.

3.2.3 Yacht Hydrodynamics

O conjunto de dados *Yacht Hydrodynamics* foi obtido no Kaggle e contém 308 observações que descrevem o desempenho hidrodinâmico de iates. Com 7 variáveis ao todo, 6 foram usadas como covariáveis, sendo elas relacionadas às características do design do iate, como suas dimensões, centro de gravidade e enquanto a variável resposta (y) é a resistência hidrodinâmica, medida em newtons. Este conjunto de dados é amplamente utilizado em estudos de engenharia naval e foi também empregado por Kuleshov e Deshpande (2022) para testar a precisão de modelos de regressão quantílica.

3.3 Métodos

3.3.1 Pré-processamento dos Dados

O pré-processamento dos dados é uma etapa fundamental para a construção de modelos preditivos robustos. Primeiramente, foi realizada a identificação e remoção de valores omissos, zeros e *outliers* nos três conjuntos de dados analisados. Esses elementos, se não forem gerenciados adequadamente, podem comprometer a integridade e a qualidade dos modelos.

Em seguida, as variáveis categóricas foram transformadas em variáveis *dummy*, o que possibilita a sua integração eficiente nos modelos preditivos. Essa transformação é crucial, pois permite que informações qualitativas sejam tratadas de maneira quantitativa.

3.3.2 Padronização dos Dados

Após a limpeza dos dados, procedeu-se à padronização das covariáveis dos três conjuntos de dados. A padronização foi realizada de forma a garantir que todas as covariáveis tivessem média zero e desvio padrão unitário, o que é crucial para a convergência mais rápida e estável dos modelos de redes neurais. Essa etapa foi aplicada exclusivamente às variáveis independentes, preservando a escala original da variável de resposta para manter a integridade da estimação quantílica.

3.3.3 Divisão dos Conjuntos de Dados

A divisão dos dados foi realizada em três conjuntos: treino, validação e teste, com uma distribuição de 70%, 20% e 10%, respectivamente. Essa divisão foi escolhida para proporcionar um equilíbrio entre o treinamento do modelo e a validação do seu desempenho em dados não vistos durante o processo de treinamento. A mesma proporção foi utilizada por Kuleshov e Deshpande (2022), proporcionando uma boa base comparativa para a avaliação da nossa proposta.

O conjunto de treino foi utilizado para ajustar os pesos do modelo, enquanto o conjunto de validação foi essencial para a otimização dos hiperparâmetros e para o ajuste fino do modelo, prevenindo o *overfitting*. Finalmente, o conjunto de teste foi reservado para a avaliação final do modelo, garantindo uma medida independente.

3.3.4 Modelos Treinados

Foram treinados três modelos distintos para este estudo, envolvendo abordagens de regressão quantílica e regressão gaussiana. O primeiro modelo, RQRN2E (Regressão Quantílica Multi-tarefa via Redes Neurais Profundas em dois Estágios), foi proposto por Kuleshov e Deshpande (2022) e consiste em uma arquitetura de dois estágios. No primeiro estágio, uma rede neural gera um conjunto de características decis (d_1, d_2, \dots, d_9) das distribuições preditivas das respostas condicionais, que são então utilizados no segundo estágio para estimar os quantis

desejados.

Segundo Kuleshov e Deshpande (2022), o parâmetro τ não deve ser inserido no início da rede, pois essa informação fica ofuscada (ou perde a relevância) junto das demais covariáveis do modelo e da sequência de transformações não lineares das camadas iniciais da rede. Dessa forma, os autores sugeriram dois estágios na modelagem. Nessa dissertação, propomos uma arquitetura unificada para estimação dos quantis em um único estágio, denominada RQRN1E (Regressão Quantílica Multitarefa via Redes Neurais Profundas em um Estágio), com a introdução do parâmetro τ diretamente em uma camada intermediária da rede, permitindo uma estimativa direta dos quantis. Ao acrescentar o τ ao final da rede, em uma camada de reduzida dimensão, maior destaque se dá à informação do nível τ do quantil, ainda mantendo a estrutura de um estágio na estimação. Dessa forma, contorna-se as limitações decorrentes da escolha das estatísticas resumo ϕ e as ineficiências da separação em dois estágios. Com isso espera-se uma representação mais robusta das dinâmicas complexas dos dados e melhor desempenho da rede.

A segunda inovação proposta foi a utilização da função de ativação logito na inserção do τ . Ao utilizar o logito do τ , o RQRN1E demonstrou uma capacidade superior de modelar as relações não lineares, resultando em um melhor ajuste e previsões mais precisas.

O terceiro modelo utilizado para comparação foi a RGRN (Regressão Gaussiana via Redes Neurais Profundas). Este modelo segue uma abordagem tradicional de regressão gaussiana em redes neurais profundas, projetada para prever a média e a variância de uma distribuição normal com base nas covariáveis disponíveis. Assumindo que os dados seguem uma distribuição Normal, podemos utilizar esse modelo também para prever os quantis, sendo uma alternativa interessante para conjunto de dados pequenos e aproximadamente Normais.

3.3.5 Métricas de Desempenho

Para avaliar o desempenho dos modelos propostos, foram utilizadas cinco métricas principais: RMSE (Root Mean Square Error), MAE (Mean Absolute Error), MADECP (Mean Absolute Distance of Empirical Cumulative Probability), Perda Quantílica e a Cobertura. Cada uma dessas

métricas oferece uma perspectiva única sobre a qualidade das previsões feitas pelos modelos, sendo fundamentais para comparar a precisão e a robustez das abordagens.

- **Root Mean Square Error**

O RMSE fornece a raiz quadrada da média entre os valores preditos e os valores reais observados. Esta métrica é amplamente utilizada para avaliar a precisão de modelos preditivos, penalizando com mais peso os erros maiores, já que as diferenças são elevadas ao quadrado (Willmott e Matsuura, 2005; Chai e Draxler, 2014). A fórmula do RMSE é dada por:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{k=1}^n (y_k - \hat{y}_k)^2}. \quad (3.1)$$

Uma métrica robusta para quantificar o erro global de um modelo, sendo especialmente sensível a grandes discrepâncias. Dentro do contexto de regressão, essa medida é utilizada para estimação do valor esperado da variável resposta.

- **Mean Absolute Error**

O MAE é uma métrica mais simples que o RMSE, que calcula a média das diferenças absolutas entre os valores preditos e os observados. Ao contrário do RMSE, o MAE não dá peso extra a grandes erros, sendo uma métrica útil quando o objetivo é avaliar a precisão de previsões de forma uniforme (Willmott e Matsuura, 2005; Chai e Draxler, 2014). A fórmula do MAE é:

$$\text{MAE} = \frac{1}{n} \sum_{k=1}^n |y_k - \hat{y}_k|. \quad (3.2)$$

Essa métrica oferece uma medida clara e intuitiva da precisão das previsões, mostrando o erro médio entre os valores preditos e observados em termos absolutos. Dentro do contexto de regressão quantílica, essa medida é utilizada para estimação do valor mediano da variável resposta.

