UNIVERSIDADE FEDERAL DO CEARÁ

CENTRO DE TECNOLOGIA

DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA

CURSO DE GRADUAÇÃO EM ENGENHARIA DE TELECOMUNICAÇÕES

JOÃO PEDRO SILVA CAMPOS

INTELIGÊNCIA COMPUTACIONAL COMO SUPORTE À TOMADA DE DECISÃO NO GERENCIAMENTO DE CAPACIDADE DE SERVIÇOS DE COMPUTAÇÃO EM NUVEM

FORTALEZA

2022

JOÃO PEDRO SILVA CAMPOS

INTELIGÊNCIA COMPUTACIONAL COMO SUPORTE À TOMADA DE DECISÃO NO GERENCIAMENTO DE CAPACIDADE DE SERVIÇOS DE COMPUTAÇÃO EM NUVEM

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Telecomunicações do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Telecomunicações.

Orientador: Prof. Dr. Alberto Sampaio Lima.

FORTALEZA

2022

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

C213i Campos, João Pedro Silva.
Inteligência Computacional como Suporte à Tomada de Decisão no Gerenciamento de Capacidade de Serviços de Computação em Nuvem / João Pedro Silva Campos. – 2022.
45 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Centro de Tecnologia, Curso de Engenharia de Telecomunicações, Fortaleza, 2022.
Orientação: Prof. Dr. Alberto Sampaio Lima.

1. Gerenciamento da capacidade. 2. Computação em nuvem. 3. Inteligência computacional. 4. Tomada de decisão. I. Título.

CDD 621.382

JOÃO PEDRO SILVA CAMPOS

INTELIGÊNCIA COMPUTACIONAL COMO SUPORTE À TOMADA DE DECISÃO NO GERENCIAMENTO DE CAPACIDADE DE SERVIÇOS DE COMPUTAÇÃO EM NUVEM

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Telecomunicações do Centro de Tecnologia da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Engenharia de Telecomunicações.

Aprovada em: 16/12/2022.

BANCA EXAMINADORA

Prof. Dr. Alberto Sampaio Lima (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Kléber Zuza Nóbrega
Universidade Federal do Ceará (UFC)

Prof. Dr. Wladimir Araújo Tavares
Universidade Federal do Ceará (UFC)

À minha família, por sua capacidade de acreditar em mim, investir em mim e dar todo o suporte para seguir em frente.

AGRADECIMENTOS

Ao Prof. Dr. Alberto Sampaio Lima, pela excelente orientação.

Aos professores participantes da banca examinadora Prof. Dr. Kléber Zuza Nóbrega e Prof. Dr. Wladimir Araújo Tavares pelo tempo, pelas valiosas colaborações e sugestões.

Aos colegas da turma da graduação, pela troca de conhecimento, pela disponibilidade em ajudar, pelo tempo, pelas valiosas colaborações e sugestões.

Aos familiares e aos amigos, pela contribuição e participação indireta no equilíbrio entre estudos, trabalho e as outras áreas da vida.

À equipe Avoante Aeromec, à equipe Siará Baja e a todos seus membros, por todas as experiências e aprendizados que me despertaram o gosto por liderança, trabalho em equipe e gerenciamento.

"Uma decisão exige julgamentos tanto preditivos como avaliativos." (KAHNEMAN; SIBONY; SUNSTEIN, 2021, p. 57.)

RESUMO

Os serviços de computação em nuvem mudaram o estilo de vida das pessoas e o uso da computação em nuvem tem aumentado devido sua conveniência, possibilidade de redução de custos e grande variedade de aplicações integradas. Por outro lado, as empresas que fornecem este tipo de serviço tem a necessidade de atender a crescente demanda com alta qualidade de serviço e fazer o gerenciamento da capacidade alinhado ao estratégico de forma mais rápida e objetiva. A alternativa para auxiliar o gerenciamento é o uso da inteligência computacional e por isso o presente trabalho tem como objetivo propor e testar uma solução por meio de um sistema que faça a previsão do uso dos recursos do servidor e o método utilizado é o teste e a avaliação de modelos de regressão e previsão aplicados a um conjunto de dados sequenciais gerados artificialmente com base em uma pesquisa na literatura que identifica as características destes dados. Como resultados, uma rede neural sem realimentação produziu um modelo que tem ótimo ajuste aos dados teste do pico de uso diário de processador e conclui-se que os gerentes podem utilizar os dados de previsão de uso de processador como ferramenta na tomada de decisão do gerenciamento de capacidade.

Palavras-chave: gerenciamento da capacidade; computação em nuvem; inteligência computacional; tomada de decisão.

ABSTRACT

Cloud computing services have changed people's lifestyles and the use of cloud computing has increased due to its convenience, possibility of cost reduction and wide variety of integrated applications. On the other hand, companies that provide this type of service need to meet the growing demand with high quality of service and manage capacity in line with the strategy in a faster and more objective way. The alternative to help management is the use of computational intelligence and therefore the present work aims to propose and test a solution through a system that predicts the use of server resources and the method used is the test and evaluation of regression and prediction models applied to a set of artificially generated sequential data based on a literature search that identifies the characteristics of these data. As a result, a neural network without feedback produced a model that has a great fit to the test data of the peak daily use of the processor and it is concluded that managers can use the forecast data of processor use as a tool in making management decisions of capacity.

Keywords: capacity management; cloud computing; computational intelligence; decision making.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

R^2	*Coefficient of Determination* / Coeficiente de Determinação
AWGN	*Additive White Gaussian Noise* / Ruído Branco Gaussiano Aditivo
CPU	*Central Processing Unit* / Unidade Central de Processamento
FNN	*Feed-forward Neural Network* / Rede Neural Direta
IA	Inteligência Artificial
ML	*Machine Learning* / Aprendizado de Máquina
MSE	*Mean Squared Error* / Erro Quadrático Médio
NN	*Neural Network* / Rede Neural
QoS	*Quality of Service* / Qualidade de Serviço
ReLU	*Rectified Linear Unit*
RMSE	*Root Mean Squared Error* / Raiz Quadrada do Erro Quadrático Médio
RSS	*Residual Sum of Squares* / Soma dos Quadrados dos Resíduos
TCC	Trabalho de Conclusão de Curso
TI	Tecnologia da Informação
TSS	*Total Sum of Squares* / Soma dos Quadrados Totais

SUMÁRIO

1 INTRODUÇÃO

Com o advento da ampliação e da crescente demanda das redes de comunicações móveis, dos serviços de nuvem, de internet e dos demais serviços de Tecnologia da Informação (TI), há a necessidade de suprir a demanda do mercado com *Quality of Service* / Qualidade de Serviço (QoS), principalmente dos serviços de computação em nuvem, porque a tecnologia de computação em nuvem mudou o estilo de vida das pessoas, ajudou a melhorar a conveniência e a qualidade de vida das pessoas e criou um futuro melhor. Seu surgimento é um retrato realista de "a tecnologia muda a vida e muda o futuro". No entanto, a própria tecnologia de computação em nuvem integra uma variedade de tecnologias de informação e é relativamente complexa (HUAWEI, 2023).

Atender as demandas dos clientes ou usuários com alta qualidade de serviço (necessidade de parecer elástico ou infinito para o cliente) e atender a necessidade de fazer gerenciamento mais rápido, objetivo e de forma automática exige práticas de gestão de serviços TI cada vez mais automatizadas e com tomada de decisões cada vez mais rápidas (IEEE PRESS, 2021). A gestão de serviços TI exige os respectivos fundamentos teóricos e práticos para sua execução. Desde questões relacionadas a boas práticas e qualidade de serviço TI até aspectos técnicos relacionados ao serviço TI em específico. Além disso, os gerentes se veem em situações de alinhar essas decisões, relacionadas ao gerenciamento da capacidade, com o planejamento estratégico de forma cada vem menos subjetiva, dependendo menos da experiência e *know how* para assim trazer mais eficiência e eficácia ao processo de tomada de decisão.

Portanto é necessário soluções que viabilizem o gerenciamento de serviços TI com rapidez e qualidade, sendo a Inteligência Artificial (IA) e algoritmos de *Machine Learning* / Aprendizado de Máquina (ML) alternativas que oferecem infinitas possibilidades. É esperado que a utilização dessas soluções fornecerão uma compreensão mais profunda e uma melhor tomada de decisão com base em dados operacionais amplamente coletados e disponíveis e também apresentarão oportunidades para melhorar os algoritmos e métodos de análise de dados em aspectos como precisão, escalabilidade e generalização (IEEE PRESS, 2021).

O objetivo principal do Trabalho de Conclusão de Curso (TCC) é propor e testar uma solução utilizando inteligência computacional para aumentar a objetividade, rapidez, eficiência e eficácia do processo de tomada de decisão no gerenciamento de capacidade de serviços de computação em nuvem. E por isso este projeto tem como objetivo secundário estudar as aplicações da IA como suporte à tomada de decisão no gerenciamento de capacidade de serviços

de computação em nuvem e a partir desse estudo desenvolver um *software* que faz a previsão do uso de *Central Processing Unit* / Unidade Central de Processamento (CPU) no servidor através de dados históricos.

A metodologia é a pesquisa de literatura e a geração do banco de dados com uma série temporal artificial para o desenvolvimento de um sistema para auxiliar na tomada de decisão no gerenciamento da capacidade de serviços de computação em nuvem. Para esse sistema são testados métodos de regressão linear e uma simples rede neural direta que obtém ótimos resultados quando aplicada para previsão em um conjunto de dados gerados artificialmente.

O trabalho está dividido em fundamentação teórica com os conhecimentos necessários para entender os métodos aplicados. Em seguida tem a seção de métodos em que são descritos os métodos aplicados no trabalho para gerar os resultados. Logo depois no texto está os resultados que possuem a análise dos principais dados gerados a partir do TCC. E o texto da monografia finaliza com a seção de conclusão que traz a essência do que o TCC gerou e do que pode ser feito no futuro para aprimorá-lo.

2 FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

Existem muitas definições para serviços e em comum elas descrevem serviços como atividades que entregam valor para o cliente, no papel de co-produtor, e são caracterizadas pela intangibilidade, inseparabilidade, heterogeneidade e perecibilidade (FITZSIMMONS, 2014). Logo o gerenciamento de serviços é um conjunto de capacidades organizacionais especializadas em gerar valor aos clientes no formato de serviços (OFFICE OF GOVERNMENT COMMERCE, 2011).

No caso dos serviços de computação em nuvem, eles são descritos como as atividades que geram valor para o cliente com um modelo para permitir acesso de rede onipresente, conveniente e sob demanda a um conjunto compartilhado de recursos de computação configuráveis (por exemplo, redes, servidores, armazenamento, aplicativos e serviços) que podem ser rapidamente provisionados e liberados com esforço mínimo de gerenciamento ou interação do provedor de serviços (MELL, 2011).

É importante ressaltar que em serviços, deve-se distinguir entre insumos e recursos. Para a indústria de serviços, insumos são os próprios clientes e recursos são os bens facilitadores (FITZSIMMONS, 2014) que no caso da computação em nuvem são os componentes de *hardware* e de *software*. Além disso, Fitzsimmons (2014) diz que para funcionar, o sistema de serviços deve interagir com os clientes como se estes fossem participantes do processo. Considerando que os clientes aparecem conforme sua própria vontade e conforme demandas únicas em relação ao sistema de serviços, combinar a capacidade do serviço com a demanda é um desafio.

Técnicas de tomada de decisão são usadas para selecionar um curso de ação a partir de diferentes alternativas (PMI, 2017). Segundo (OFFICE OF GOVERNMENT COMMERCE, 2011), o gerenciamento de capacidade fornece as informações necessárias sobre a utilização de recursos atuais e planejadas de componentes individuais para permitir que as organizações decidam com confiança:

- Quais componentes atualizar - mais memória, dispositivos de armazenamento mais rápidos, processadores mais rápidos, maior largura de banda;
- Quando atualizar - idealmente, isso não é muito cedo, resultando em excesso de capacidade caro, nem muito tarde, deixando de aproveitar os avanços em novas tecnologias, resultando em gargalos, desempenho inconsistente e, em última análise, insatisfação do cliente e perda de oportunidades de negócios;
- Quanto custará a atualização - os elementos de previsão e planejamento do gerenciamento

de capacidade alimentam os ciclos de vida orçamentários, garantindo o investimento planejado.

