UNIVERSIDADE PRESBITERIANA MACKENZIE

LEONARDO DOS REIS OLHER

THAINÁ VIEIRA DOS SANTOS

VINICIUS CAUMO SEGATTO

NICOLAS PINOTTI

VINÍCIUS VIEIRA DA CUNHA OLIVEIRA

PROJETO APLICADO: DATATRENDS INSIGHTS

São Paulo

2023

LEONARDO DOS REIS OLHER

THAINÁ VIEIRA DOS SANTOS

VINICIUS CAUMO SEGATTO

NICOLAS PINOTTI

VINÍCIUS VIEIRA DA CUNHA OLIVEIRA

PROJETO APLICADO: DATATRENDS INSIGHTS

Trabalho Aplicando conhecimento para entrega no Moodle referente ao conteúdo 3 de aprendizagem do componente curricular Ciência, tecnologia e sociedade;

ORIENTADOR: Prof. ANDERSON ADAIME DE BORBA

São Paulo

2023

LEONARDO DOS REIS OLHER

THAINÁ VIEIRA DOS SANTOS

VINICIUS CAUMO SEGATTO

NICOLAS PINOTTI

VINÍCIUS VIEIRA DA CUNHA OLIVEIRA

PROJETO APLICADO: DATATRENDS INSIGHTS

Trabalho Aplicando conhecimento para entrega no Moodle referente ao conteúdo 3 de aprendizagem do componente curricular Ciência, tecnologia e sociedade;

Aprovado em

BANCA EXAMINADORA

_____

Prof. Anderson Adaime de Borba

Universidade Presbiteriana Mackenzie

LISTA DE FIGURAS

SUMÁRIO

1. INTRODUÇÃO

1.1 APRESENTAÇÃO DO PROJETO

A empresa *DataTrend Insights* está enfrentando um problema interno relacionado à retenção de Cientistas de Dados Júnior. Nossa empresa está perdendo talentos promissores para concorrentes devido a diferenças salariais e benefícios. Precisamos identificar a faixa salarial competitiva para Cientistas de Dados Júnior em 2023 e criar uma estratégia de retenção eficaz.

1.2 ÁREA DE ATUAÇÃO

A empresa atua principalmente nos ramos de consultoria em remuneração oferecendo serviços de consultoria para outras empresas, ajudando-as a compreender e ajustar suas políticas de remuneração em relação às tendências salariais em Ciência de Dados, envolvendo análises detalhadas do mercado, identificação de faixas salariais competitivas e recomendações específicas. Além disso, a empresa atua no ramo de soluções de software, onde desenvolve e vende soluções de software especializadas para a gestão de remuneração em Ciência de Dados. Isso incluiria ferramentas de análise de dados, modelagem de salários e previsão de tendências salariais.

1.3 DADOS UTILIZADOS

A *DataTrend Insights* baseia suas análises de tendências salariais em Ciência de Dados em 2023 em um conjunto de dados cuidadosamente selecionado e abrangente. Este conjunto de dados serve como a espinha dorsal de nossos esforços analíticos, fornecendo informações essenciais para compreender a dinâmica salarial na indústria de Ciência de Dados. Abaixo estão os principais componentes desse conjunto de dados: o Ano de Trabalho: Esta coluna representa o ano específico da coleta de dados salariais. o Nível de Experiência: Os funcionários são categorizados de acordo com seu nível de experiência, incluindo iniciantes, experientes, de nível médio e sêniores. o Tipo de Emprego: Cada profissional é rotulado com seu tipo de emprego, que pode ser tempo integral, contratado, freelancer ou meio período. o Cargo: Registramos os cargos dos funcionários, abrangendo uma variedade de títulos, como "Cientista Aplicado" e "Analista de Qualidade de Dados". o Salário: Esta coluna contém os valores salariais, expressos em suas respectivas moedas locais. o Moeda do Salário: Indica o código da moeda que representa o salário em questão. o Salário em USD: Todos os salários foram convertidos