- **Mean Absolute Distance of Empirical Cumulative Probability**

A métrica MADECP foi proposta por Amorim (2024) na sua dissertação, ela é utilizada para avaliar a qualidade dos múltiplos quantis preditos em modelos de Regressão Quantílica, medindo a consistência entre os quantis preditos e a distribuição empírica dos dados. Para calculá-la, utiliza-se uma grade de 99 valores de τ ($\tau_{grid} = [0.01, 0.02, \dots, 0.99]^T$). Para cada observação k , é feita uma predição de quantil para cada valor de τ na grade. A fórmula da MADECP é:

$$\text{MADECP} = \frac{1}{99} \sum_{i=1}^{99} \left| \tau_i - \frac{1}{n} \sum_{k=1}^n \mathbb{1}(y_k \leq \hat{q}_k(\tau_i)) \right|, \quad (3.3)$$

onde $\mathbb{1}(\cdot)$ é a função indicadora, que retorna 1 se a condição for verdadeira e 0 caso contrário. Essa métrica é particularmente útil para verificar se os quantis preditos refletem adequadamente a distribuição empírica dos dados ao longo de todo o espaço de probabilidades. Um valor mais baixo de MADECP indica maior consistência entre os quantis preditos e a distribuição empírica.

- **Perda Quantílica**

A Perda Quantílica, também conhecida como Perda Pinball, é a métrica padrão para avaliar o desempenho de modelos de regressão quantílica, conforme descrito nas equações 2.3 e 2.15. Ela mede a precisão do modelo na previsão de um quantil específico da distribuição condicional da variável dependente. A função de perda penaliza de forma assimétrica os desvios entre os valores observados e preditos, dependendo do quantil τ alvo, sendo especialmente eficaz para capturar a variação da distribuição (Koenker, 2005). A fórmula da Perda Quantílica é:

$$J_\tau = \frac{1}{n} \sum_{k=1}^n \rho_\tau(y_k - \hat{q}(\tau|x_k)), \quad (3.4)$$

onde $\rho_\tau(r) = r(\tau - \mathbb{1}(r < 0))$ é a função de perda ponderada pelo quantil τ . Essa métrica é amplamente utilizada para avaliar a precisão das previsões de quantis em um modelo de regres-

são quantílica. Valores mais baixos de Perda Quantílica indicam predições mais precisas para o quantil alvo.

- **Cobertura**

A métrica Cobertura mede a proporção de valores observados que se encontram dentro dos intervalos de confiança preditos pelo modelo, para um determinado nível de confiança (geralmente 95%). A fórmula da Cobertura é:

$$\text{Cobertura} = \frac{1}{n} \sum_{k=1}^n \mathbb{1}(\hat{q}\tau_L(x_k) \leq y_k \leq \hat{q}\tau_U(x_k)), \quad (3.5)$$

onde $\hat{q}\tau_L(x_k)$ e $\hat{q}\tau_U(x_k)$ são os quantis preditos para os níveis inferiores e superiores do intervalo de confiança, respectivamente. Um valor de Cobertura de 95%, por exemplo, indica que 95% dos valores observados estão dentro do intervalo predito pelo modelo. Essa métrica é importante para avaliar a calibragem do modelo em capturar a incerteza nas predições (Davison e Hinkley, 1997).

Capítulo 4

Resultados

Neste capítulo, serão apresentados os resultados dos modelos RQRN2E, RQRN1E e RGRN, aplicados aos dados Diamonds, Boston e Yacht Hydrodynamics. A análise comparativa utilizará métricas como perda quantílica, MADECP, RMSE, MAE e Cobertura (95%), destacando o desempenho e eficiência de cada modelo, com foco na precisão e na identificação do modelo mais adequado para a estimativa de quantis.

4.1 Aplicação no Banco Diamonds

Antes da modelagem, realizou-se uma análise exploratória da variável resposta (y), referente ao preço dos diamantes em dólares americanos. Com base na Figura 4.1, a distribuição dos preços apresenta uma acentuada assimetria à direita, o preço da maioria dos diamantes está concentrado entre 300 e 5.000 dólares, enquanto o preço máximo atinge 18.823 dólares. Essa variabilidade sugere a necessidade de modelos que capturem a heterocedasticidade e a assimetria, como os modelos de regressão quantílica utilizados neste estudo, para uma análise preditiva mais pertinente.

O ajuste dos hiperparâmetros dos modelos foi realizado manualmente, explorando diferentes valores até alcançar um desempenho ótimo em termos de minimizar a função de perda