O planejamento de capacidade é a atividade de criar um plano que gerencia recursos para atender a demanda de serviços e o objetivo da prática de gerenciamento de capacidade e desempenho é garantir que os serviços atinjam o desempenho acordado e esperado, satisfazendo a demanda atual e futura de maneira econômica (OFFICE OF GOVERNMENT COMMERCE, 2011). As decisões sobre capacidade nos serviços têm uma importância estratégica baseada no período de tempo em questão. Como a capacidade física (isto é, instalações e equipamentos) é adicionada em unidades discretas, a possibilidade de adequação da capacidade à demanda é infrutífera, e uma estratégia de construir prevendo a demanda futura muitas vezes é utilizada para evitar a perda de clientes (FITZSIMMONS, 2014).

A capacidade é determinada pelos recursos disponíveis para a organização. O planejamento de capacidade é o processo de definição dos tipos e montantes de recursos exigidos para implementar o plano estratégico de negócios de uma organização. O objetivo do planejamento estratégico de capacidade é determinar o nível adequado da capacidade de atendimento ao especificar o *mix* apropriado de instalações, equipamentos e mão de obra necessários para atender à demanda prevista. O planejamento de capacidade é um desafio para as empresas de serviços devido à natureza de sistemas abertos das operações de serviços e, desse modo, à impossibilidade de criar um fluxo estável de atividade para utilizar totalmente a capacidade (FITZSIMMONS, 2014).

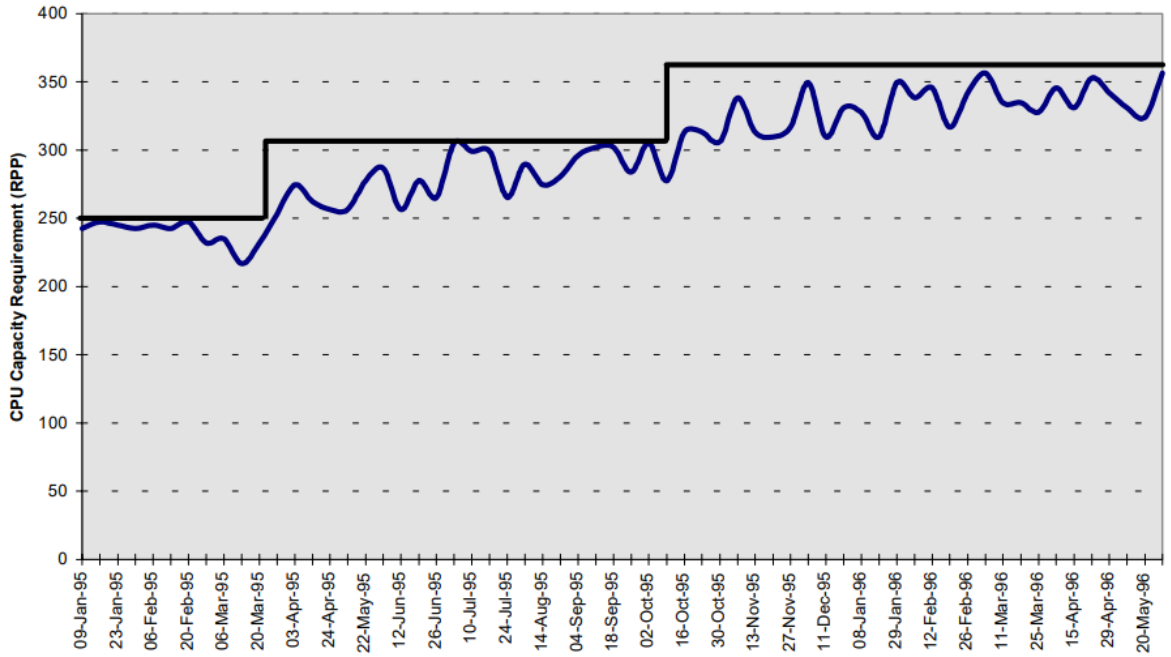
2.1 Gerenciamento de Serviço de Computação em Nuvem

O planejamento da capacidade pode ser considerado como um método de previsão do uso de hardware de forma que tenha menos impacto sobre a qualidade do serviço com otimização dos custos. Dentre os recursos de hardware que são comumente planejados, CPU é geralmente considerado como o mais importante (NEIL MCMENEMY THE ROYAL BANK OF SCOTLAND PLC, 1996). Além disso, o processo geral de gerenciamento de capacidade está continuamente tentando adequar os recursos e a capacidade de TI de maneira econômica às necessidades e requisitos do negócio em constante mudança. Isso requer o ajuste (ou "otimização") dos recursos atuais e a estimativa e planejamento efetivos dos recursos futuros (OFFICE OF GOVERNMENT COMMERCE, 2011). Por isso o serviço de computação em nuvem, como tipo de serviço de TI, para o gerenciamento efetivo precisa fazer a previsão da demanda.

QoS geralmente especifica um entendimento comum sobre responsabilidades, garantias, garantias, níveis de desempenho em termos de disponibilidade, tempo de resposta, etc. E Ran *et al.* (2015) considera a probabilidade de sobrecarga do serviço como uma métrica de QoS entre o provedor de serviços e usuários finais. Isso porque instâncias superprovisionadas podem resultar em uma baixa taxa de utilização e um custo desnecessário, enquanto a ativação de instâncias insuficientes resulta em um alto tempo de espera e alta taxa de espera e leva à degradação de QoS. Assim, Ran *et al.* (2015) formula o problema de provisionamento de instância dinâmica durante o tempo de execução como o cálculo da quantidade ideal de instâncias ativas sujeitas a um requisito de QoS. A medida de QoS é a probabilidade de sobrecarga de que a carga de trabalho de computação geral exceda a capacidade fornecida pelas instâncias ativas (RAN *et al.*, 2015).

De acordo com Neil McMenemy The Royal Bank of Scotland Plc (1996) o responsável pelo planejamento da capacidade concentra-se na hora de pico e, ignora todas as outras horas, e assim pode observar as tendências históricas da hora de pico. Na Figura 1 mostra uma série temporal que representa o requisito da hora de pico para capacidade de CPU.

Figura 1 – Série Temporal da Hora de Pico de CPU



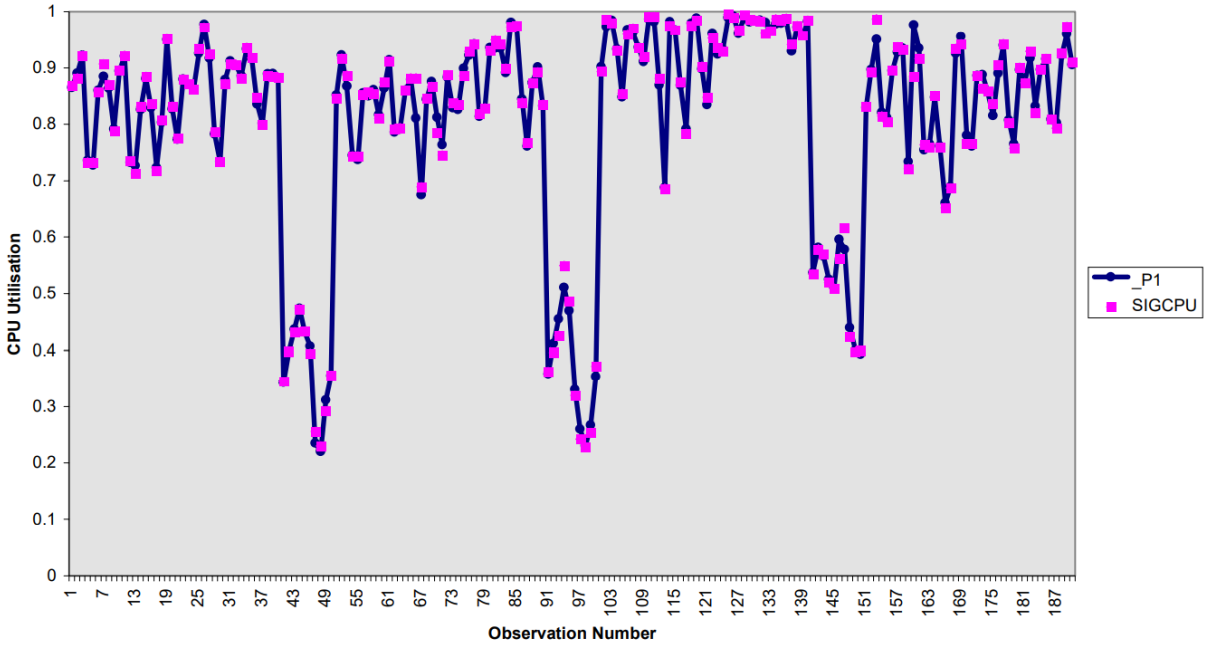
Fonte: adaptado de Neil McMenemy The Royal Bank of Scotland Plc (1996, p. 2).

A Figura 1 mostra a sazonalidade característica e padrões que se repetem com o tempo e em determinados intervalos de tempo, ou seja, com periodicidade. Além disso, durante o período mostrado, houve duas atualizações na máquina e é bastante perceptível que, assim que

uma máquina é atualizada, a utilização ultrapassa imediatamente a capacidade da máquina antes da atualização (NEIL MCMENEMY THE ROYAL BANK OF SCOTLAND PLC, 1996).

O comportamento característico de uso de CPU apresenta altos e baixos em curto período de tempo, um comportamento cíclico semanal que pode ser visto na Figura 2.

Figura 2 – Série Temporal de 21 dias da Utilização de CPU



Fonte: adaptado de Neil McMenemy The Royal Bank of Scotland Plc (1996, p. 8).

A Figura 2 representa uma visualização de 21 dias com dois comportamentos bem diferentes no que diz respeito a média dos valores, algumas vezes com a média mais baixa e a maior parte das vezes com a média mais alta.

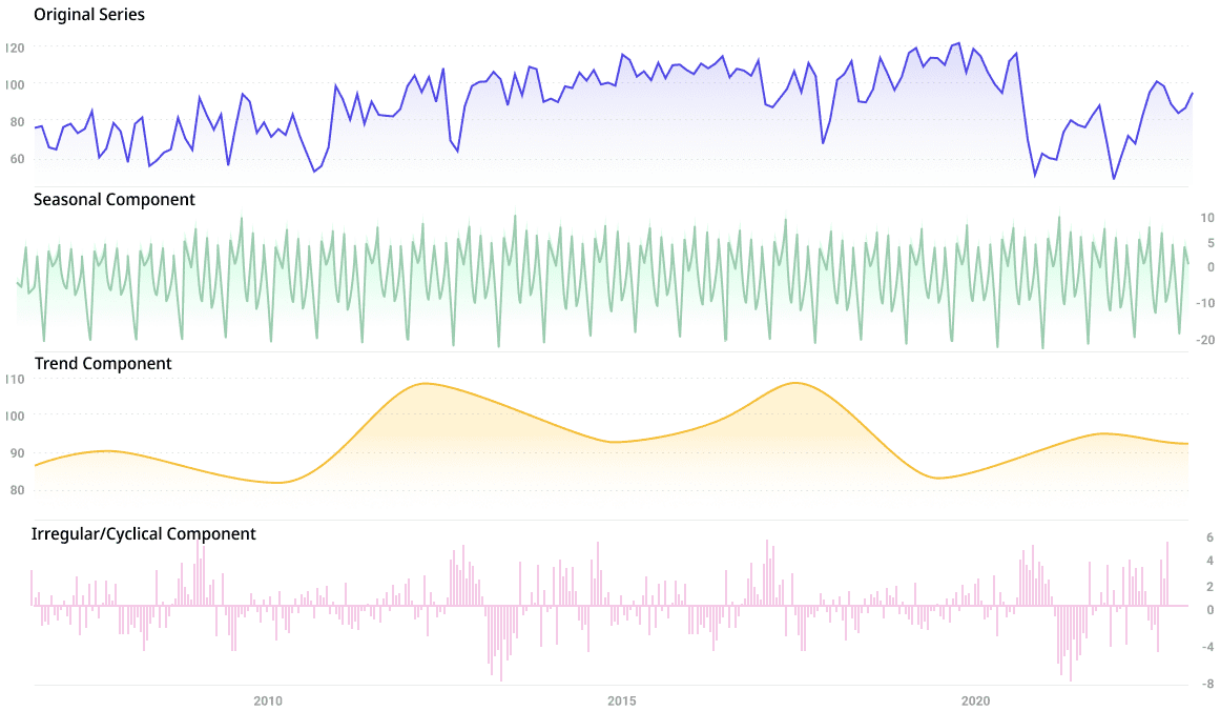
2.2 Séries Temporais

De acordo com (FITZSIMMONS, 2014) a previsão de demanda é feita a partir de métodos de modelos de séries temporais. Os modelos de séries temporais servem para fazer previsões de curto prazo quando os valores das observações ocorrem seguindo um padrão de comportamento identificável ao longo do tempo.