para dólares americanos (USD) para permitir uma comparação uniforme. o Localização da Empresa: Esta coluna especifica a localização das empresas, identificadas por códigos de país, como "US" para Estados Unidos e "NG" para Nigéria. o Tamanho da Empresa: As empresas são classificadas em categorias de tamanho, que incluem grande, média e pequena. Esses dados, cuidadosamente coletados e preparados, servirão como a base para nossas análises estatísticas avançadas, modelagem preditiva e visualizações de dados. Com esses insights, nossa equipe pode ajudar empresas a tomar decisões estratégicas informadas sobre remuneração, retenção de talentos e estratégias de aquisição de talentos em Ciência de Dados. A integridade, qualidade e relevância desses dados são fundamentais para garantir que nossas análises sejam precisas e confiáveis, permitindo-nos fornecer serviços de consultoria e soluções personalizadas de alta qualidade para nossos clientes.

Há também um cuidado crucial na abordagem deste projeto: a análise da base de dados não pode se limitar apenas ao salário, pois o salário não é uma variável textual, mas sim uma variável numérica. Na realidade, o salário pode ser considerado uma variável contínua. O sucesso da nossa análise depende da consideração de variáveis textuais, como o nível de experiência (junior, senior), a moeda (BRL, USD) e outras variáveis textuais presentes na base de dados. Essas variáveis desempenham um papel fundamental na compreensão das dinâmicas salariais em Ciência de Dados. Ao considerar essas variáveis textuais, seremos capazes de explorar relacionamentos e tendências significativas que podem influenciar os salários, como a diferença salarial entre Analistas de Dados Júnior e Sênior, a variação de salários em diferentes moedas, e outros fatores que podem ser cruciais para a nossa estratégia de retenção de talentos.

2. DESENVOLVIMENTO

2.1 DEFINIÇÃO DA LINGUAGEM DE PROGRAMAÇÃO

Após cuidadosa consideração, a equipe de desenvolvedores da *DataTrend Insights* optou por conduzir a análise exploratória do dataset em Python, em vez da linguagem R. Essa decisão foi tomada com base em vários benefícios que o Python oferece para a nossa equipe e para o projeto como um todo. Um dos principais benefícios de escolher Python para a AED é a versatilidade da linguagem. Python é amplamente conhecido por sua capacidade de integração com várias bibliotecas de análise de dados, machine

learning e visualização. Isso nos permite criar um fluxo de trabalho contínuo, onde podemos realizar a análise exploratória, a preparação de dados e a modelagem preditiva, tudo em um único ambiente. Além disso, Python é uma linguagem de programação de propósito geral, o que significa que muitos de nossos desenvolvedores já têm experiência com ela. Isso facilita a colaboração e o compartilhamento de código entre a equipe. Também permite que nossos desenvolvedores usem suas habilidades de programação Python para automatizar tarefas e criar pipelines de análise de dados personalizados. Outro benefício importante é a vasta comunidade Python e a disponibilidade de recursos educacionais. Isso significa que podemos encontrar suporte, soluções para desafios técnicos e documentação facilmente. Além disso, as bibliotecas de Python, como Pandas, são bem documentadas e oferecem ampla funcionalidade para análise de dados e modelagem preditiva.

No Apêndice C, para uma referência mais interativa e acesso fácil, o código está disponível no repositório GitHub.

2.2 ANÁLISE EXPLORATÓRIA DE DADOS

A análise exploratória de dados é fundamental para o sucesso deste projeto, pois é o alicerce sobre o qual as decisões e estratégias serão construídas. Através dela, seremos capazes de extrair insights valiosos, identificar áreas de atenção e fundamentar as próximas etapas de análise e modelagem. Portanto, uma análise completa e bem executada é essencial para alcançar os objetivos definidos. O código completo da análise exploratória de dados se encontra no Apêndice A.

I) Carregamento dos dados e visualização das primeiras linhas de código

No item 1, importamos as bibliotecas necessárias, como pandas, numpy, statistics, matplotlib e seaborn. Essas bibliotecas desempenham um papel fundamental em nossa análise. Em seguida, no item 2, carregamos o conjunto de dados principal, denominado 'ds_salaries.csv'. Este conjunto de dados serve como a espinha dorsal de nossas análises, fornecendo informações essenciais sobre salários na área de Ciência de Dados.