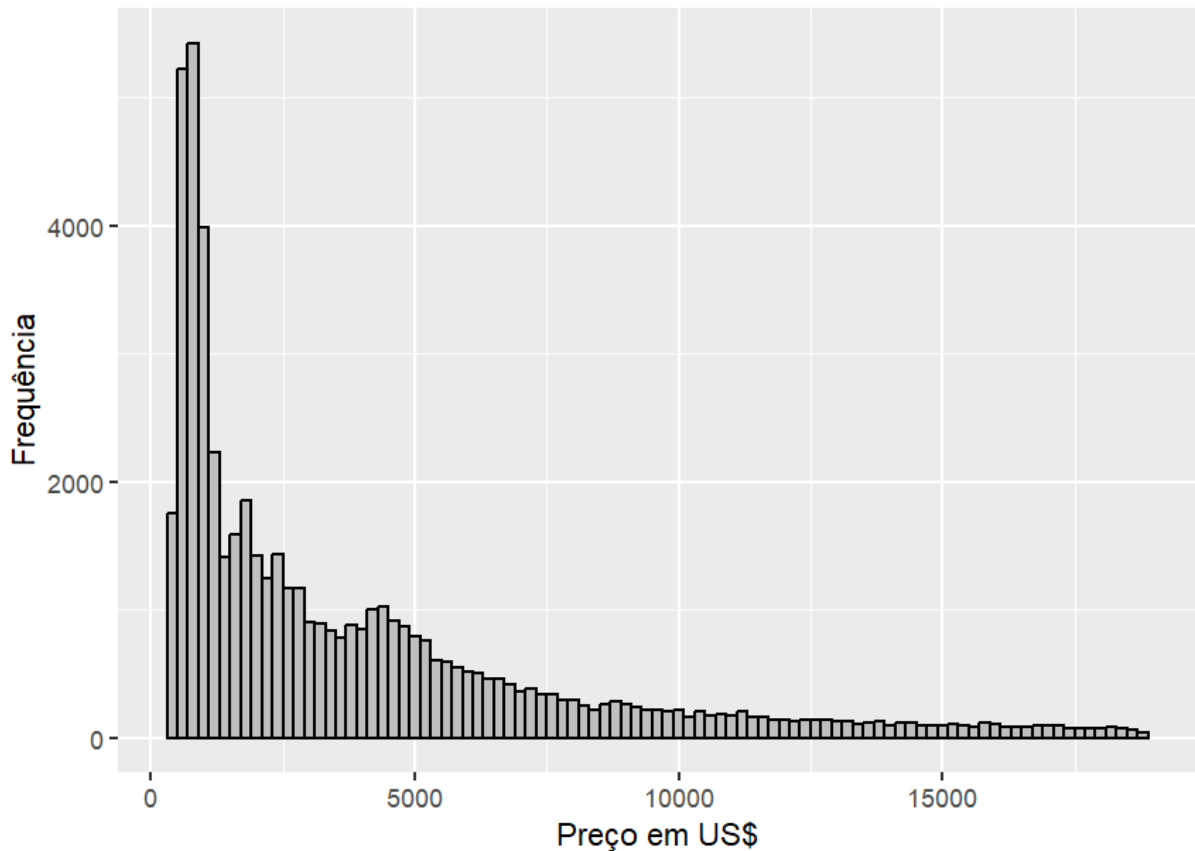


Figura 4.1: Distribuição da variável preço dos diamantes em dólares.

Fonte: Elaboração própria.

quantílica na validação evitando o *overfitting*.

Para o modelo RQRN2E, proposto por Kuleshov e Deshpande (2022), o ajuste ocorreu em dois estágios, partindo do chute inicial proposto por Torres et al. (2024). No primeiro estágio, foram testadas três camadas ocultas, com a primeira variando entre 100 e 150 neurônios, sendo 150 o valor ótimo. As camadas subsequentes apresentaram entre 5 e 10 neurônios, com o valor ótimo estabelecido em 10 neurônios para cada. O número de épocas variou entre 250 e 500, sendo 250 o ideal, com uma taxa de aprendizagem de 0,01 e tamanho de lote (*batch size*) de 500. A paciência para o *early stopping* foi definida em 10 épocas.

No segundo estágio, foram utilizadas duas camadas ocultas, ambas com 30 neurônios, e o modelo foi treinado por 2.000 épocas. A taxa de aprendizagem foi ajustada para 0,001 e o

tamanho de lote permaneceu em 500, com paciência de 50 épocas para o *early stopping*.

Já para o modelo RQRN1E, proposto nesta dissertação, foi feito um ajuste inicial com quatro camadas ocultas. As duas primeiras camadas variaram entre 100 e 128 neurônios, sendo 100 o valor ideal. Nas duas últimas camadas, a variação foi entre 50 e 64 neurônios, com 50 como valor ótimo. O modelo foi treinado por 500 épocas, com uma taxa de aprendizagem de 0,01 e tamanho de lote de 500, e paciência de 10 épocas para o *early stopping*. Em relação à inserção do logito do parâmetro τ , foram testadas a penúltima e a última camada, sendo a penúltima a que apresentou melhores resultados.

Finalmente, o modelo RGRN utilizou os mesmos hiperparâmetros do modelo RQRN1E, já que ambos compartilham a mesma arquitetura para facilitar a comparação. A única diferença é que o RQRN1E introduziu o logito de τ em uma de suas camadas, enquanto o RGRN não tem a informação do quantil e utilizou a função de perda Gaussiana em vez da função de perda quantílica.

A Tabela 4.1 resume o desempenho dos modelos nos conjuntos de treino e validação. Para o modelo RGRN, a distribuição Gaussiana foi assumida aos dados para estimação dos quantis e para o cálculo da perda quantílica e da cobertura de 95%. A medida de cobertura preditiva indica o percentual de vezes que o valor observado está contido dentro do intervalo de confiança preditivo. Considerando a Tabela 4.1, a proximidade das perdas do treino e validação de cada modelo indica que nenhum apresentou *overfitting* expressivo. Comparando a perda quantílica na validação, observa-se que o modelo RQRN1E apresentou os melhores resultados com taxa de cobertura próxima a 95%.

Tabela 4.1: Desempenho dos modelos nos conjuntos de treino e validação no banco *Diamonds*.

Modelos	Perda Quantílica (Treino)	Perda Quantílica (Validação)	Cobertura (95%) (Validação)	Épocas	Tempo (Segundos)
RQRN2E	127,2	129,1	63,1%	1.739	415
RQRN1E	103,4	107,4	94,5%	59	59
RGRN	124,4	130,0	94,5%	92	93

Além disso, a eficiência computacional do RQRN1E é outro aspecto interessante. Com um tempo de convergência de apenas 59 segundos e 59 épocas, ele se mostrou mais eficaz em comparação ao RQRN2E, que demandou 415 segundos e 1.739 épocas. Esse desempenho mais rápido do RQRN1E é particularmente vantajoso em cenários onde o tempo de processamento é crítico. A arquitetura simplificada em um único estágio, aliada à introdução do logito de τ em uma camada intermediária, provavelmente contribuiu para essa eficiência, sem comprometer a qualidade das previsões, além dos hiperparâmetros ótimos de cada modelo.

A Tabela 4.2 complementa essas informações ao apresentar os resultados nos dados de teste, com ênfase nas métricas de erro relacionados à média e mediana (RMSE e MAE), na perda quantílica e na medida de erro baseada nas probabilidades acumuladas empíricas (MADECP). O modelo RQRN1E apresentou uma menor perda quantílica de 107,6, uma menor MADECP de 0,024 e uma cobertura de 94,5%. Estes resultados indicam que o RQRN1E é o mais eficiente na estimativa dos quantis e na cobertura preditiva, alinhando-se bem com o objetivo do estudo. No cálculo do RMSE e MAE dos modelos RQRN1E e RQRN2E, ao invés de utilizar a média estimada \hat{y}_i , foi utilizada a mediana $q_i(\hat{0}.5)$, visto que esses modelos não estimam de forma direta a média.

Em termos de erros associados à média e mediana, o modelo RQRN1E teve uma performance razoavelmente boa para um quantil fixo de 0,5 (mediana), com RMSE de 556,8 e MAE de 288,9. Considerando que o objetivo central do modelo RQRN1E é a predição de múltiplos quantis, e não apenas da média e da mediana, as medidas de perda quantílica, cobertura 95% e MADECP são mais relevantes.