Os componentes de uma previsão são média, tendência e sazonalidade. A média é uma estimativa da média subjacente de uma variável aleatória (p. ex., demanda do cliente), a tendência é um incremento crescente ou decrescente em cada período, e a sazonalidade é um círculo recorrente (FITZSIMMONS, 2014). Como descrito em Molodoria (2022) a série temporal que representa a demanda pode ser representada através da composição por três componentes.

A componente cíclico irregular semanal, a componente de sazonalidade e a componente de tendência. A Figura 3 mostra a visualização da série temporal característica da demanda de um serviço e mostra a extração da componentes que formam a série original.

Figura 3 – Decomposição de uma Série Temporal



Fonte: adaptado de Molodoria (2022).

Muitas fontes de dados são de natureza sequencial e exigem tratamento especial ao criar modelos preditivos. As séries exibem autocorrelação — neste caso, os valores próximos no tempo tendem a ser semelhantes entre si e isso distingue as séries temporais de outros conjuntos de dados que encontramos, nos quais as observações podem ser consideradas independentes umas das outras (HASTIE ROBERT TIBSHIRANI, 2013).

2.3 Métodos de Previsão

No ambiente de computação em nuvem, todos os recursos de computação podem ser aumentados ou diminuídos dinamicamente da infraestrutura de hardware e podem ser expandidos e contraídos de forma flexível para atender às necessidades das tarefas de trabalho. A essência da infraestrutura de computação em nuvem é maximizar a utilização do investimento em TI, integrando e compartilhando o fornecimento dinâmico de equipamentos de hardware, o que reduz significativamente o custo unitário do uso da computação em nuvem e também é muito

propício à operação comercial de IA (HUAWEI, 2023).

Os métodos de previsão e a análise dos dados têm a necessidade da divisão do conjunto de dados em dados de entrada X e dados de saída Y. O conjunto de dados de entrada são utilizados para predição saída como na Equação 2.1.

$$\hat{Y} = f(X) + \varepsilon \tag{2.1}$$

Onde $\hat{Y}$ representa os resultados da predição do modelo e ε é o erro irredutível.

Os métodos criam modelos com maior performance e desempenho quando os dados estão previamente tratados, ou seja, condições prévias que favorecem a aplicação do método. Essas condições podem ser condições estatísticas, formato dos dados, entre outras dependendo da característica do método. Em muitos casos é necessário fazer a centralização e escalonamento dos dados X para construir o modelo de regressão sem enviesamento subtraindo a média e dividindo o resultado pelo desvio padrão resultando em média zero e variância unitária.

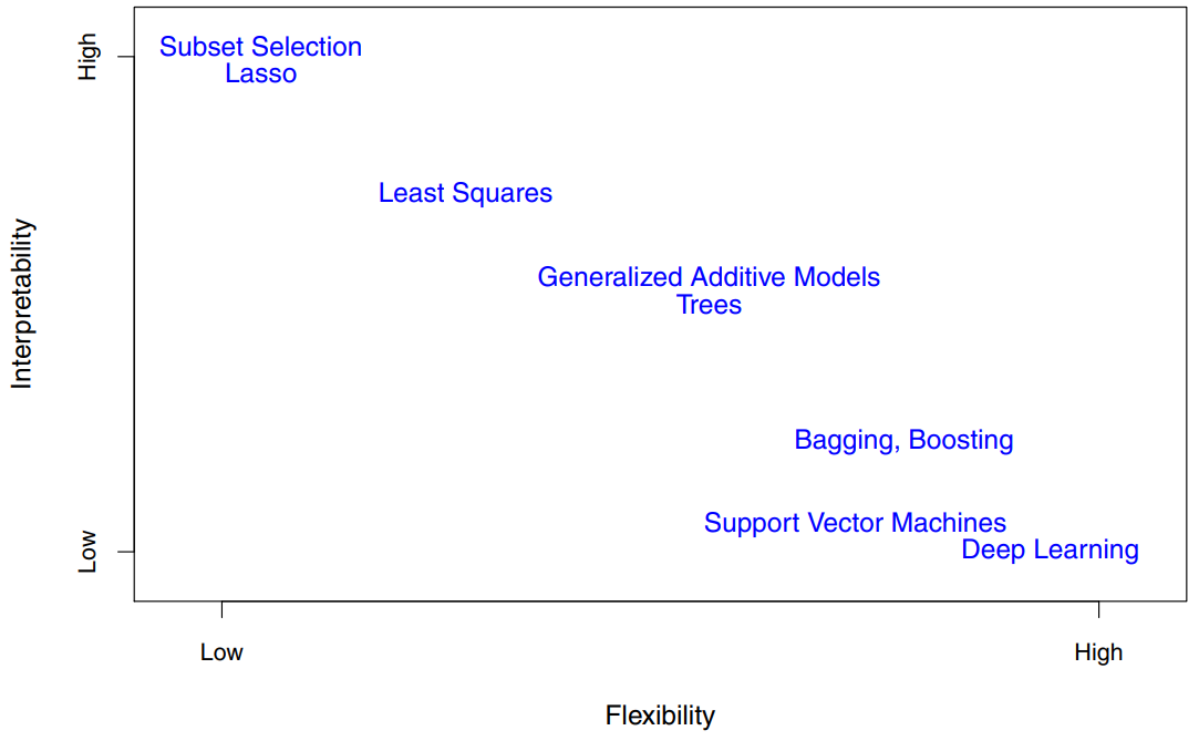
Com o pré-processamento feito em todo o conjunto de dados de entrada é necessário fazer a divisão em treino e teste e assim dividir os dados em duas partes, uma para treinar o modelo e outra para testar o modelo com dados que ele não conhece.

O melhor método de regressão quanto a performance na predição dependerá das características do banco de dados aumentando a importância da análise estatística e do pré-processamento dos dados (HASTIE ROBERT TIBSHIRANI, 2013). Além disso, segundo Hastie Robert Tibshirani (2013) em geral quando a flexibilidade de um método aumenta a sua interpretabilidade diminui. Na Figura 4 está a representação entre o compromisso, do inglês *tradeoff*, entre a interpretabilidade e a flexibilidade de um método de previsão.

A regressão tem como objetivo principal encontrar a relação entre as entradas que são as variáveis independentes e a saída que é a variável dependente. Essa relação é estimada através dos parâmetros que são calculados em um método supervisionado com a observação da saída real e da saída estimada. A diferença entre as saídas é chamada de erro sendo um problema de otimização com a minimização da função custo ou com a maximização da função de verossimilhança.

Existem vários tipos de regressão e suas performances em determinado conjunto de dados pode ser comparada através do calculo da *Root Mean Squared Error* / Raiz Quadrada do Erro Quadrático Médio (RMSE), e através do calculo do *Coefficient of Determination* / Coeficiente de Determinação (R^2). O métodos de regressão podem ser divididos em métodos

Figura 4 – Representação do Compromisso entre Interpretabilidade e Flexibilidade



Fonte: adaptado de Hastie Robert Tibshirani (2013, p. 25).

lineares e não lineares. Um exemplo de método de regressão não linear é a regressão utilizando redes neurais.

Após o treinamento, o modelo pode ter sobre-ajuste, conhecido como *overfitting*, quando tem alta variância e pode ter subajuste, conhecido como *underfitting*, quando tem alto enviesamento, conhecido como *bias*. Assim tem-se o compromisso (*trade-off*) do equilíbrio entre polarização e variância para encontrar a melhor estimativa.

Independente do método de regressão o conjunto de dados é dividido em conjunto de treino e em conjunto de teste. O conjunto de treino é usado para treinamento do modelo e estimação dos parâmetros e o conjunto de teste é usado para validação do modelo e avaliação da performance do método.

A aplicação dos métodos de previsão pode ser feito através de implementação utilizando alguma linguagem de programação no ambiente computacional escolhido ou pode ser utilizado bibliotecas prontas. Dois exemplos de bibliotecas são *Scikit-learn* e *TensorFlow*. *Scikit-learn* é uma biblioteca de aprendizado de máquina de código aberto que suporta aprendizado supervisionado e não supervisionado. Ela também fornece várias ferramentas para ajuste de modelo, pré-processamento de dados, seleção de modelo, avaliação de modelo e muitos outros utilitários (GOOGLE, 2022).

2.3.1 Regressão Linear

A regressão linear simples consiste em uma abordagem para prever uma resposta quantitativa Y dado um preditor X.

$$Y \approx \beta_0 + \beta_1 X \tag{2.2}$$

Os coeficientes β_0 e β_1 representam a interceptação e a inclinação do modelo linear. Treinando o modelo, podemos obter as estimativas $\hat{\beta}_0$ e $\hat{\beta}_1$, ou seja, a predição de Y dado $X = x$ é

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x, \tag{2.3}$$

Utilizamos os n dados de entrada e saída para descobrir os coeficientes β_0 e β_1, tal que, para n pares de observação, temos

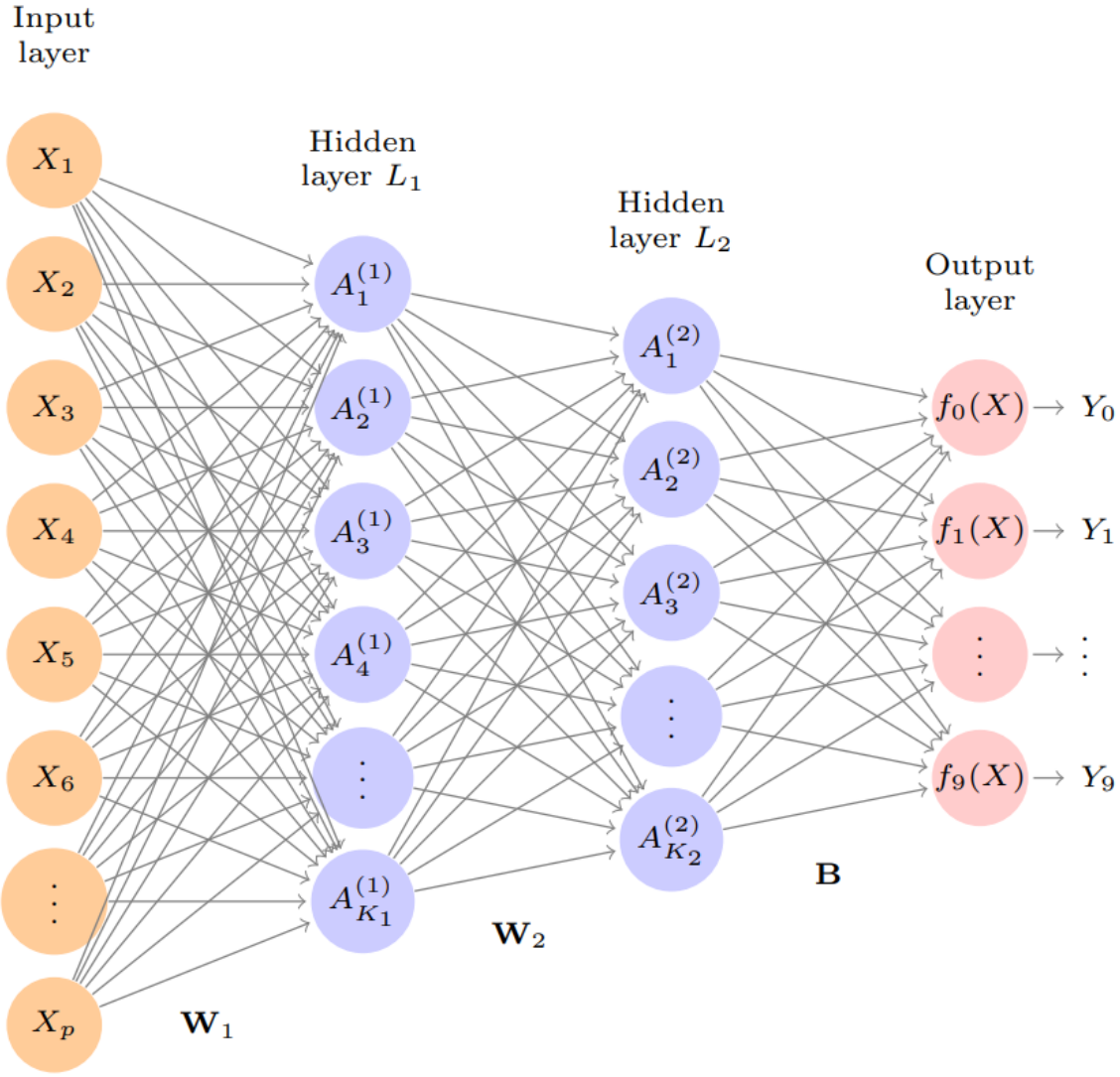
$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \tag{2.4}$$