II) Informações iniciais

No item 3 fizemos análises e tratamentos iniciais. No 3.1, verificamos a presença de valores nulos em nosso conjunto de dados, Isso nos ajuda a identificar qualquer informação ausente que possa requerer tratamento posterior. Além disso, com o seguinte

código, examinamos os tipos de dados presentes nas colunas. Essa etapa é importante para garantir que as variáveis sejam interpretadas corretamente em nossa análise. A fim de compreender a dimensão do conjunto de dados no item, verificamos seu tamanho (número de linhas e colunas) e por último geramos estatísticas descritivas para as variáveis numéricas do conjunto de dados, essas estatísticas resumidas fornecem informações sobre a distribuição e centralidade dos dados.

III) Limpeza e Transformação de Dados

Para garantir a qualidade dos dados e prepará-los para análises mais avançadas, é necessário realizar um pré-processamento. No item 4, realizamos uma filtragem de dados com base no título do cargo. Focamos nossa análise apenas nos cargos relacionados à "data analyst". O código filtra o Data-Frame para incluir apenas as entradas que contenham essa referência. Essa etapa nos permite concentrar nossa análise nas posições que são diretamente relevantes ao escopo do projeto. Para simplificar o conjunto de dados e remover informações que não são essenciais para nossos objetivos, identificamos um conjunto de colunas que podem ser descartadas, isso ajuda a trabalhar com um conjunto de dados mais enxuto e direcionado às informações de interesse. Além disso, para tornar as colunas mais descritivas e alinhadas com nossos objetivos de análise, renomeamos algumas delas, Essa etapa melhora a clareza e a usabilidade do conjunto de dados.

Essas etapas de pré-processamento de dados são fundamentais para garantir que as análises subsequentes sejam direcionadas e produzam resultados relevantes. Elas simplificam e preparam os dados para as etapas avançadas da análise exploratória e modelagem.

IV) Transformações de variáveis categóricas

No item 5 como um todo, descreveremos o processo de transformação de variáveis categóricas para tornar os dados mais informativos e adequados às análises.

IV.I) Categorização de "remote"

Para tornar a variável "remote" mais informativa, a classificamos em três categorias com base nos valores. Se o valor de "remote" for menor que 20, consideramos "No remote". Se estiver entre 20 e 79, classificamos como "Partially". E se for igual ou superior a 80, atribuímos "Remote".

IV.II) Renomeação de níveis de experiência

A coluna "experience" é atualizada para que os níveis de experiência sejam mais descritivos. Por exemplo, "EN" é substituído por "1 - Entry" e assim por diante.

IV.III) Atualização do "employment_type"

A coluna "employment_type" é redefinida com rótulos mais compreensíveis. "FT" é renomeado para "Full-Time", "CT" para "Contratante", "FL" para "Freelancer" e "PT" para "Meio período". Outros valores são mantidos como "-".

IV.IV) Reestruturação de "company_size"

A coluna "company_size" é reclassificada para categorias mais descritivas. "L" é renomeado para "3 - Grande", "M" para "2 - Média", "S" para "1 - Pequena" e outros valores são mantidos como "-".

IV.V) Criação da Variável "imigrante"

Uma nova variável chamada "imigrante" é criada para indicar se o local de residência do funcionário difere do local da empresa. Se houver diferença, a coluna "imigrante" recebe o valor True; caso contrário, recebe False. Essa variável pode ser útil para análises relacionadas à localização dos funcionários em relação às empresas onde trabalham.

V) Visualização dos dados

Como continuação da análise exploratória de dados, continuaremos a explorar visualmente as variáveis em nosso conjunto de dados. A visualização é uma ferramenta poderosa para identificar tendências, padrões e insights.

Nesta seção, realizamos análises estatísticas relacionadas à distribuição de salários e outras variáveis. O processo é dividido em várias etapas, cada uma focada em uma variável específica.

V.I) Análise de Variáveis Principais

- Para um conjunto de colunas selecionadas, calculamos a média salarial.
- As variáveis analisadas incluem "year", "experience", "residence", "remote", "company_location", "company_size", "employment_type" e "imigrante".