Comparando com o RQRN2E, embora este modelo tenha apresentado valores melhores para RMSE (565,0) e MAE (287,1) para um quantil fixo de 0,5 (mediana), sua perda quantílica (121,1) e MADECP (0,059) foram mais altos. A taxa de cobertura também foi menor no RQRN2E (63,0%), indicando que este modelo não capturou tão bem os intervalos de confiança preditivos quanto o RQRN1E. É possível que uma configuração de rede otimizada possa melhorar as métricas apresentadas, visto que a otimização dos hiperparâmetros foi realizada ma-

nualmente. Técnicas de otimização automática não foram aplicadas e podem ser consideradas em trabalhos futuros na tentativa de melhorar ainda mais a performance dos modelos.

Tabela 4.2: Desempenho dos modelos no conjunto de teste para o banco *Diamonds*.

Modelos	RMSE	MAE	Perda Quantílica	Cobertura (95%)	MADECP
RQRN2E	565,0	287,1	121,1	63,0%	0,059
RQRN1E	556,8	288,9	107,6	94,5%	0,024
RGRN	540,3	287,3	127,2	95,0%	0,104

O modelo RGRN, por outro lado, apresentou o melhor RMSE (540,3) e cobertura (95%), e um bom valor de MAE (287,3), o que indica uma boa performance na previsão de valores médios e medianos. No entanto, sua perda quantílica (127,2) e MADECP (0,104) foram as piores entre os modelos testados. O valor MADECP igual a 0,104 indica que houve, em média, 10,4% de desvio (erro) dos níveis dos percentis estimados pelo modelo RGRN. Por exemplo, o percentil 25% estimado pode apresentar 14,6% (ou 35,4%) das observações abaixo dele, enquanto esperava-se que esse valor fosse 25%, aproximadamente. Esse desempenho sugere que a RGRN não é tão eficiente na estimação dos quantis do conjunto de dados *Diamonds*. Considerando a assimetria da distribuição do preço dos diamantes (Figura 4.1), esse resultado já era esperado.

Vale ressaltar que a taxa de cobertura, que reflete o percentual de vezes que o valor observado da variável resposta está dentro do intervalo de confiança preditivo, é uma medida global que não captura adequadamente as deficiências locais de calibração dos modelos (ou seja, em sub-regiões do espaço de covariáveis). Como discutido por Torres et al. (2024), modelos gaussianos como a RGRN podem não ser bem calibrados localmente, mas a calibração local não foi investigada nessa dissertação. A medida MADECP, que leva em consideração diversas probabilidades acumuladas da distribuição empírica (ainda que de forma global), melhor retrata a calibração do modelo na estimativa dos quantis.

4.2 Aplicação no Banco Boston

A Figura 4.2 ilustra o histograma da variável resposta, que representa o valor mediano das casas em *Boston* em milhares de dólares. Nota-se que o preço tem uma distribuição assimétrica à direita, com a maioria das observações concentradas entre 12 e 30 mil dólares. O valor mínimo é de 5 mil dólares e o máximo de 50 mil dólares, com um possível efeito de truncamento nesse limite superior. Essa distribuição sugere a necessidade de modelos que capturem adequadamente a distribuição dos dados, como os modelos quantílicos, especialmente para estimar valores extremos.

Assim como na aplicação anterior, o ajuste dos hiperparâmetros dos modelos foi realizado manualmente, explorando diferentes valores até alcançar um desempenho ótimo em termos de minimizar a função de perda quantílica na validação e evitar o *overfitting*.

Para o modelo RQRN2E, conforme proposto por Kuleshov e Deshpande (2022), o ajuste ocorreu em dois estágios, partindo do chute inicial proposto por Torres et al. (2024). No primeiro estágio, foram testadas três camadas ocultas, onde o número de neurônios na primeira camada variou de 10 a 200, sendo 128 o valor ótimo. As camadas subsequentes variaram entre 5 e 10 neurônios, com 10 sendo o valor ideal. O número de épocas variou de 500 a 700, sendo 500 o valor ótimo, e a taxa de *dropout* oscilou entre 10% e 50%, com 50% se mostrando ideal. A taxa de aprendizagem testada variou entre 0,001 e 0,01, com 0,01 sendo o valor ótimo, enquanto o tamanho do lote (*batch size*) variou de 250 a 1.000, sendo 500 o valor ideal. A paciência para o *early stopping* foi ajustada para 10 épocas, num intervalo entre 10 e 100.

No segundo estágio do RQRN2E, foram utilizadas duas camadas ocultas, onde o número de neurônios na primeira camada variou de 30 a 175, sendo 20 o valor ótimo, e a última camada variou de 15 a 50, com 20 sendo o valor ideal. O número de épocas testado foi entre 500 e 1.500, com 1.000 sendo o valor ideal. A taxa de aprendizagem variou de 0,001 a 0,01, sendo 0,01 o valor ótimo, e o tamanho do lote variou de 500 a 750, sendo 750 o ideal. A paciência para o *early stopping* foi ajustada para 50 épocas, num intervalo entre 10 e 50.