Para $y_i \approx \beta_0 + \beta_1 x_i$, dado $i = 1, 2, \ldots, n$, deseja-se encontrar β_0 e β_1 a fim de obter uma linha resultante mais próxima possível dos n dados. No caso de mais de um preditor, essa abordagem, entretanto, não é satisfatória, pois não leva em consideração a correlação entre os preditores. Assim, expande-se o conceito de regressão linear para acomodar múltiplos preditores, dando a cada preditor um coeficiente de inclinação em um único modelo. Esse conceito é chamado de regressão linear múltipla (HASTIE ROBERT TIBSHIRANI, 2013).

2.3.2 Rede Neural Feed-forward

O aprendizado profundo, ou *deep learning* do inglês, é uma área de pesquisa muito ativa nas comunidades de aprendizado de máquina e inteligência artificial, sendo a *Neural Network* / Rede Neural (NN) a sua base e a sua essência. Uma rede neural consiste em um método para receber um vetor de entrada de p variáveis $X = (X_1, X_2, ..., X_p)$ e construir uma função não linear $f(X)$ para fazer a previsão da resposta Y. A *Feed-forward Neural Network* / Rede Neural Direta (FNN) é um tipo mais simples de NN (HASTIE ROBERT TIBSHIRANI, 2013). Na Figura 5 mostra um exemplo genérico da estrutura FNN que no caso de uma regressão tem uma única saída quantitativa, um único elemento na camada de saída, *Output layer* do inglês. A FNN também é conhecida por Rede Neural sem realimentação e os elementos da mesma camada, os neurônios, não são conectados entre si (WIKIPÉDIA, 2022). Cada neurônio da camada anterior, com o sentido de alimentação da esquerda para direita, tem conexão com

Figura 5 – Diagrama que Representa uma Rede Neural



Fonte: adaptado de Hastie Robert Tibshirani (2013, p. 409).

todos os elementos da camada posterior e isso é indicado pelas setas, assim as setas indicam as possibilidade de alimentação (HASTIE ROBERT TIBSHIRANI, 2013). As camadas entre a camada de entrada, *Input layer* do inglês, e a camada de saída são chamadas de camadas intermediárias, ocultas ou escondidas, *Hidden layer* do inglês (WIKIPÉDIA, 2022).

O modelo de FNN para saída quantitativa tem a forma

$$f(X) = \beta_0 + \sum_{l=1}^{K_2} \beta_l h_l^{(2)}(X) = \beta_0 + \sum_{l=1}^{K_2} \beta_l A_l^{(2)} \tag{2.5}$$

e para a primeira camada intermediária tem a forma de

$$A_k^{(1)} = h_k^{(1)}(X) = g\left(w_{k0}^{(1)} + \sum_{j=1}^{p} w_{kj}^{(1)} X_j\right) \tag{2.6}$$

e para a segunda camada intermediária tem a forma de

$$A_l^{(2)} = h_l^{(2)}(X) = g\left(w_{l0}^{(2)} + \sum_{k=1}^{K_1} w_{lk}^{(2)} A_k^{(1)}\right) \tag{2.7}$$

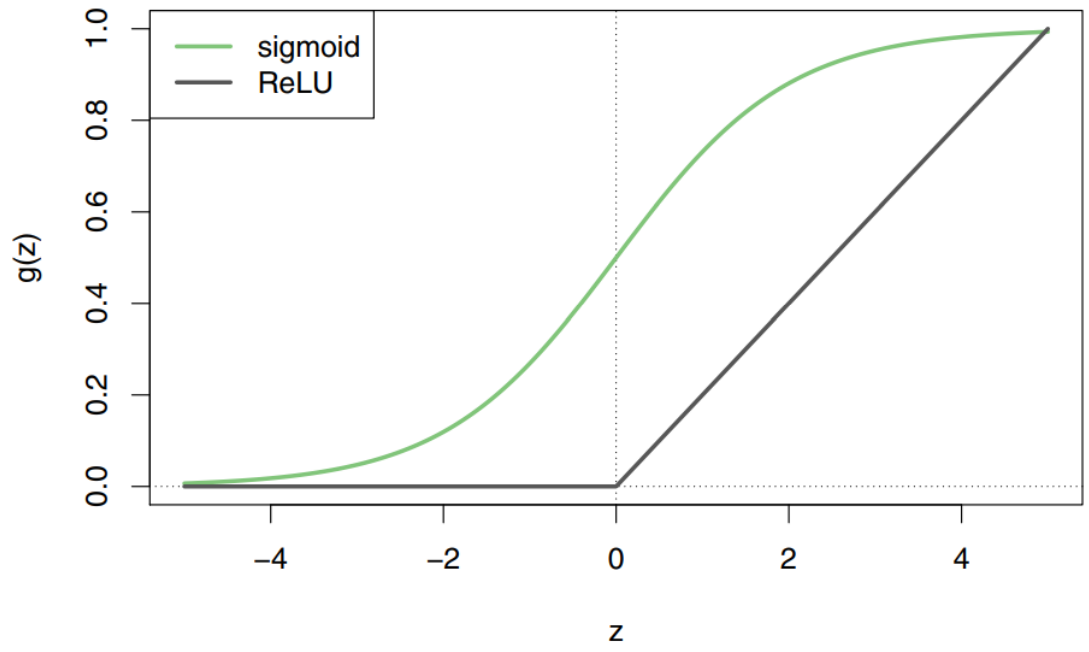
sendo $g(z)$ é uma função de ativação não linear e primeiramente era utilizada a função de ativação *sigmoid* representada pela Equação 2.8 e na Figura 6.

$$g(z) = \frac{e^z}{1+e^z} = \frac{1}{1+e^{-z}} \tag{2.8}$$

Atualmente a escolha preferida para NN moderna é a função de ativação *Rectified Linear Unit* (ReLU) que tem a forma da Equação 2.9 e está representada na Figura 6.

$$g(z) = (z)_+ = \begin{cases} 0 & \text{se } z < 0 \\ z & \text{caso contrário} \end{cases} \tag{2.9}$$

Figura 6 – Representação das Funções de Ativação Sigmoid e ReLU



Fonte: adaptado de Hastie Robert Tibshirani (2013, p. 406).

Ter uma função de ativação não linear permite que o modelo capture não linearidades complexas e efeitos de interação. As redes neurais modernas normalmente têm mais de uma camada oculta e muitas vezes muitas unidades por camada. Em teoria, uma única camada oculta com um grande número de unidades tem a capacidade de aproximar a maioria das funções. No entanto, a tarefa de aprendizado de descobrir uma boa solução é muito mais fácil com várias camadas, cada uma de tamanho modesto (HASTIE ROBERT TIBSHIRANI, 2013). Segundo

Hastie Robert Tibshirani (2017) de um modo geral, é melhor ter muitas unidades ocultas do que poucas. Com poucas unidades ocultas, o modelo pode não ter flexibilidade suficiente para capturar as não linearidades nos dados; com muitas unidades ocultas, os pesos extras podem ser reduzidos a zero se a regularização apropriada for usada. Normalmente, o número de unidades ocultas está entre 5 e 100, com o número aumentando com o número de entradas e o número de casos de treinamento. O mais comum é abaixar um número razoavelmente grande de unidades e treiná-las com regularização.

A escolha do número de camadas ocultas é guiada pelo conhecimento prévio e pela experimentação. Cada camada extrai recursos da entrada para regressão ou classificação. O uso de várias camadas ocultas permite a construção de recursos hierárquicos em diferentes níveis de resolução (HASTIE ROBERT TIBSHIRANI, 2017).

2.4 Análise dos dados

A análise dos dados pode ser feita de forma qualitativa e de forma quantitativa. Assim os resultados e os modelos criados podem ser avaliados pela performance e pelo desempenho. Para a análise dos resultados de um método é necessário medir de alguma forma o quão bom é a predição do seu modelo. A medida mais comum para essa avaliação é *Mean Squared Error* / Erro Quadrático Médio (MSE), que pode ser expresso pela Equação 2.10.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{f}(x_i)\right)^2 \tag{2.10}$$

Onde y é a saída original e $\hat{f}(x)$ é a predição feita da saída para cada n-ésima observação. Assim na Equação 2.10 se a saída predita for próxima a saída observada o MSE será pequeno e nesse caso o melhor método para determinado conjunto de dados é aquele que possui menor MSE. Porém baixo MSE no treinamento não significa baixo MSE no teste. Quando temos um método com pequeno MSE no treino, porém um grande MSE no teste isso é chamado de *overfitting* caracterizando um modelo com grande complexidade que está relacionado a encontrar relações e padrões específicos do banco de dados de treino e que podem ser resultado da aleatoriedade. O caso contrário ao descrito é denominado *underfitting*. Dito isto, a escolha dos melhores parâmetros de $\hat{f}(x)$ acontece através da otimização do erro e do compromisso entre o *bias* e a variância. No geral em métodos mais flexíveis a variância aumentará e o *bias* diminuirá.

Para confirmar as informações apresentadas acima podemos partir da Equação 2.10

e aplicar o operador linear Esperança decompondo em três termos: a variância de $\hat{f}(x_o)$, *bias* ao quadrado de $\hat{f}(x_o)$ e a variância do erro.

$$E\left\{\left(y_o-\hat{f}(x_o)\right)^2\right\}=Var\left(\hat{f}(x_o)\right)+\left(Bias\left(\hat{f}(x_o)\right)\right)^2+Var(\varepsilon) \tag{2.11}$$

Portanto podemos evidenciar e confirmar a partir da Equação 2.11 que para minimizar o valor médio do erro temos que escolher o método que simultaneamente minimize a variância e o *bias*.

A análise descrita anteriormente relacionada ao MSE pode ser estendida para o conceito de RMSE que está representado na Equação 2.12 e assim usar o RMSE para comparação de diferentes modelos, caracterizando uma análise quantitativa dos resultados.

$$RMSE=\sqrt{MSE}=\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i-\hat{f}(x_i)\right)^2} \tag{2.12}$$

Outra forma de comparar modelos e analisar a sua performance é utilizando o coeficiente de determinação R^2. O valor do coeficiente de determinação R^2 indica o quanto o modelo justifica as variações das saídas, ou seja, o quanto o nosso modelo está explicando os dados ou o quanto o modelo pode prever corretamente.