V.II) Distribuição de Salários - Histograma

- Este gráfico exibe a distribuição de salários no conjunto de dados.
- Linhas verticais representam a média, mediana e moda dos salários.

V.III) Média Salarial e Crescimento por Ano

- Analisamos a média salarial e o crescimento dos salários ao longo dos anos.
- A análise inclui uma representação gráfica da média salarial e do crescimento.

V.IV) Média Salarial e Crescimento por Experiência

- Investigamos a média salarial e o crescimento com base nos níveis de experiência.
- Visualizamos os resultados por meio de gráficos de barras.

V.V) Análise por Variáveis Categóricas

- Exploramos a relação entre o salário e várias variáveis categóricas, incluindo "remote", "company_size" e "employment_type".
- Apresentamos análises e distribuições em gráficos de barras e box plots.

V.VI) Análise por Localização e Imigração

- Analisamos a média salarial com base na localização da empresa, localização de residência e status de imigrante.
- Os resultados são representados visualmente em gráficos de barras e box plots.

V.VII) Análise de Variáveis Categóricas Adicionais

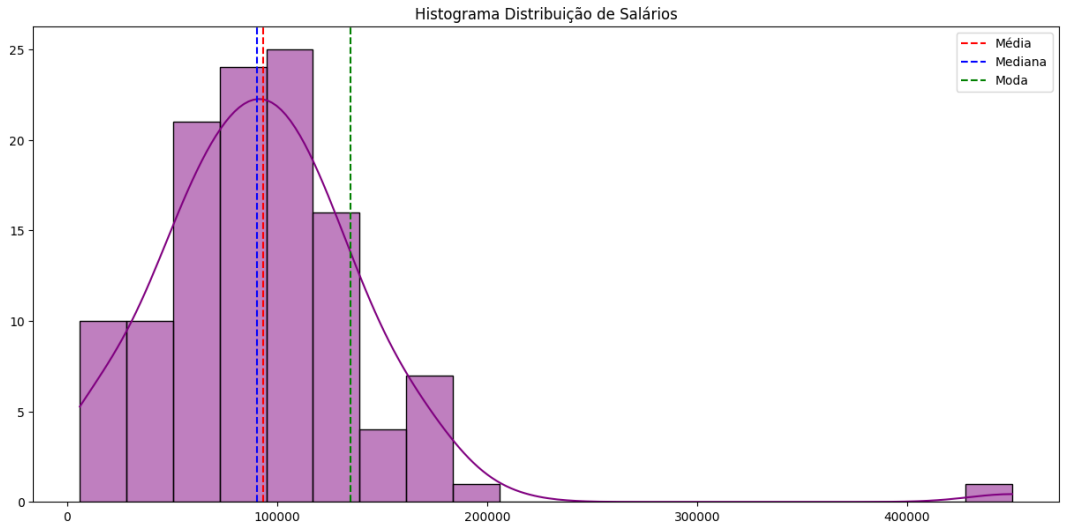
- Exploramos a relação entre o salário e outras variáveis categóricas, incluindo "experience", "employment_type" e "company_size".
- Os resultados são exibidos em gráficos de barras.

V.VIII) Análise Detalhada por Localização

- Investigamos a relação entre o salário e variáveis categóricas, considerando a localização da empresa.
- Os resultados são apresentados em gráficos de barras com informações detalhadas.