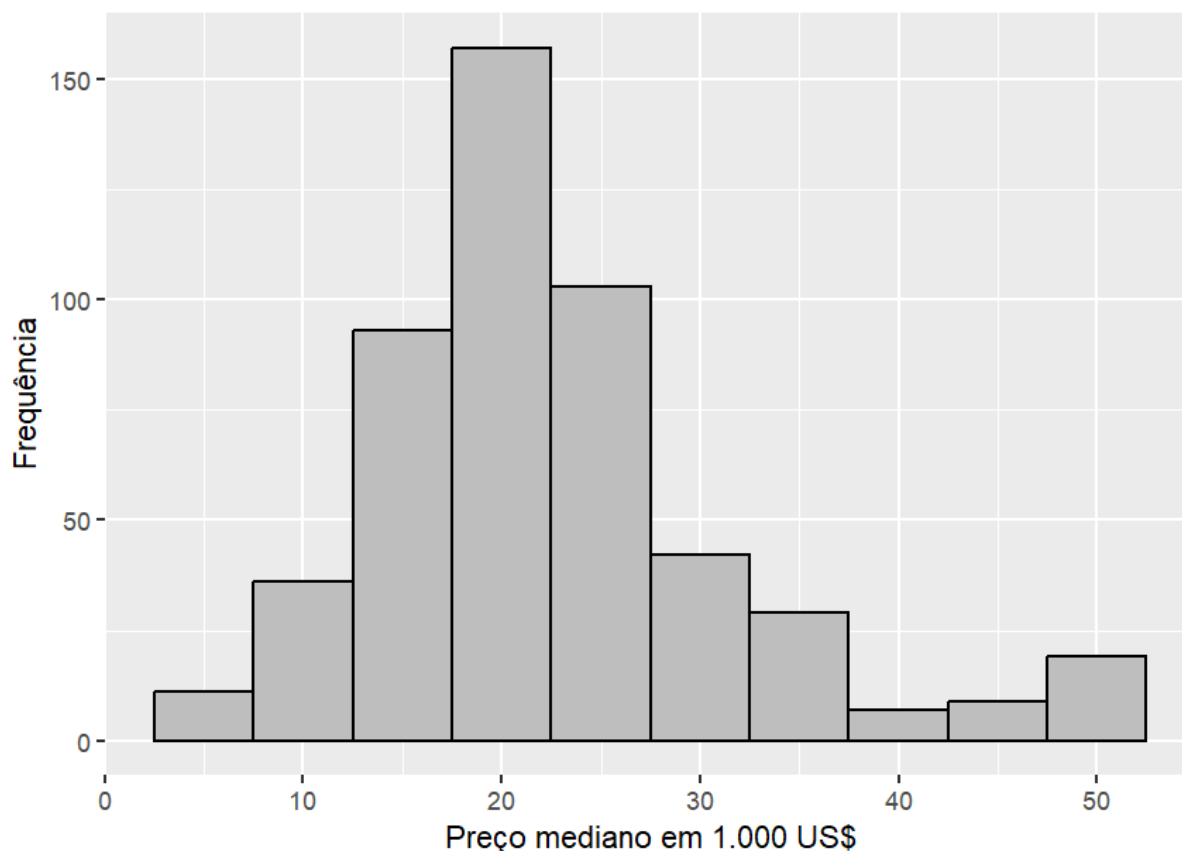


Figura 4.2: Distribuição do preço mediano das casas do banco *Boston*.

Fonte: Elaboração própria.

Para o modelo RQRN1E, desenvolvido nesta dissertação, o ajuste também considerou diferentes variações. Foram testadas quatro camadas ocultas, com o número de neurônios variando entre 25 e 190, sendo 75 o valor ótimo em todas as camadas. A taxa de *dropout* variou entre 10% e 30%, sendo 10% o valor ideal. O número de épocas foi ajustado entre 500 e 1.000, com 500 sendo o valor ótimo. A taxa de aprendizagem variou entre 0,001 e 0,01, sendo 0,01 o valor ótimo, enquanto o tamanho do lote variou entre 500 e 1.000, sendo 500 o valor ideal. A paciência para o *early stopping* foi ajustada para 10 épocas, num intervalo entre 10 e 50. Em relação à inserção do logito do parâmetro τ , foram testadas a penúltima e a última camada, sendo a penúltima a que apresentou melhores resultados. Para fins de comparação, o modelo RGRN foi treinado utilizando os mesmos hiperparâmetros do RQRN1E.

Na Tabela 4.3, que resume o desempenho dos modelos nos conjuntos de treino e validação para o conjunto de dados de *Boston*, o modelo RQRN1E se destaca por apresentar a menor perda quantílica tanto no treino quanto na validação, com valores de 1,2 em ambas as fases. Isso indica que o RQRN1E tem uma boa capacidade de generalização e menor erro de ajuste para os quantis. Além disso, o modelo obteve a taxa de cobertura de 95% igual a 95,1%, fornecendo estimativas melhor calibradas do que os outros modelos comparados. Com um tempo de convergência de apenas 39 segundos e 130 épocas, o RQRN1E também se mostrou o mais eficiente em termos de tempo computacional.

Tabela 4.3: Desempenho dos modelos nos conjuntos de treino e validação no banco *Boston*.

Modelos	Perda Quantílica (Treino)	Perda Quantílica (Validação)	Cobertura (95%) (Validação)	Épocas	Tempo (Segundos)
RQRN2E	1,8	1,7	81,3%	520	156
RQRN1E	1,2	1,2	95,1%	130	39
RGRN	1,5	1,4	96,1%	181	52

Na Tabela 4.4, que avalia o desempenho dos modelos no conjunto de teste, o modelo RQRN1E continua a se destacar, com a menor perda quantílica de 1,4 e cobertura de aproximadamente 90% e menor erro MADECP (0,023). Esses resultados reforçam a sua eficácia na estimativa dos quantis, com um boa performance nas métricas RMSE e MAE para um quantil fixo de 0,5 (mediana), mas um pouco abaixo em comparação com a RGRN, conforme também observado no conjunto de dados *Diamonds*.

Tabela 4.4: Desempenho dos modelos no conjunto de teste no banco *Boston*.

Modelos	RMSE	MAE	Perda Quantílica	Cobertura (95%)	MADECP
RQRN2E	8,0	5,5	2,2	85,4%	0,033
RQRN1E	6,4	3,9	1,4	89,6%	0,023
RGRN	6,2	3,7	1,6	91,7%	0,037

O modelo RGRN se destaca na minimização do RMSE e MAE, que são métricas de erro focadas nas medidas de centralidade (média e mediana) dos dados. A cobertura de 92%, aproximadamente, foi um bom resultado para calibração global. Já o modelo RQRN2E mostrou-se

menos eficaz nas métricas apresentadas. É possível que uma configuração de rede otimizada possa melhorar as métricas apresentadas, visto que o ajuste dos hiperparâmetros foi efetuado manualmente. Entretanto, dentre as diversas arquiteturas de rede testadas, as configurações de rede aqui apresentadas foram as que minimizaram a perda quantílica na validação sem *overfitting* expressivo.

4.3 Aplicação no Banco Yacht Hydrodynamics

Ao analisar o banco *Yacht Hydrodynamics*, iniciamos com uma exploração da variável de interesse, a resistência hidrodinâmica medida em newtons. Na Figura 4.3, o histograma mostra que a distribuição é assimétrica positiva, com a maioria dos valores próximos a zero e uma cauda longa se estendendo até 62,42 newtons. Essa distribuição indica que a maioria dos iates têm baixa resistência, mas alguns apresentam valores significativamente maiores. Essa assimetria é importante na escolha do modelo, com provável impacto negativo na RGRN devido à suposição de Normalidade para o cálculo dos quantis.

O ajuste dos hiperparâmetros dos modelos foi realizado manualmente, explorando diferentes valores para alcançar um desempenho ótimo na minimização da função de perda quantílica na validação, sem comprometer consideravelmente o *overfitting*, assim como nos outros banco de dados aqui explorados.