Para calcular o valor de R^2 é necessário calcular *Residual Sum of Squares* / Soma dos Quadrados dos Resíduos (RSS) na Equação 2.13

$$RSS=\sum_{i=1}^{n}(y_i-\hat{y}_i)^2 \tag{2.13}$$

O *Total Sum of Squares* / Soma dos Quadrados Totais (TSS) mede a variância total na resposta Y e pode ser considerado como a quantidade de variabilidade inerente à resposta antes que a regressão seja executada. Em contraste, o RSS mede a quantidade de variabilidade que fica sem explicação após a execução da regressão (HASTIE ROBERT TIBSHIRANI, 2013). O TSS é calculado a seguir na Equação 2.14

$$TSS=\sum_{i=1}^{n}(y_i-\overline{y})^2 \tag{2.14}$$

onde

$$\overline{y}=\frac{1}{n}\sum_{i=1}^{n}y_i \tag{2.15}$$

Para saber o quanto o nosso modelo está explicando os dados, ou o quanto o modelo pode prever corretamente, utiliza-se a medida R^2 na Equação 2.16.

$$R^2=\frac{TSS-RSS}{TSS}=1-\frac{RSS}{TSS} \tag{2.16}$$

A medida R^2 geralmente apresenta valores de 0 a 1 e quando o modelo prever corretamente, ou seja, a acuracidade do modelo é alta, o valor de R^2 deve se aproximar de 1.

3 METODOLOGIA

A metodologia utilizada pode ser classificada como descritiva e o método utilizado é a pesquisa bibliográfica que conta com a revisão da literatura relacionada ao gerenciamento de serviços TI, aos serviços de computação em nuvem e a inteligência computacional.

A metodologia também pode ser classificada como associativa porque utiliza simulação computacional para geração de um banco de dados no formato de uma série temporal do uso de CPU e testa métodos para fazer a previsão dos picos diários de uso de processamento dos servidores de computação em nuvem. Métodos estes que utilizam algoritmos matemáticos para estimar parâmetros que relacionam as amostras com o tempo e as amostras com elas mesmas.

Os métodos de simulação computacional produzem modelos matemáticos e computacionais utilizando linguagem de programação em Python no ambiente computacional do Google Colab acessado por computador pessoal.

Todos os recursos necessários ao projeto como o computador, software para simulação e referências bibliográficas são viabilizados pelo aluno e pelo orientador.

Os métodos seguem uma sequência de execução para gerarem os resultados e todas as fases e etapas do trabalho podem ser descritas como:

1. Pesquisa bibliográfica e revisão da literatura;
2. Caracterização e escolha do banco de dados;
   - Caracterização e procura do banco de dados;
   - Geração e avaliação dos dados característicos;
3. Aplicação dos métodos de regressão ou previsão;
4. Análise dos dados;
   - Avaliação dos métodos e modelos produzidos;
   - Previsão e avaliação dos próximos dias;
5. Escrita da monografia.

3.1 Pesquisa bibliográfica

A pesquisa bibliográfica tem o foco em livros e guias de gerenciamento de TI, além de trabalhos relacionados a computação em nuvem em formato de artigo. Primeiramente a pesquisa bibliográfica é focada em nomenclaturas, definição de terminologias e conhecimentos teóricos a respeito do tema, depois é uma revisão da literatura e trabalhos relacionados pes-

quisados através das palavras-chave encontradas no aprendizado do conhecimento teórico. O direcionamento também acontece através da busca por artigos mencionados no trabalho estudado e disponibilizado pelo professor Orientador chamado *Business-Driven Support for Infrastructure as a Service Capacity Management Through System Dynamics Simulations* (FENNER *et al.*, 2021) e que estão disponíveis de forma gratuita na *internet*.

Os livros sobre inteligência computacional aplicada pesquisados também estão disponíveis gratuitamente na *internet* e os programas e algoritmos que servem de base para o desenvolvimento da parte prática do trabalho são disponibilizados através de dois vídeos no *YouTube* sobre uma introdução para previsão de séries temporais e aprendizado de máquina para gerenciamento da capacidade.

3.2 Banco de dados

A fim de atingir os objetivos do trabalho é necessário adquirir um banco de dados relacionado com o gerenciamento de serviços de computação em nuvem, pois o banco de dados em questão faz parte do sujeito do estudo. Porém a pesquisa bibliográfica e a revisão dos trabalhos relacionados ao tema não foram suficientes para encontrá-lo. Além disso, plataformas de compartilhamento de conjunto de dados e algoritmos, como o *Kaggle*, só oferecem dados para regressão e previsão da demanda de outros setores de serviços que não estão relacionados com a computação em nuvem. Então não foi possível utilizar um banco de dados real, fazer um estudo de caso ou utilizar qualquer outro tipo de banco de dados publico como previsto inicialmente.

O gerenciamento da capacidade conta com o processo de planejamento da capacidade e este utiliza a previsão do uso de *hardware*, sendo a CPU o recurso de *hardware* mais importante. Como a previsão da demanda é essencial para o planejamento da capacidade, o presente trabalho utiliza a previsão de utilização de CPU como parâmetro para planejamento da capacidade. Então foi decidido criar um banco de dados com as características dos dados operacionais amplamente coletados do funcionamento de servidor de computação em nuvem.

A lógica para geração do conjunto de dados a seguir é baseada em um vídeo no *YouTube* que utiliza a linguagem de *Python* para gerar um modelo usado para prever e que possui características de sazonalidade, tendência e ruído (PAGE, 2020a; PAGE, 2020b).

A metodologia utilizada para geração da série temporal foi a análise visual dos gráficos e figuras dos trabalhos relacionados, especificamente o trabalho de Neil McMenemy The Royal Bank of Scotland Plc (1996). E de forma qualitativa, através das características

observadas, foi feito a configuração dos parâmetros das componentes que somadas geram a sequência de valores artificiais para o uso de CPU de um servidor de computação em nuvem. Assim, são geradas três componentes, a componente cíclico irregular semanal, a componente de sazonalidade e a componente de tendência.

Foi escolhida uma taxa de amostragem de uma amostra por dia para o sinal com intervalo de duração de 1030 dias, aproximadamente três anos. E cada amostra representa o pico diário de uso de CPU em um servidor de computação em nuvem.

3.2.1 Geração da componente cíclico irregular semanal

Para gerar os dados relacionados a utilização de CPU na Figura 7 foi utilizado as premissas de que a utilização de CPU pode ser analisada a partir do pico de utilização diária e os dias da semana tem maiores picos de utilização que os dias de final de semana como descrito em Neil McMenemy The Royal Bank of Scotland Plc (1996). Também foi analisado os gráficos disponíveis na mesma fonte para definir os parâmetros para geração do gráfico da componente em particular. Assim, através de análise visual da do gráfico da Figura 2, os valores inferiores (final de semana) tem média de 30% variando entre 20% e 45% e os valores superiores (dias da semana) tem média de 80% variando entre 70% e 95%.

Para representação desses dados, supõe que estão sob efeito de *Additive White Gaussian Noise* / Ruído Branco Gaussiano Aditivo (AWGN) como representado na Equação 3.1 que foi modelada em função de um vetor de tempo com 28 amostras e duração de 28 dias, ou aproximadamente um mês

$$ciclico(t) = p(t) + n(t) \tag{3.1}$$

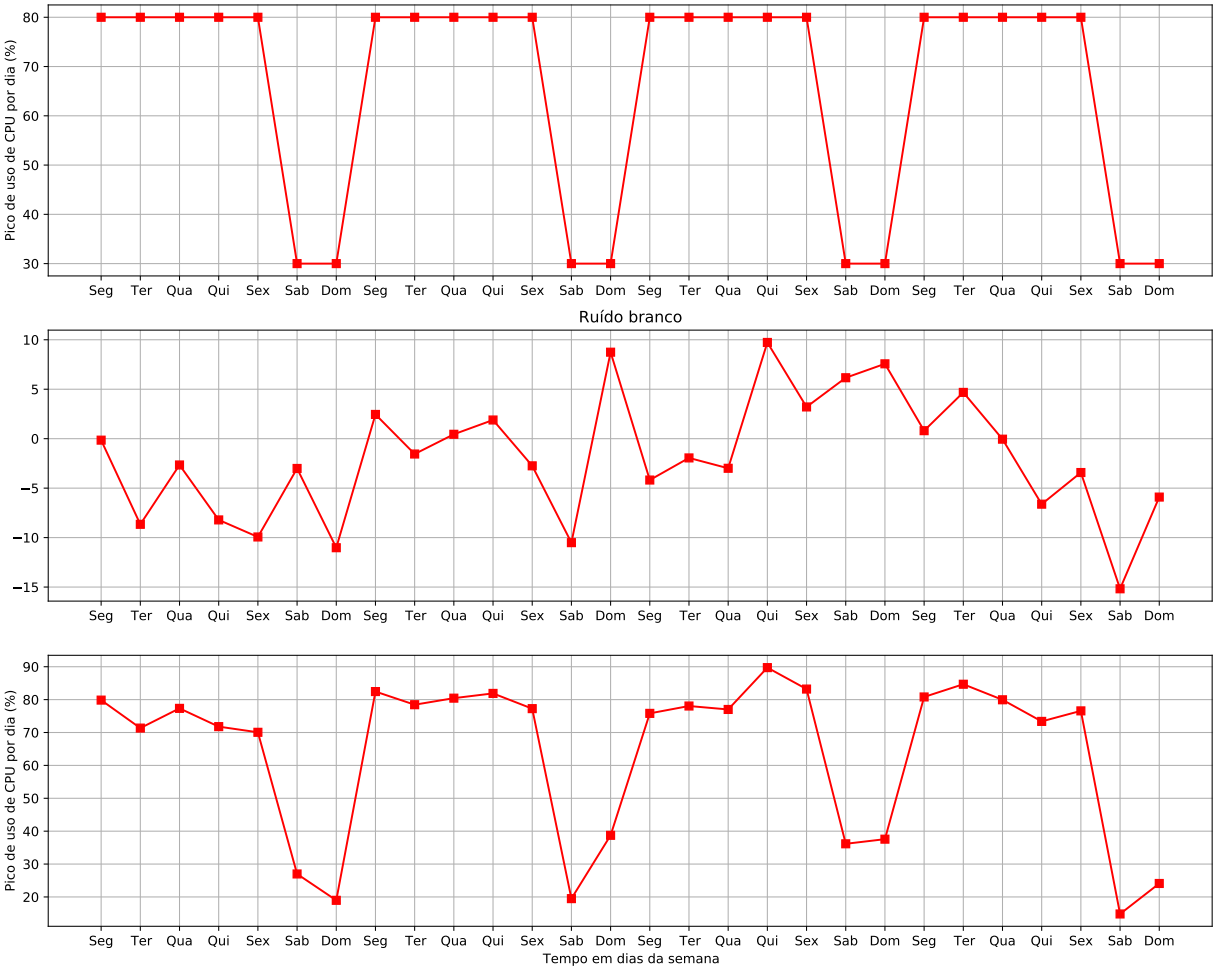
em que p é a média diária dos picos dos dias da semana ou dos dias do final de semana, como mostrado na Equação 3.2, e n representa o termo de ruído aditivo, modelado como uma variável aleatória Gaussiana com media zero e variância σ^2, ou seja, $n \sim \mathcal{N}(0, \sigma^2)$, também chamado de ruído branco. Sendo o desvio padrão σ=7.

$$p(t) = \begin{cases} 30u(t) & \text{se } t = 6,7,13,14,20,21,27,28, \text{ ou seja, dias do final de semana} \\ 80u(t) & \text{caso contrário} \end{cases} . \tag{3.2}$$

O resultado da geração do dados que representam o componente cíclico irregular semanal está representado na Figura 7 e esta figura mostra como foi composto esses dados através

do componente irregular, representado através do ruído branco, adicionado a característica de uso semanal.

Figura 7 – Componente cíclico irregular semanal



Fonte: elaborado pelo autor através do arquivo de Campos (2022).

3.2.2 Geração da componente de sazonalidade

Para representar a componente de sazonalidade primeiramente foi criado um vetor de tempo com 360 amostras e duração de 360 dias, ou aproximadamente um ano, para utilizar na senoide definindo a frequência de amostragem de uma amostra por dia.

A fim de construir um sinal resultante da soma de senoides, primeiramente é necessário gerar as senoides com os parâmetros escolhidos mostrados na Tabela 1. O método utilizado para definir os parâmetros da função geradora da componente de sazonalidade é a análise visual e estimativa da periodicidade através da Figura 1 do artigo de Neil McMenemy The Royal Bank of Scotland Plc (1996). .