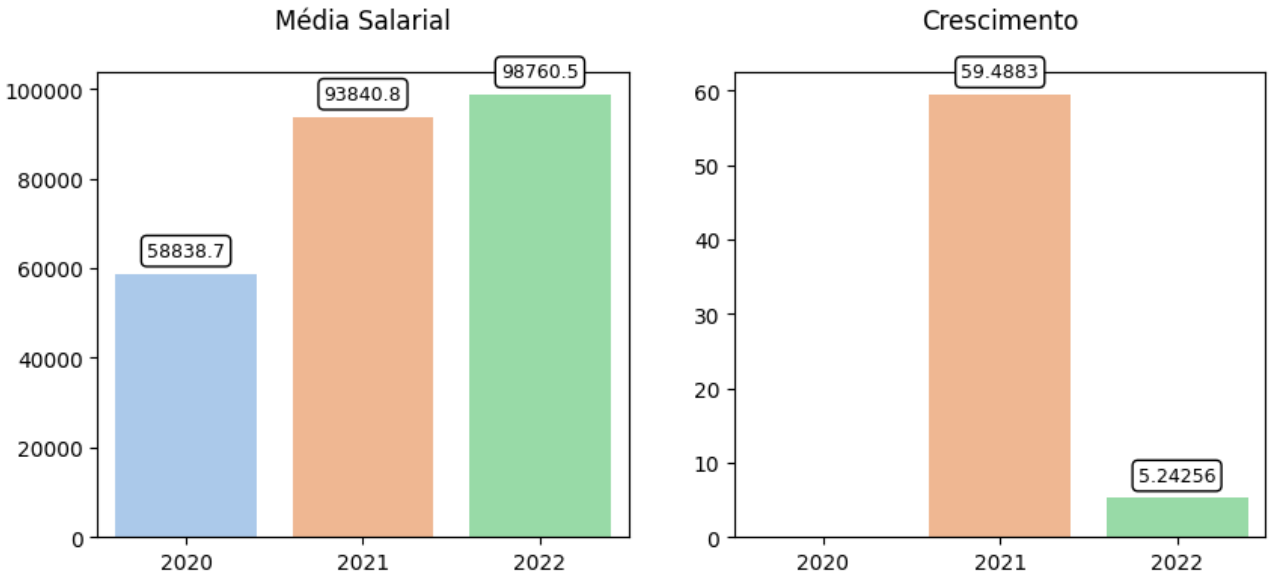
Essas análises estatísticas fornecem insights valiosos sobre a distribuição de salários e sua relação com várias variáveis no conjunto de dados, ajudando na compreensão das tendências e padrões subjacentes.

Figura 1 - Histograma Distribuição de Salários (item V.II)



Fonte: os autores, 2023

Figura 2 - Média salarial e crescimento por ano (item V.III)



Fonte: os autores, 2023

VI) Correlação e Significância

VI.I) Visualização de relações

- Codifica as variáveis "experiência" e "tamanho da empresa" usando Label Encoding.

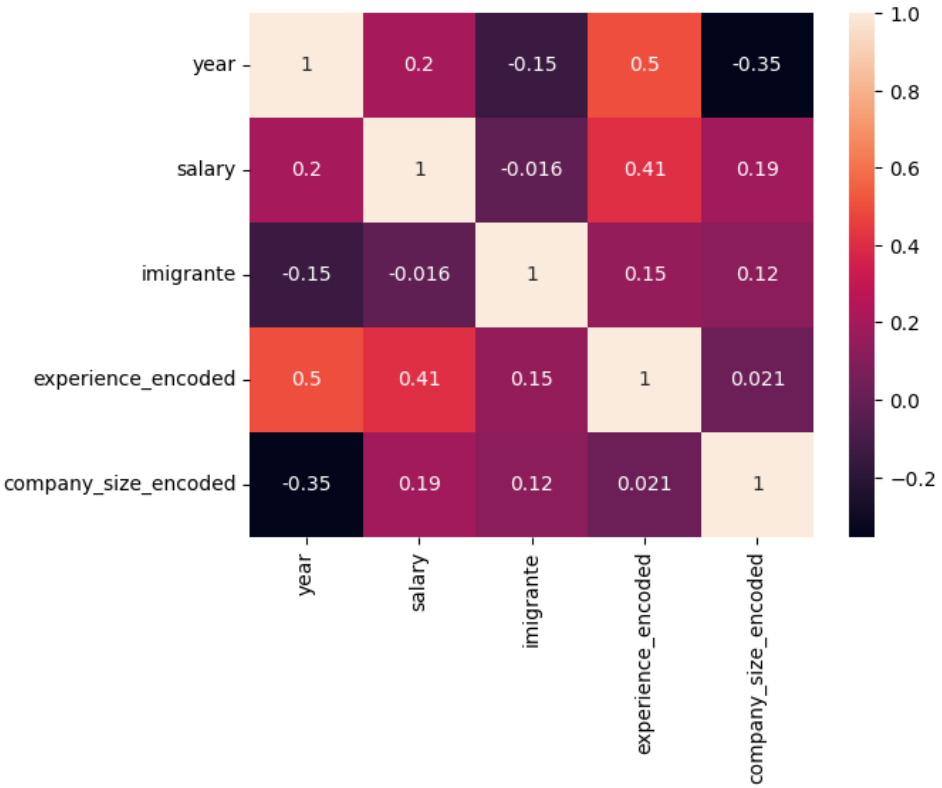
- Seleciona as colunas "ano", "imigração", "experiência codificada" e "tamanho da empresa codificado".
- Plota quatro gráficos de dispersão (scatter plots) com regressões lineares para analisar a relação entre as variáveis selecionadas e os salários.
- Os gráficos ajudam a visualizar como as variáveis estão relacionadas com os salários.