Para o modelo RQRN2E, proposto por Kuleshov e Deshpande (2022), o ajuste ocorreu em dois estágios, partindo do chute inicial proposto por Torres et al. (2024). No primeiro estágio, foram testadas três camadas ocultas, com a primeira camada variando entre 100 e 200 neurônios, sendo 100 o valor considerado ideal. A segunda camada apresentou entre 5 e 10 neurônios, mas a configuração ideal incluiu 5 neurônios, e a terceira camada foi excluída no ajuste final. O número de épocas variou entre 1.000 e 5.000, com 1.000 sendo considerado o ideal, e a taxa de aprendizagem variou entre 0,001 e 0,01, sendo 0,01 o valor ideal. O tamanho de lote (*batch size*) variou entre 250 e 750, sendo 750 considerado o melhor ajuste. A paciência para o *early*

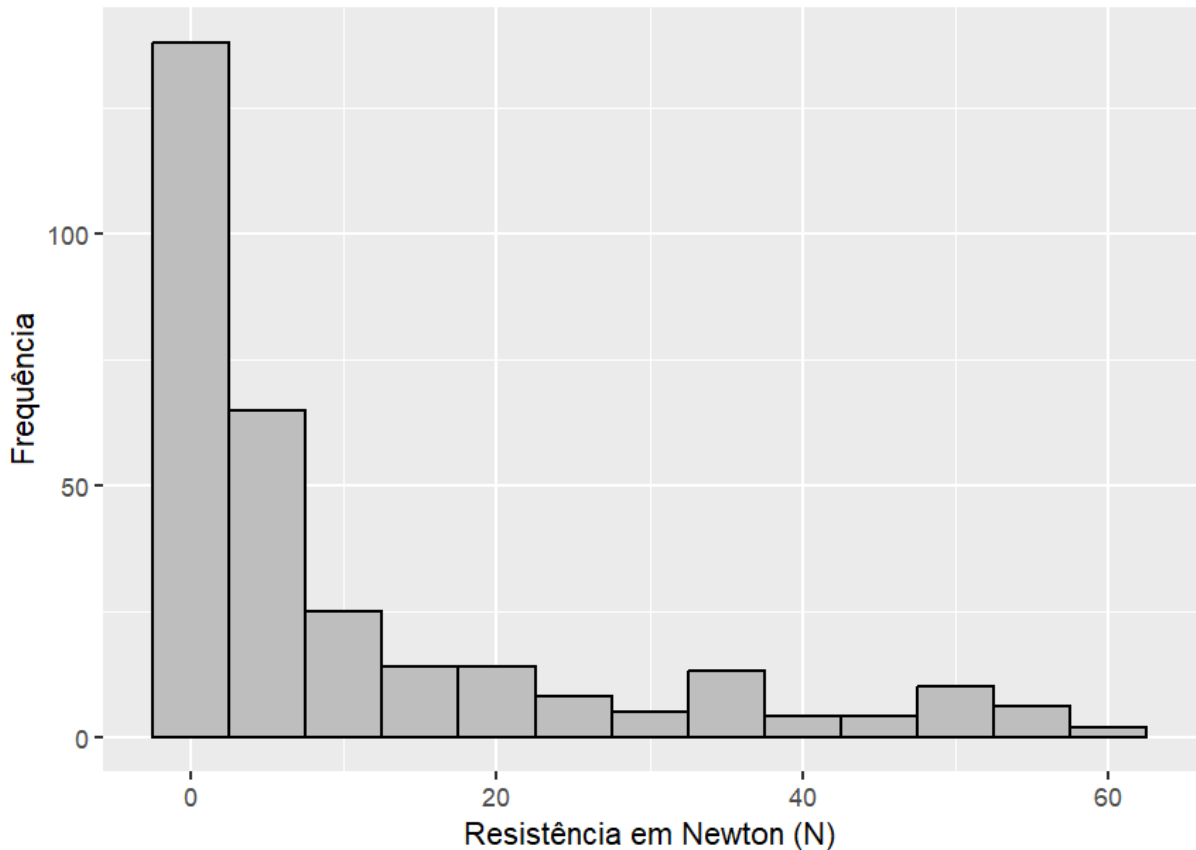


Figura 4.3: Distribuição da variável resistência no banco *Yacht Hydrodynamics*.
Fonte: Elaboração própria.

stopping variou inicialmente entre 10 e 20 épocas, sendo definida em 20 épocas como valor ideal.

No segundo estágio do modelo RQRN2E, foram consideradas duas camadas ocultas, com a primeira variando entre 30 e 40 neurônios e a segunda entre 25 e 30 neurônios. Os valores ótimos encontrados foram 30 neurônios para a primeira camada e 25 neurônios para a segunda. As épocas variaram entre 500 e 1.000, com 500 épocas consideradas ideais, e a taxa de aprendizagem foi mantida entre 0,001 e 0,01, com 0,01 como valor final. O tamanho de lote variou entre 500 e 750, sendo 750 o ideal, e a paciência para o *early stopping* variou entre 15 e 30 épocas, com 30 épocas definidas como valor final.

Para o modelo RQRN1E, uma proposta desta dissertação, foram testadas quatro camadas

ocultas. Nas camadas 1 e 2, o número de neurônios variou entre 100 e 150, sendo 150 o valor ótimo para ambas as camadas. Nas camadas 3 e 4, o número de neurônios variou entre 5 e 100, sendo 100 neurônios o valor ótimo para a camada 3 e 50 neurônios para a camada 4. As épocas variaram entre 500 e 1.000, com 1.000 consideradas ideais. A taxa de aprendizagem foi ajustada entre 0,001 e 0,01, sendo 0,01 o valor ótimo, e o tamanho de lote foi fixado em 500. A paciência para o *early stopping* foi estabelecida em 100 épocas, sem variações. Em relação à inserção do logito do parâmetro τ , foram testadas a penúltima e a última camada, sendo a penúltima a que apresentou melhores resultados.

O modelo RGRN compartilha a mesma arquitetura básica que o RQRN1E, mas com a diferença fundamental de utilizar a função de perda Gaussiana em vez da função de perda quantílica. Para o RGRN, foram utilizados os mesmos hiperparâmetros do RQRN1E.

Na Tabela 4.5, que apresenta o desempenho dos modelos nos conjuntos de treino e validação para o conjunto de dados *Yacht*, o modelo RQRN1E destaca-se pela menor perda quantílica, registrando valores de 0,12 no treino e 0,14 na validação. Esses resultados indicam uma boa capacidade do modelo em estimar quantis tanto nos dados de treino quanto nos de validação. A taxa de cobertura foi de aproximadamente 97%, indicando uma calibração próxima ao valor nominal de 95%. O tempo computacional do RQRN1E foi de 97 segundos, com um total de 407 épocas, demonstrando um equilíbrio entre precisão e eficiência computacional. Em comparação, os tempos computacionais dos modelos RQRN1E e RGRN foram similares, ambos consumindo cerca da metade do tempo necessário para o modelo de dois estágios, o RQRN2E.

Tabela 4.5: Desempenho dos modelos nos conjuntos de treino e validação no banco *Yacht Hydrodynamics*.