Tabela 1 – Parâmetros das senoides

Período	Período (dias)	Frequência (Hz)	f
Anual	365	$\frac{1}{365}$	f_1
Semestral	181	$\frac{1}{181}$	f_2
Bimestral	61	$\frac{1}{61}$	f_3
Mensal	30	$\frac{1}{30}$	f_4

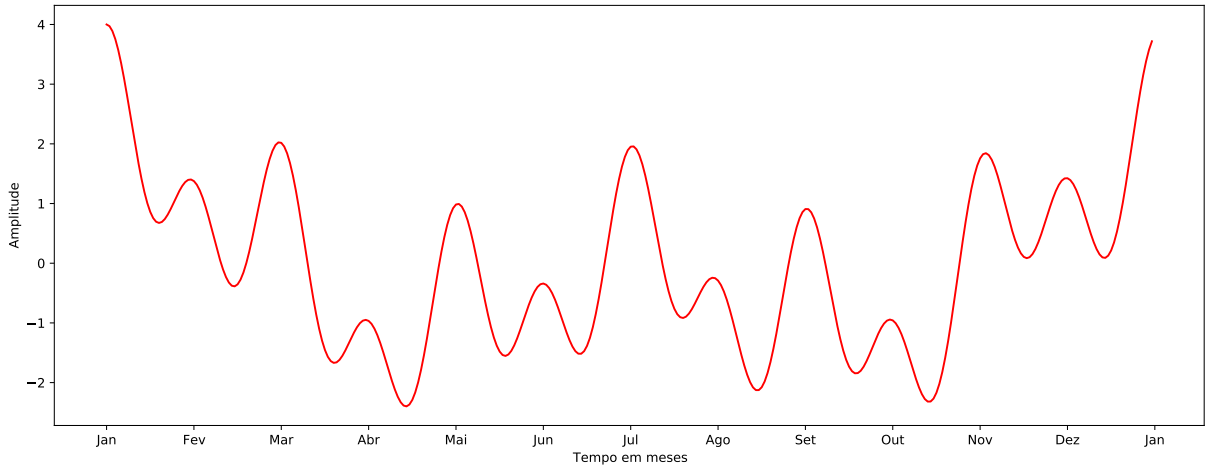
Fonte: elaborado pelo autor.

Foi escolhido utilizar senoides normalizadas, ou seja, com amplitude unitária, quatro funções cosseno com frequências diferentes como mostrado na Equação 3.3

$$sazonal(t) = cos(2\pi f_1 t) + cos(2\pi f_2 t) + cos(2\pi f_3 t) + cos(2\pi f_4 t) \tag{3.3}$$

O resultado do somatório das funções cosseno em um intervalo anual pode ser visto na Figura 8. Nesta figura há padrão de repetição mensal, semestral e anual principalmente. Isso pode ser observado através dos picos em cada mês, pico maior no meio do ano e um maior pico no início ou final do ano.

Figura 8 – Componente de sazonalidade



Fonte: elaborado pelo autor através do arquivo de Campos (2022).

3.2.3 Geração da componente de tendência

Para representar a componente de tendência foi escolhido uma função linear que pode ser representada pela Equação 3.4 e está em função de um vetor de tempo.

$$tend(t) = 0,03t \tag{3.4}$$

3.2.4 Composição da série temporal gerada

As componentes compõem de forma aditiva a série temporal característica do pico de uso de CPU em servidores de computação em nuvem, como na Equação 3.5. Os parâmetros das funções correspondentes às componentes são ajustados para não ultrapassar 100% de uso de CPU e representar as características observadas nas Figuras 1 e 2 após a composição final da série temporal. Esse ajuste para não ultrapassar 100% de uso de CPU é devido a assunção de que no período observado não houve alteração nos recursos de *hardware* e não houve aumento na capacidade.

Assim a componente cíclico irregular semanal tem os valores inferiores (final de semana) tem média ajustada para 20% e os valores superiores (dias da semana) tem média de 60% com ruído aditivo modelado como uma variável aleatória Gaussiana com media zero, variância σ^2 e desvio padrão $\sigma = 5$. Já a componente de sazonalidade da Equação 3.3 foi ajustada com o fator multiplicativo de cinco vezes.

$$s(t) = (5 \times sazonal(t)) + tend(t) + ciclico(t) \tag{3.5}$$

em que $sazonal(t)$ é a componente da sazonalidade, $tend(t)$ é a componente da tendência e $ciclico(t)$ é a componente cíclico irregular semanal.

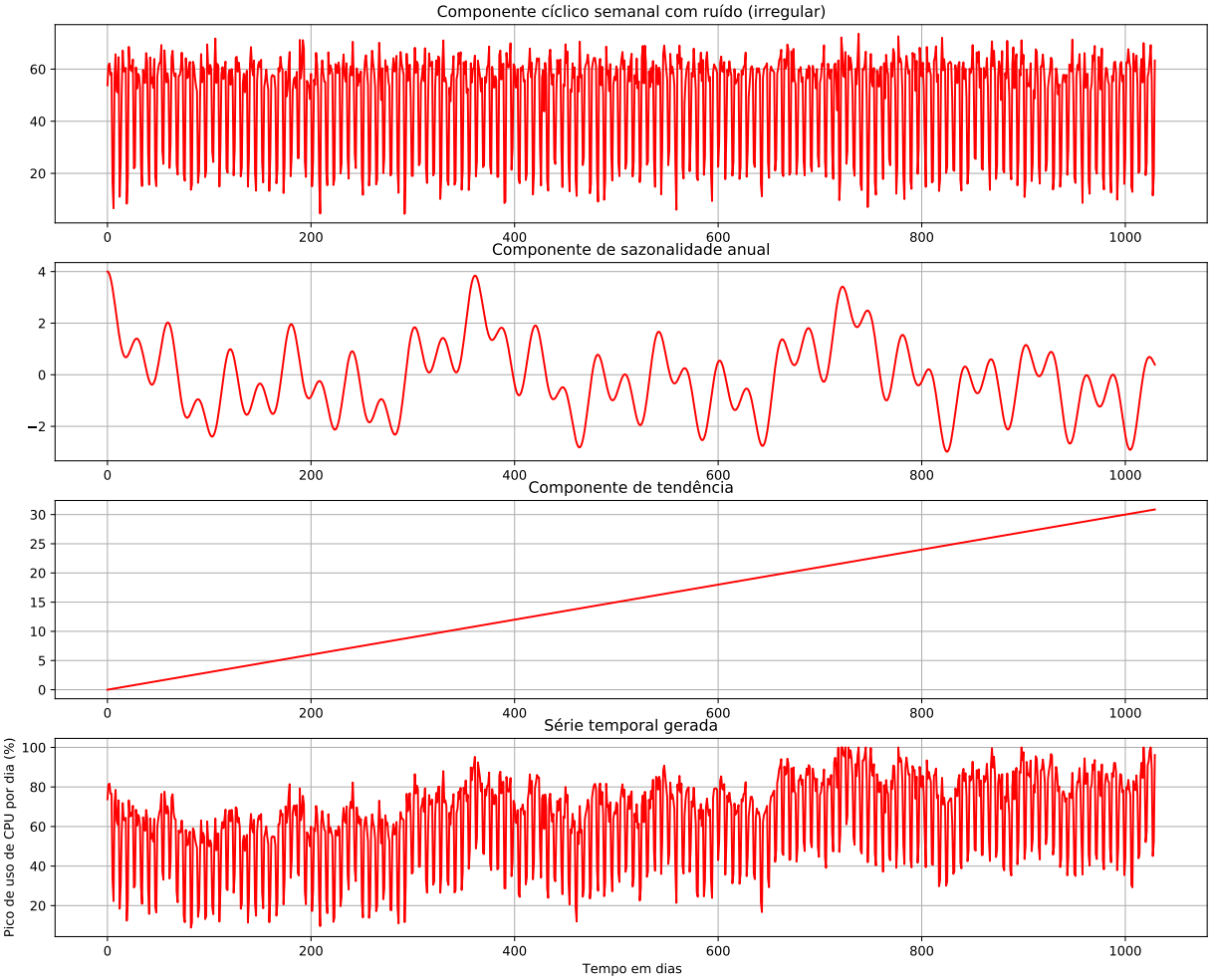
A componente cíclico irregular semanal, a componente de sazonalidade e a componente de tendência são modeladas em função de um vetor de tempo com 1030 amostras e duração de 1030 dias, ou aproximadamente três anos. O resultado da composição da série temporal por essas três componentes está representado na Figura 9.

3.3 Métodos de previsão

O método de previsão é escolhido de acordo com as características dos dados com implementação do mais simples para o mais complexo. O método escolhido para previsão final é aquele que apresenta melhor avaliação de acordo com as métricas escolhidas balanceado com a complexidade do método, ou seja, escolha leva em conta o compromisso entre custo e benefício do método.

É esperado que os métodos lineares de regressão não se adéquem bem ao problema por causa da natureza não linear dos dados e é esperado que métodos com grande flexibilidade se ajustem bem. Como se trata de uma série temporal, de acordo com a pesquisa, é esperado que o método de FNN seja suficiente e adequado para o problema.

Figura 9 – Componentes de uma série temporal e dados gerados



Fonte: elaborado pelo autor através do arquivo de Campos (2022).

3.3.1 Regressão Linear

O método mais simples é a regressão linear que pode ser utilizada para descobrir a tendência linear dos dados e tem alto grau de interpretabilidade, porém com pouca flexibilidade como podemos ver na Figura 4. É esperado que esse método não tenha se ajuste bem aos dados, por causa das características da série temporal.

Considerando que não é necessário pré-processamento, o primeiro passo é dividir o conjunto de dados em conjunto de dados de treino e conjunto de dados de teste. Como se trata de uma série temporal e os dados são sequenciais, a divisão é feita com as primeiras 730 amostras ou aproximadamente 2 anos para treino e as seguintes 300 amostras ou aproximadamente 1 ano para teste.

Depois é feito o treinamento do modelo, a predição da saída e a avaliação do modelo. O treinamento do modelo é feito com o conjunto de dados de treino e utilizando a biblioteca do *Scikit-learn* com o seu modelo para regressão linear. O modelo resultante é utilizado para fazer a

previsão da saída através da entrada referente ao conjunto de dados de teste. A saída é o uso de CPU e a entrada é o vetor de tempo. A previsão da saída é comparada através do RMSE e R^2 com a saída observada referente ao conjunto de dados de teste.

3.3.2 Rede Neural Feed-forward

O método com maior flexibilidade e que é considerado mais simples para previsão de séries temporais é o FNN. É esperado que o modelo criado com esse método se ajuste bem aos dados e também represente bem, através da sua previsão, a variabilidade característica de uma série temporal.

A lógica a seguir foi baseada em um vídeo no *YouTube* que utiliza uma FNN para um problema em que, dado um ano e um mês, a tarefa é prever o número de passageiros de companhias aéreas internacionais (BHATTIPROLU, 2020a; BHATTIPROLU, 2020b).

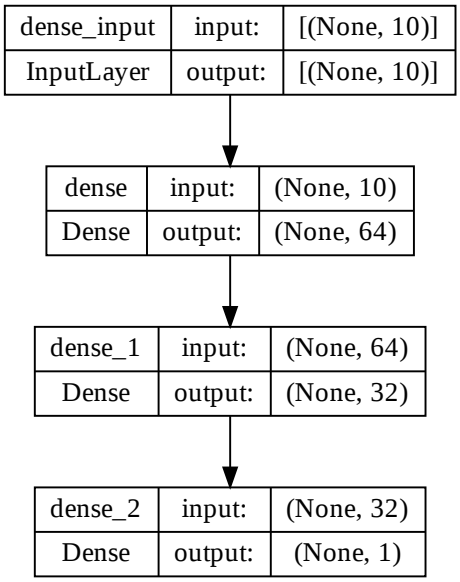
Primeiramente para aplicação do FNN é necessário o pré-processamento dos dados. Neste caso é feita a mudança de escala utilizando a função `MinMaxScaler().fit_transform()` da biblioteca *Scikit-learn* transformando os números para uma nova escala de 0 a 1. Depois é feita a divisão do conjunto de dados em conjunto de dados de treino e conjunto de dados de teste, seguindo o mesmo critério estabelecido para o método de regressão linear. As primeiras 730 amostras ou aproximadamente 2 anos para o conjunto de dados de treino e as seguintes 300 amostras ou aproximadamente 1 ano para o conjunto de dados de teste.