VI.II) Correlação de Pearson

- Calcula a correlação de Pearson entre as variáveis selecionadas e os salários.
- Avalia a significância estatística (p-value) da correlação.
- Exibe os resultados com estatísticas, coeficientes de correlação e significância.
- Plota um mapa de calor (heatmap) das correlações de Pearson.

A figura abaixo ilustra o resultado da aplicação da *Correlação de Pearson* na base de dados:

Figura 3 - Correlação de Pearson

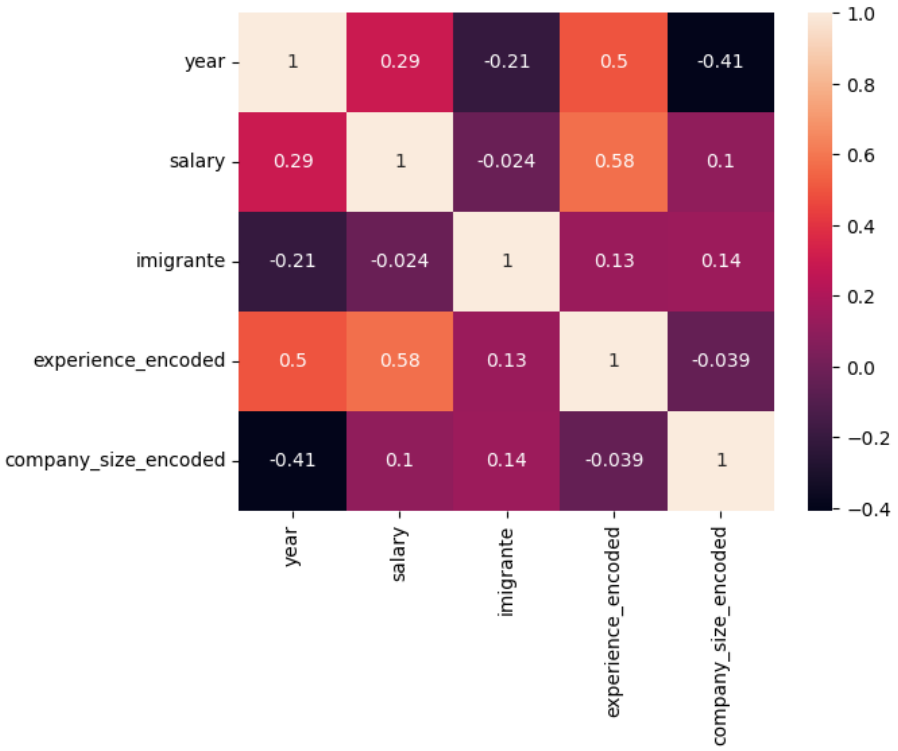


Fonte: os autores, 2023

VI.III) Correlação de Spearman

- Calcula a correlação de Spearman entre as variáveis selecionadas e os salários.
- Avalia a significância estatística (p-value) da correlação.
- Exibe os resultados com estatísticas, coeficientes de correlação e significância.
- Plota um mapa de calor (heatmap) das correlações de Spearman.