Modelos	Perda Quantílica (Treino)	Perda Quantílica (Validação)	Cobertura (95%) (Validação)	Épocas	Tempo (Segundos)
RQRN2E	0,28	0,27	85,0%	807	216
RQRN1E	0,12	0,14	96,9%	407	97
RGRN	0,26	0,26	95,3%	298	81

Na Tabela 4.6, que avalia o desempenho dos modelos no conjunto de teste para o conjunto

de dados *Yacht*, o modelo RQRN1E continua a se destacar com perda quantílica e MADECP consideravelmente inferiores aos demais modelos. O RQRN1E também obteve valores competitivos para RMSE e MAE, de 1,34 e 0,62 para um quantil fixo de 0,5 (mediana), respectivamente. Neste conjunto de dados, a RGRN teve desempenho inferior até mesmo na métrica RMSE, possivelmente devido à acentuada assimetria dos dados e a penalização maior da perda quadrática aos desvios maiores. Apesar da cobertura de 95% ser aproximadamente 93%, uma avaliação de calibração local é necessária.

Tabela 4.6: Desempenho dos modelos no conjunto de teste no banco *Yacht Hydrodynamics*.

Modelos	RMSE	MAE	Perda Quantílica	Cobertura (95%)	MADECP
RQRN2E	1,93	0,98	0,31	85,7%	0,106
RQRN1E	1,34	0,62	0,14	82,1%	0,028
RGRN	1,82	0,59	0,25	92,9%	0,117

Por fim, o modelo RQRN2E teve a maior perda quantílica no teste (0,31) e uma taxa de cobertura de 85,7%. A MADECP do RQRN2E é de 0,106, que está entre as maiores, apontando para uma calibração menos precisa para os quantis.

Capítulo 5

Conclusões

Nesta dissertação, desenvolvemos e avaliamos uma nova arquitetura para a Regressão Quantílica Multitarefa via Redes Neurais Profundas, denominada RQRN1E, que visa superar limitações dos modelos existentes, como a Regressão Quantílica Multitarefa via Redes Neurais Profundas em 2 Estágios (RQRN2E, Kuleshov e Deshpande (2022)) e a tradicional Regressão Gaussiana via Redes Neurais Profundas (RGRN). O principal objetivo foi aprimorar a estimativa de quantis e a capacidade de modelar a distribuição completa dos dados. Para a comparação entre os modelos, foram utilizados três conjuntos de dados distintos: *Diamonds*, *Boston* e *Yacht Hydrodynamics*.

A análise dos resultados dos dados *Diamonds* confirmou a eficácia da arquitetura RQRN1E, destacando-se pela menor perda quantílica, boa taxa de cobertura e um valor de MADECP mais baixo, indicando melhor consistência entre os quantis preditos e a distribuição empírica. Em relação as RMSE e MAE, que priorizam a média e a mediana individualmente, o RQRN1E ofereceu um bom desempenho para um quantil fixo de 0,5 (mediana) e um desempenho superior na captura da variabilidade dos quantis. Esse desempenho foi atribuído ao fato de apresentar uma arquitetura unificada para estimação dos quantis em um único estágio e à introdução do logito do τ na penúltima camada intermediária da rede, aprimorando a abordagem em dois estágios de Kuleshov e Deshpande (2022).

Nos dados de *Boston*, o RQRN1E continuou a se destacar, com os melhores resultados em termos de perda quantílica, cobertura de quantis e MADECP, além de uma rápida convergência e menor tempo computacional, reforçando sua eficácia e aplicabilidade. Nos dados de *Yacht Hydrodynamics*, o modelo mostrou robustez na modelagem de quantis, embora com uma leve queda na taxa de cobertura em relação ao RGRN.

A comparação entre os modelos evidenciou que o RQRN1E, além de ser eficiente graças à sua arquitetura simplificada, demonstrou uma boa capacidade para modelar relações não lineares complexas dos quantis da distribuição condicional da variável resposta. Embora o RGRN tenha obtido resultados ligeiramente melhores em métricas de erro da média e mediana, como RMSE e MAE, o desempenho do RQRN1E na estimativa de quantis, medido principalmente pelas métricas de perda quantílica e MADECP, foi mais eficaz. Isso reforça a adequação do RQRN1E para as tarefas de regressão quantílica e calibração dos intervalos de confiança.

Para trabalhos futuros, sugere-se a implementação de um ajuste automatizado dos hiperparâmetros das redes, avaliando o custo computacional dessa tarefa, a inclusão de estudos de simulação para explorar nuances adicionais (como a escolha da camada de inserção do vetor τ) e a exploração de outras métricas de comparação entre os modelos (por exemplo, avaliação da calibração local). A exploração desses temas é relevante para uma continuidade da pesquisa, mas os resultados aqui apresentados sugerem que o modelo proposto apresenta potencial significativo. O trabalho correlato de Amorim (2024), no contexto de Redes Neurais com Grafos, também mostrou ganhos expressivos dessas (e outras) modificações na arquitetura da Regressão Quantílica via Redes Neurais.

Referências Bibliográficas

- Amorim, William Edward Rappel de (2024). “Positional Encoder Graph Quantile Neural Networks for Geographic Data”. Master’s Dissertation. Brasília: University of Brasília (UnB), Institute of Exact Sciences, Department of Statistics.
- Bengio, Y., Simard, P. e Frasconi, P. (1994). “Learning long-term dependencies with gradient descent is difficult”. *IEEE Transactions on Neural Networks* 5.2, pp. 157–166.
- Cannon, A. J. (2011). “Quantile regression neural networks: Implementation in R and application to precipitation downscaling”. *Computers & Geosciences* 37.9, pp. 1277–1284.
- (2018). “Non-crossing nonlinear regression quantiles by monotone composite quantile regression neural network, with application to rainfall extremes”. *Stochastic Environmental Research and Risk Assessment* 32.11, pp. 3207–3225.
- Chai, Tianfeng e Draxler, Roland R. (2014). “Root mean square error (RMSE) or mean absolute error (MAE)?—Arguments against avoiding RMSE in the literature”. *Geoscientific Model Development* 7.3, pp. 1247–1250.
- Davison, A. C. e Hinkley, D. V. (1997). *Bootstrap Methods and their Application*. Cambridge: Cambridge University Press.
- DSA (2022). *Deep Learning Book*. Em Português, Online e Gratuito. <https://www.deeplearningbook.com.br/o-que-sao-redes-neurais-artificiais-profundas>.
- Efron, B. e Tibshirani, R. (1993). *An Introduction to the Bootstrap*. Chapman e Hall/CRC.
- Efron, Bradley e Tibshirani, Robert J. (1994). *An Introduction to the Bootstrap*. CRC press.