O treinamento do modelo começa após a organização da nova dimensão dos dados, pois o modelo leva em conta a sequência dos dados, considerando múltiplas entradas e uma única saída. É criada uma janela de treinamento que desloca as múltiplas entradas pela sequência dos dados da série temporal. O tamanho da janela escolhida é de 10 amostras e assim é utilizada para fazer a previsão da 11ª, depois as 10 amostras anteriores a 12ª amostra são usadas como entradas e assim por diante.

A rede neural é configurada para um vetor de entrada com 10 amostras, primeira camada densa com 64 neurônios, segunda camada densa com 32 neurônios e uma saída. É uma NN do tipo FNN com múltiplas entradas (10 neurônios), múltiplas camadas (2 camadas intermediárias) e única saída. A Figura 10 representa a rede neural criada. A função de ativação usada é a ReLU e o modelo é o `Sequential()` do *Keras*.

Ao final do treinamento, é utilizado o modelo treinado para previsão do conjunto de dados de teste seguindo o mesmo molde do treinamento, ou seja, a previsão começa para

Figura 10 – Diagrama da Rede Neural Criada



Fonte: elaborado pelo autor através do arquivo de Campos (2022).

a 11ª amostra do conjunto de testes. Em seguida é feita a transformação inversa utilizando `MinMaxScaler().inverse_transform()` retornando os dados a escala que podem ser interpretados.

3.4 Análise dos dados

A análise dos resultados é feita de forma visual, observando e analisando os gráficos gerados que comparam os dados de teste com os dados da previsão, uma análise qualitativa. E a análise dos resultados também é feita de forma quantitativa através das medidas RMSE e R^2, além da comparação entre os modelos.

É esperado que o modelo de regressão linear ofereça previsões com RMSE maiores e R^2 menores se comparado às previsões utilizando o modelo FNN. Além disso, espera-se que a regressão linear tenha um gráfico de previsão bem diferente dos dados de teste, em forma de uma função linear.

3.5 Escrita da monografia

A escrita da monografia segue a metodologia do Professor Gilson Volpato (VOLPATO, 2012) que está explicada em vídeo-aulas no *YouTube*. Assim o início da escrita do texto

científico parte do pressuposto que a pesquisa foi concluída, o TCC está concluído e os resultados foram alcançados, seguindo a seguinte ordem para elaboração do texto:

1. Conclusão;
2. Resultados;
3. Métodos;
4. Fundamentação teórica;
5. Introdução;
6. Resumo.

A escrita da monografia, dessa forma, segue o objetivo de se deter as explicações importantes necessárias para o entendimento da essência do texto, o que foi concluído através do trabalho. As etapas de escrita começam pela conclusão e as seguintes se detém as informações necessárias para entender a etapa escrita anteriormente.

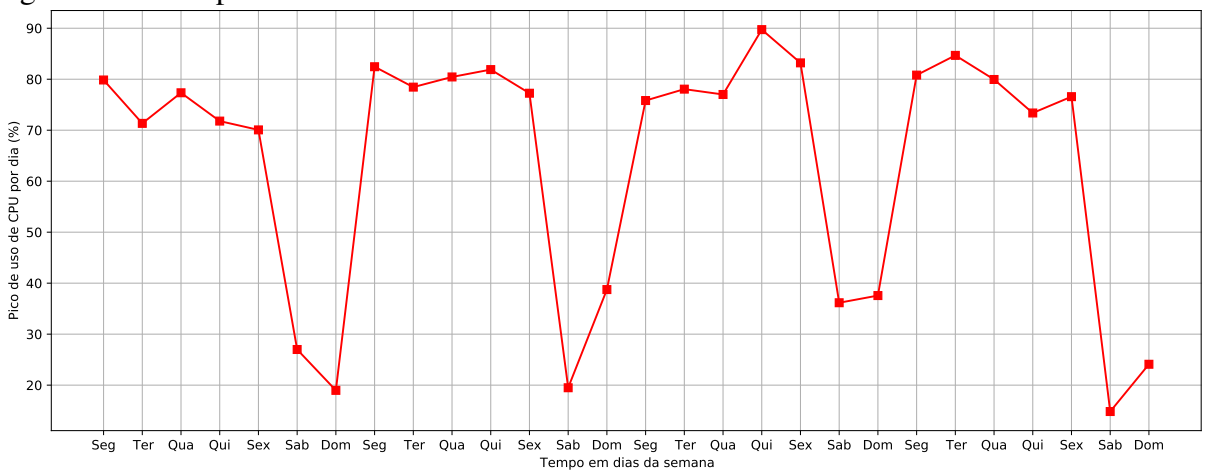
A estrutura do texto da monografia foi montada com base no *template* do modelo de trabalho acadêmico (UNIVERSIDADE FEDERAL DO CEARÁ, 2022b) utilizando os recursos do LaTeX na plataforma do *Overleaf* e seguido as normas técnicas apresentadas no Guia de Normalização de Trabalhos Acadêmicos da Universidade Federal do Ceará (UNIVERSIDADE FEDERAL DO CEARÁ, 2022a).

4 RESULTADOS

A previsão da demanda necessária para o gerenciamento da capacidade foi feita utilizando um banco de dados artificial criado baseado nas características dos dados operacionais amplamente coletados do funcionamento de servidor de computação em nuvem. Este banco de dados se caracteriza por representar, por meio de uma série temporal, uma sequência de valores de pico diário da porcentagem de uso de CPU em servidores de computação em nuvem.

Na Figura 11 está representada a variação semanal do pico diário de uso de CPU em servidores de computação em nuvem de uma sequência de quatro semanas quaisquer sem influência do componente de sazonalidade e de tendência. Notou-se que durante o final de semana os picos de uso registraram menores valores e durante os dias da semana registraram maiores valores resultando em um comportamento que, em média, se repete semanalmente e com variações irregulares em torno da média.

Figura 11 – Componente Cíclico semanal com ruído

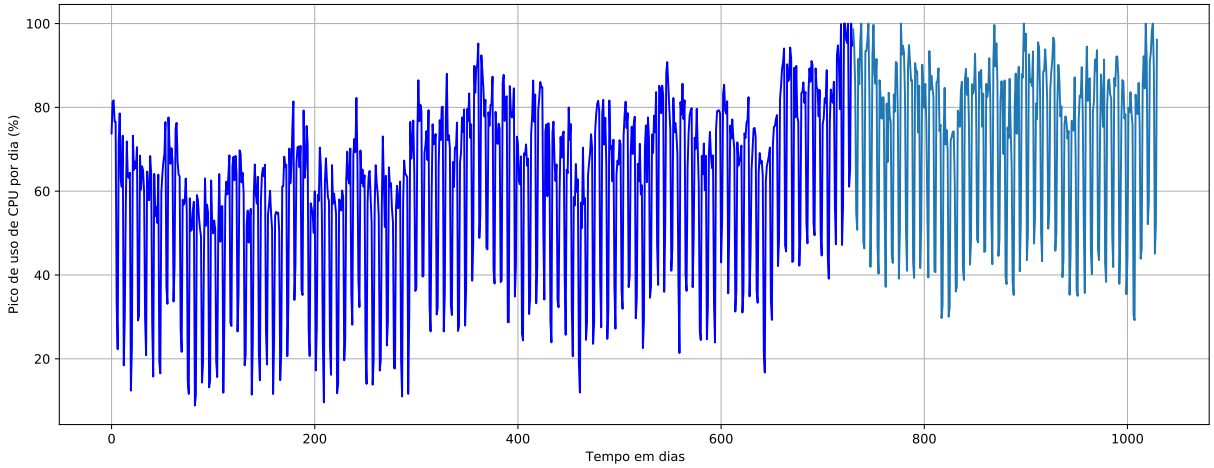


Fonte: elaborado pelo autor através do arquivo de Campos (2022).

Na Figura 12 está representada a série temporal gerada de uma sequência de aproximadamente três anos em que os dados gerados artificialmente possuem características combinadas das componentes cíclico irregular semanal, de sazonalidade e de tendência. Além disso, a Figura 12 evidencia a divisão entre dados de treino e dados de teste. Notou-se que no período de treino e teste não houve ultrapassagem da marca de 100 % do pico de uso, como esperado. Além disso, os valores estão em cresceram com o avançar dos dias.

Como foi visto na revisão de literatura, as características das séries temporais, como a sequência temporal do uso de CPU, trazem complexidade e os métodos mais simples como a regressão linear não se ajustam bem. Na Figura 13 evidenciou que métodos lineares não se

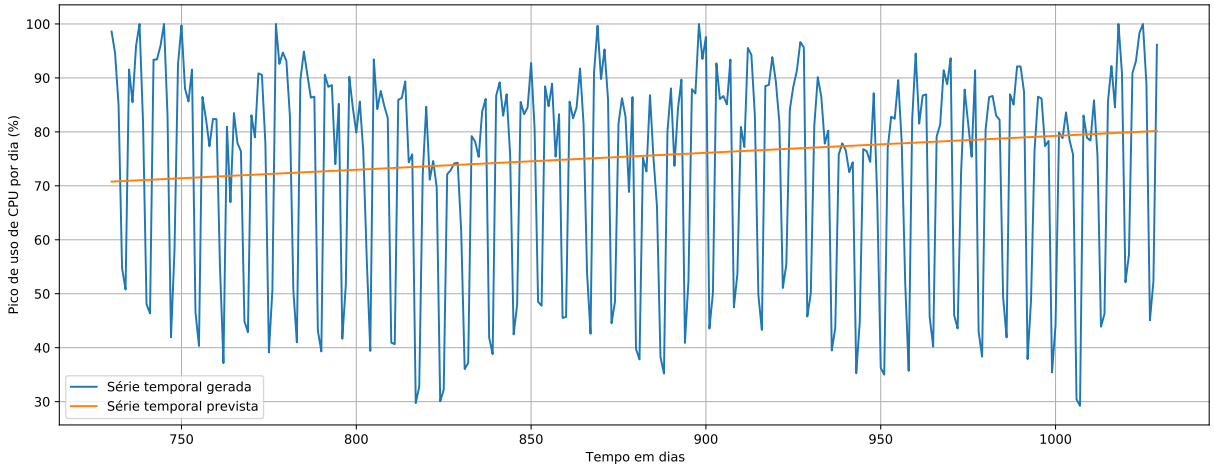
Figura 12 – Série Temporal Gerada e Divisão entre Treino e Teste



Fonte: elaborado pelo autor através do arquivo de Campos (2022).

ajustam bem às características do conjunto de dados gerados, pois visualmente percebeu-se que a série temporal prevista não apresenta a variabilidade dos dados, representando somente a tendência linear.

Figura 13 – Previsão utilizando regressão linear

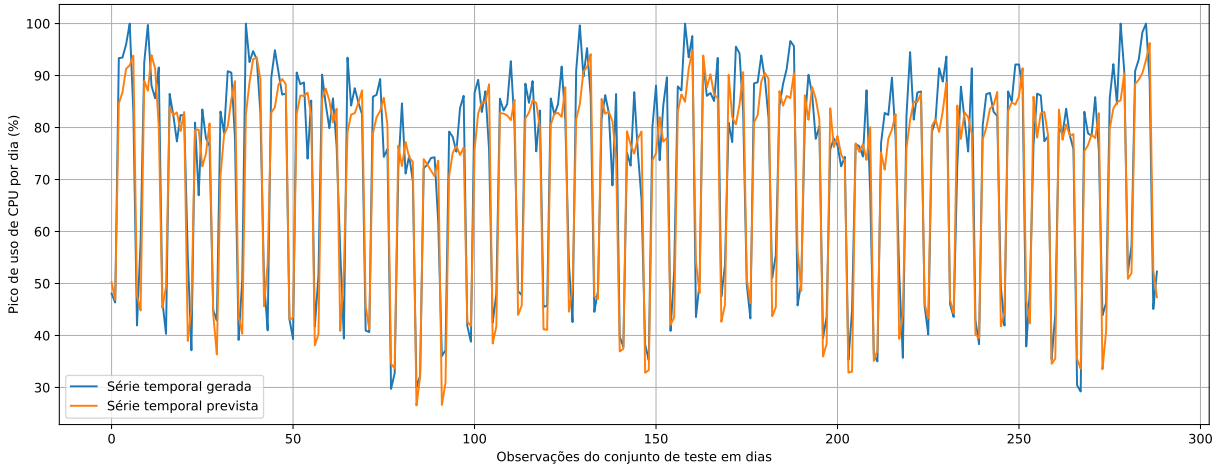


Fonte: elaborado pelo autor através do arquivo de Campos (2022).