Figura 4 - Correlação de Spearman



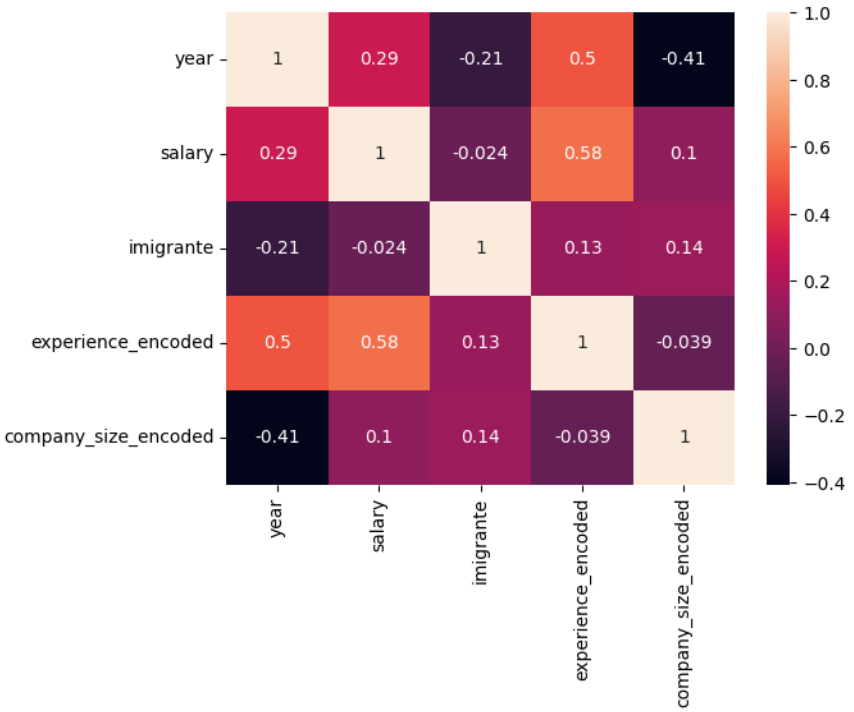
Fonte: os autores, 2023

VI.IV) Correlação de Kendall

- Calcula a correlação de Kendall entre as variáveis selecionadas e os salários.
- Avalia a significância estatística (p-value) da correlação.

- Exibe os resultados com estatísticas, coeficientes de correlação e significância.
- Plota um mapa de calor (heatmap) das correlações de Kendall.

Figura 5 - Correlação de Kendall



Fonte: os autores, 2023

Essas análises estatísticas e gráficas são importantes para compreender as relações entre as variáveis e os salários, fornecendo insights sobre quais fatores podem influenciar os salários em Ciência de Dados.

2.3) APRENDIZADO DE MÁQUINA: A METODOLOGIA RANDOM FOREST

Nesta seção, avançamos no projeto para aplicar técnicas de machine learning a fim de obter previsões e insights relacionados aos salários em empregos de Data Science. O machine learning desempenha um papel fundamental na análise de tendências salariais, permitindo-nos entender como os títulos de emprego influenciam os salários e criar modelos que possam prever esses salários com base em dados passados.

Utilizamos duas abordagens diferentes: k-Nearest Neighbors (k-NN) e Random Forest. O k-NN é uma técnica de aprendizado supervisionado que nos ajuda a entender a proximidade entre os salários de diferentes empregos de Data Science. O Random Forest é um modelo de aprendizado de máquina que é menos propenso a overfitting ou erros de

viés, que são mais comuns e fáceis de acontecer em árvores de decisão. Em Random Forest, você adiciona várias árvores de decisão para obter um resultado melhor. No entanto, você não quer complicar demais e adicionar árvores demais, o que pode causar lentidão e um gasto desnecessário de processamento**.** Uma árvore de decisão começa com uma pergunta simples. A partir dessa pergunta, podemos fazer várias perguntas para determinar uma resposta final para essa pergunta. Cada pergunta, com sua resposta, ajuda a tomar a decisão final.