- Faustine, A. et al. (2022). “UNet-NILM: A Deep Neural Network for Multi-tasks Appliances State Detection and Power Estimation in NILM”. *Virtual Event, Japan*.
- Flood, I. e Kartam, N. (1994). “Neural Networks in Civil Engineering I: Principles and Understanding”. *ASCE Journal of Computing in Civil Engineering*.
- Gan, D. et al. (2018). “Embedding based quantile regression neural network for probabilistic load forecasting”. *Journal of Modern Power Systems and Clean Energy* 6.2, pp. 244–254. DOI: [10.1007/s40565-018-0380-x](https://doi.org/10.1007/s40565-018-0380-x).
- Gharat, S. (2019). *O quê, por que e qual? Funções de Ativação*. <https://medium.com/@snehalgharat>.
- Goh, A. T. C. (1995). “Back-propagation neural networks for modeling complex systems”. *Artificial Intelligence in Engineering* 9.3, pp. 143–151. DOI: [10.1016/0954-1810\(94\)00011-S](https://doi.org/10.1016/0954-1810(94)00011-S).
- Goodfellow, I., Bengio, Y. e Courville, A. (2017). *Deep Learning*. MIT Press.
- Géron, A. (2019). *Mãos à Obra: Aprendizado de Máquina com Scikit-Learn, Keras e TensorFlow: Conceitos, Ferramentas e Técnicas Para a Construção de Sistemas Inteligentes*. 2ª ed. Alta Books.
- Hao, R. et al. (2023). *DAmcqrnn: An approach to censored monotone composite quantile regression neural network estimation*. School of Statistics e Mathematics, Zhejiang Gongshang University; Collaborative Innovation Center of Statistical Data Engineering Technology & Application, Zhejiang Gongshang University.
- Hastie, Trevor, Tibshirani, Robert e Friedman, Jerome (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media.
- Haykin, S. (2001). *Redes Neurais - Princípios e Práticas*. 2ª ed. BOOKMAN.
- He, X. e Hu, F. (2002). “Markov Chain Marginal Bootstrap”. *Journal of the American Statistical Association*.

- Jantre, S.R., Bhattacharya, S. e Maiti, T. (2021). “Quantile Regression Neural Networks: A Bayesian Approach”. *Journal of Statistical Theory and Practice* 15.1. DOI: [10.1007/s42519-021-00189-w](https://doi.org/10.1007/s42519-021-00189-w).
- Koenker, R. (2005). *Quantile Regression*. 1ª ed. Vol. 1. Cambridge University Press, p. 349.
- Koenker, R. e Bassett, G. (1978). “Regression quantiles”. *Econometrica* 46.
- Kuleshov, Volodymyr e Deshpande, Shachi (2022). “Calibrated and Sharp Uncertainties in Deep Learning via Density Estimation”. *ICML*, pp. 11683–11693.
- Lawrence, R. (1997). *Using Neural Networks to Forecast Stock Market Prices*.
- McCulloch, W. e Pitts, W. (1943). “A Logical Calculus of the Ideas Immanent in Nervous Activity”. *The Bulletin of Mathematical Biophysics* 5.4, pp. 115–133.
- Moon, S. J. et al. (2021). “Learning multiple quantiles with neural networks”. *Journal of Computational and Graphical Statistics* 30.4, pp. 1238–1248. DOI: [10.1080/10618600.2021.1915819](https://doi.org/10.1080/10618600.2021.1915819).
- Oleskovicz, M., Coury, D. V. e Aggarwal, R. K. (2003). “O emprego de Redes Neurais Artificiais na detecção, classificação e localização de faltas em linhas de transmissão”. *Revista Controle e Automação* 14.2, pp. 138–150.
- Pan, W.T., Huang, C.E. e Chiu, C.L. (2016). “Study on the performance evaluation of online teaching using the quantile regression analysis and artificial neural network”. *Journal of Supercomputing* 72.3, pp. 789–803. DOI: [10.1007/s11227-015-1599-1](https://doi.org/10.1007/s11227-015-1599-1).
- Pradeepkumar, D. e Ravi, V. (2017). *Forecasting financial time series volatility using Particle Swarm Optimization trained Quantile Regression Neural Network*. Center of Excellence in Analytics, Institute for Development, Research in Banking Technology (IDRBT), Castle Hills, Masab Tank, Hyderabad 500 057, India; School of Computer e Information Sciences (SCIS), University of Hyderabad, Hyderabad 500 046, India.
- Rodrigues, Filipe e Pereira, Francisco C (2020). “Beyond expectation: Deep joint mean and quantile regression for spatiotemporal problems”. *IEEE transactions on neural networks and learning systems*.

- Rousseeuw, Peter J. e Leroy, Annick M. (2003). *Robust Regression and Outlier Detection*. John Wiley & Sons.
- Si, Phillip, Kuleshov, Volodymyr e Bishop, Allan (2022). “Autoregressive Quantile Flows for Predictive Uncertainty Estimation”. *International Conference on Learning Representations*.
- Spörl, C., Castro, E. G. e Luchiari, A. (2011). “Aplicação de Redes Neurais Artificiais na construção de modelos de fragilidade ambiental”. *Revista do Departamento de Geografia* 21.1, pp. 113–135.
- Srivastava, N. et al. (2014). “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”. *Journal of Machine Learning Research* 15.56, pp. 1929–1958. URL: <http://jmlr.org/papers/v15/srivastava14a.html>.
- Taylor, J.W. (2000). “A quantile regression neural network approach to estimating the conditional density of multiperiod returns”. *Journal of Forecasting* 19.4, pp. 299–311.
- Tibshirani, R et al. (2017). *An Introduction to Statistical Learning with Applications in R*. Springer Science+Business Media New York.
- Torres, R. et al. (2024). “Model-Free Local Recalibration of Neural Networks”. *arXiv preprint arXiv:2403.05756v1*.
- Vogado, L. et al. (2019). “Rede Neural Convolutacional para o Diagnóstico de Leucemia”. *Anais do XIX Simpósio Brasileiro de Computação Aplicada à Saúde*. SBC, pp. 46–57. DOI: [10.5753/sbcas.2019.6241](https://doi.org/10.5753/sbcas.2019.6241).
- Willmott, Cort J. e Matsuura, Kenji (2005). “Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance”. *Climate Research* 30.1, pp. 79–82.
- Xu, Q., Deng, K., Jiang, C. et al. (2017). “Composite quantile regression neural network with applications”. *Expert Systems with Applications* 76, pp. 129–139.
- Zhang, W., Quan, H. e Srinivasan, D. (2019). “An Improved Quantile Regression Neural Network for Probabilistic Load Forecasting”. *IEEE Transactions on Smart Grid* 10.4, pp. 4425–4434. DOI: [10.1109/TSG.2018.2859749](https://doi.org/10.1109/TSG.2018.2859749).

Şengönül, E. et al. (2023). “An Analysis of Artificial Intelligence Techniques in Surveillance Video Anomaly Detection: A Comprehensive Survey”. *Applied Sciences* 13.8, p. 4956.