Na Figura 14 está representado o resultado da previsão utilizando FNN. Notou-se que o método se ajustou bem às características dos dados gerados, pois no gráfico da previsão que utilizou FNN, a linha amarela seguiu o comportamento da linha azul e apresentou uma variabilidade similar. Ainda na Figura 14 notou-se que o modelo não utilizou todo o conjunto de teste, pois o gráfico é menor do que a quantidade de número de dias utilizado para validação e os dez primeiros dias do conjunto de teste são utilizados para a previsão, assim a amostra zero é equivalente ao 11° dia do conjunto de teste.

Na Tabela 2 as métricas de avaliação dos modelos confirmam quantitativamente o

Figura 14 – Previsão utilizando FNN para validação



Fonte: elaborado pelo autor através do arquivo de Campos (2022).

que foi observado nos gráficos. O valor de R^2 para o modelo FNN próximo ao valor unitário é típico de um modelo que está explicando os dados ou que prever corretamente. O modelo de regressão linear próximo a zero mostra que a regressão linear não prevê corretamente. Os valores de RMSE da regressão linear confirmam o *underfitting*.

Tabela 2 – Métricas de avaliação do modelo

Método	RMSE treino	RMSE teste	R2
Regressão Linear	19,9281	20,2932	-0,0411
Feedforward NN	5,4498	6,8540	0,8811

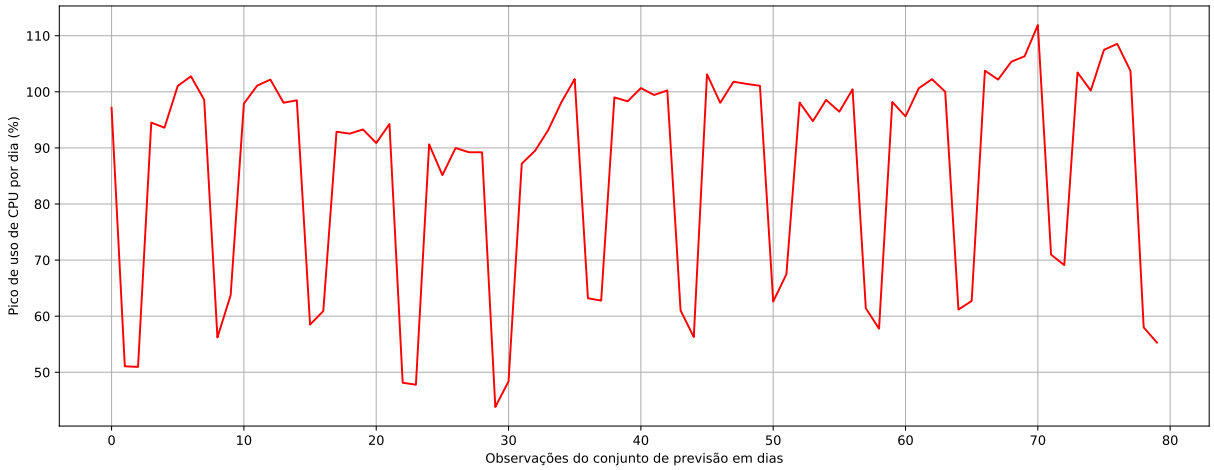
Fonte: elaborado pelo autor através do arquivo de Campos (2022).

A partir da Figura 15 mostra a previsão dos próximos 90 dias após a série temporal observada resultou em exceder a capacidade de uso de CPU no 25° dia, ou seja, a demanda apresentou maior valor que a capacidade. Sendo necessário a ampliação da capacidade total e sendo possível gerenciar os recursos relacionados a capacidade de acordo com o comportamento cíclico e sazonal previsto.

Outra observação é que o gráfico é menor do que a quantidade de número de dias utilizado para previsão e os dez primeiros dias do conjunto de previsão são utilizados como entrada para o modelo e assim a amostra zero é equivalente ao 11° dia do conjunto de previsão.

Como o modelo se ajustou bem aos dados e foi simples a visualização que a capacidade atual foi excedida, confirmou-se que os gerentes podem utilizar esses dados de previsão para auxílio na tomada de decisão, os quais auxiliam no planejamento da capacidade, tornando-se um processo menos subjetivo.

Figura 15 – Previsão utilizando FNN



Fonte: elaborado pelo autor através do arquivo de Campos (2022).

5 CONCLUSÕES E TRABALHOS FUTUROS

Conclui-se que métodos de inteligência computacional podem ser utilizados para previsão da demanda do uso de CPU nos servidores de serviços de computação em nuvem, especificamente os métodos não lineares, pois se ajustam bem às características dos dados, os quais apresentam sazonalidade, tendência, componentes cíclicos e aleatórios. Tal ajuste é evidenciado através dos testes com ótimos resultados nas métrica de avaliação RMSE e R^2 para redes neurais, confirmando a hipótese de que são métodos eficientes para previsão de séries temporais não estacionárias como o uso de CPU em servidores de computação em nuvem.

A partir disso, também conclui-se que os gerentes podem utilizar esses dados de previsão da demanda do uso de CPU como ferramenta para auxílio da tomada de decisão relacionada com o planejamento da capacidade que por sua vez está relacionado com o gerenciamento da capacidade. Portanto essa ferramenta objetiva baseados em dados históricos traz um horizonte de previsão que torna o processo em questão menos subjetivo.

Os resultados são representativos e demonstram a potencialidade do método, porém se limitam a origem dos dados, pois as previsões e o modelo desenvolvido utilizando FNN é aplicado a dados gerados, ou seja, dados artificiais e que levam somente em consideração o recurso de *hardware* CPU.

O trabalho também se limita a suposição de sazonalidade anual, pois faz com que a previsão seja mais eficiente com muitas observações, como utilizada nos resultados com a previsão do terceiro ano de uma série temporal. Além disso, pela literatura (FITZSIMMONS, 2014) o horizonte de previsão de curto prazo é o recomendado para modelos de séries temporais.

Recomenda-se a continuação dos estudos com a implementação e comparação de outros métodos no conjunto de dados gerados, além de testes com diferentes horizontes de previsão. Também recomenda-se a utilização de dados reais para realizar estudo de caso com o objetivo de validação e comprovação na prática do que foi simulado e do processo de previsão que foi desenvolvido para tomada de decisão.

REFERÊNCIAS

BHATTIPROLU, S. **164 - An introduction to time series forecasting**: - part 4 using feed forward neural networks. 2020. Disponível em: https://www.youtube.com/watch?v=tKM5d8Ll1k0. Acesso em: 9 dez. 2022.

BHATTIPROLU, S. **Python for microscopists**: 164a intro to time series forecasting using feed forward nn. 2020. Disponível em: https://github.com/bnsreenu/python_for_microscopists/blob/master/164a-Intro_to_time_series_Forecasting_using_feed_forward_NN.py. Acesso em: 9 dez. 2022.

CAMPOS, J. P. S. **Trabalho de Conclusão de Curso**: Inteligência computacional como suporte à tomada de decisão no gerenciamento de capacidade de serviços de computação em nuvem. 2022. Disponível em: https://github.com/CamposJoao/TCC-2022/blob/main/Resultados_TCC_JoaoPedroSilvaCampos.ipynb. Acesso em: 19 dez. 2022.

FENNER, G.; LIMA, A. S.; SOUZA, J. N. de; MOURA, J. A. B.; BEZERRA, T. R. Business-driven support for infrastructure as a service capacity management through system dynamics simulations. **IEEE Transactions on Network and Service Management**, v. 18, n. 2, p. 2063–2076, 2021.

FITZSIMMONS, M. J. F. J. A. **Administração de serviços**: operações, estratégia e tecnologia da informação. 7. ed. [*S. l.*]: AMGH, 2014.

GOOGLE. **Getting Started – scikit-learn 1.1.3 documentation**. 2022. Disponível em: https://scikit-learn.org/stable/getting_started.html. Acesso em: 6 dez. 2022.

HASTIE ROBERT TIBSHIRANI, G. J. D. W. T. **An Introduction to Statistical Learning**: With applications in r. 2. ed. Springer, 2013. Disponível em: https://hastie.su.domains/ISLR2/ISLRv2_website.pdf. Acesso em: 9 dez. 2022.

HASTIE ROBERT TIBSHIRANI, J. F. T. **The Elements of Statistical Learning**: Data mining, inference, and prediction. 2. ed. Springer, 2017. Disponível em: https://hastie.su.domains/Papers/ESLII.pdf. Acesso em: 9 dez. 2022.

HUAWEI. **Cloud Computing Technology**. Hangzhou, Zhejiang, China: Springer, 2023. Disponível em: https://link.springer.com/content/pdf/10.1007/978-981-19-3026-3.pdf?pdf=button%20sticky. Acesso em: 9 dez. 2022.

IEEE PRESS. **Communication Networks and Service Management in The Era of Artificial Intelligence and Machine Learning**. Piscataway NJ: John Wiley & Sons, 2021.

MELL, T. G. P. **The NIST Definition of Cloud Computing**: Recommendations of the national institute of standards and technology. Gaithersburg, MD: NIST, 2011. Disponível em: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-145.pdf. Acesso em: 8 dez. 2022.

MOLODORIA, A. **How To Apply Machine Learning To Demand Forecasting**. MobiDev, 2022. Disponível em: https://mobidev.biz/blog/machine-learning-methods-demand-forecasting-retail. Acesso em: 20 nov. 2022.

NEIL MCMENEMY THE ROYAL BANK OF SCOTLAND PLC. **Early Experiences with Neural Networks in Capacity Planning**. Glasgow, Scotland: The SAS Users Scotland Group, 1996. Disponível em: https://support.sas.com/resources/papers/proceedings-archive/SEUGI1997/MCMENEMY_DATAMIN.PDF. Acesso em: 20 nov. 2022.

OFFICE OF GOVERNMENT COMMERCE. **ITIL**: Service design. London UK: TSO, 2011.

PAGE, D. **Machine Learning for Capacity Management**. EnterpriseDB Corporation, 2020. Disponível em: https://youtu.be/sEUOzU9GMYI. Acesso em: 9 dez. 2022.

PAGE, D. **ml experiments**: time series cnn demo. EnterpriseDB Corporation, 2020. Disponível em: https://github.com/dpage/ml-experiments/blob/main/time_series/cnn-demo.py. Acesso em: 9 dez. 2022.

PMI, P. M. I. **PMBOK Guide**: A guide to the project management body of knowledge. 6. ed. Newtown Square, PA: Project Management Institute, 2017.

RAN, Y.; YANG, J.; ZHANG, S.; XI, H. Dynamic iaas computing resource provisioning strategy with qos constraint. **IEEE Transactions on Services Computing**, v. 10, 08 2015.

UNIVERSIDADE FEDERAL DO CEARÁ. **Guia de Normalização de Trabalhos Acadêmicos da Universidade Federal do Ceará**. Biblioteca Universitária, Comissão de Normalização, 2022. Disponível em: https://biblioteca.ufc.br/wp-content/uploads/2022/05/guianormalizacaotrabalhosacademicos-17.05.2022.pdf. Acesso em: 9 dez. 2022.

UNIVERSIDADE FEDERAL DO CEARÁ. **Modelo de trabalho acadêmico utilizando o Overleaf (2022) - UFC**. Biblioteca Universitária, 2022. Disponível em: https://www.overleaf.com/project/614dcd6b914d0cc23f819ec9. Acesso em: 9 dez. 2022.

VOLPATO, G. **Método Lógico para Redação Científica**: bases e aplicação. 2012. Disponível em: https://www.youtube.com/playlist?list=PLMmWegTl-vzV7ScJqOiXI-p0QamOE8hBy. Acesso em: 9 dez. 2022.

WIKIPÉDIA. **Rede Neural Artificial**. 2022. Disponível em: https://pt.wikipedia.org/wiki/Rede_neural_artificial. Acesso em: 9 dez. 2022.