As árvores de decisão tentam encontrar a melhor divisão dos dados, usando classificação ou regressão (CART). Métricas como impureza de Gini, ganho de informação ou erro quadrático médio (MSE) são usadas para avaliar a qualidade da divisão dos dados durante o treinamento.

O algoritmo de floresta aleatória, é uma extensão entre o método conhecido por escaneamento e também o de aleatoriedade, juntos essa combinação formam uma variação tão grande de dados que podem ser usados, que isso reduz o risco de causar overfiting, bias e outras variações criando assim um resultado mais preciso

Algoritmos de floresta aleatória precisam de três hiperparâmetros que devem ser passados antes do treinamento. Esses hiperparâmetros são:

O tamanho dos nós: determina o número de amostras necessárias para dividir um nó. Um tamanho de nó menor resulta em árvores mais complexas, enquanto um tamanho de nó maior resulta em árvores mais simples. A quantidade de árvores aleatórias: determina o número de árvores que serão treinadas. Um número maior de árvores resulta em um modelo mais robusto, mas também requer mais tempo de treinamento. O tamanho da amostra: determina o número de amostras que serão usadas para treinar cada árvore. Um tamanho de amostra maior resulta em árvores mais robustas, mas também requer mais memória. Cada amostra de dados é extraída de um conjunto de treinamento com reposição, chamada de amostra bootstrap. Essa amostra de treinamento é então dividida em duas partes:

A primeira parte é usada para treinar a árvore de decisão. A segunda parte é reservada como dados de teste, conhecido como amostra fora do saco (oob). Outra

instância de aleatoriedade é então injetada por meio do bagging de recursos. O bagging de recursos é um processo que seleciona aleatoriamente um subconjunto de recursos para cada árvore de decisão. Isso ajuda a reduzir a correlação entre as árvores de decisão, o que pode melhorar a generalização do modelo.

Agora, vamos detalhar cada etapa dos códigos relacionados a essas técnicas para entender como elas contribuem para o nosso projeto. Assim como ocorreu na AED, o APÊNDICE B contém todo o código dessa parte do KNN e Random Forest.

I) Limpeza dos dados

O código realiza as seguintes operações:

Previsão apenas de empregos full-time

O código remove todos os pontos de dados do conjunto de dados que não correspondem ao tipo de emprego Full-Time.

O código remove as colunas do conjunto de dados.

Adiciona uma coluna ao conjunto de dados para indicar o continente da empresa

II) Remoção de outliers

O código cria dois boxplots para visualizar a distribuição de salários antes e depois de remover os outliers.

O primeiro boxplot (ax[0]) mostra a distribuição de salários original. O segundo boxplot (ax[1]) mostra a distribuição de salários depois de remover os outliers.

Para identificar os outliers, o código usa o método IQR (Interquartile Range). O IQR é a diferença entre o terceiro e o primeiro quartil. Os outliers são definidos como os pontos de dados que estão fora de 1,5 IQRs do quartil inferior ou superior.

III) Codificação

O código codifica as variáveis categóricas no conjunto de dados usando Label Encoder e One Hot Encoder.

O Label Encoder codifica as variáveis categóricas atribuindo a cada categoria um valor numérico exclusivo. Por exemplo, se a variável experience tiver as seguintes categorias: junior, mid-level e senior, o Label Encoder atribuirá a cada categoria um valor numérico exclusivo, como 0, 1 e 2, respectivamente.

O One Hot Encoder codifica as variáveis categóricas criando uma nova variável binária para cada categoria. Por exemplo, se a variável company_location tiver as

seguintes categorias: São Paulo, Rio de Janeiro e Belo Horizonte, o One Hot Encoder criará três novas variáveis binárias: company_location_São Paulo, company_location_Rio de Janeiro e company_location_Belo Horizonte. Cada uma dessas novas variáveis binárias será igual a 1 se o ponto de dados pertencer à categoria correspondente e 0 caso contrário.

Diferenças entre Label Encoder e One Hot Encoder

A principal diferença entre Label Encoder e One Hot Encoder é que o Label Encoder codifica as variáveis categóricas para que possam ser usadas em modelos de aprendizado de máquina que só aceitam variáveis numéricas, enquanto o One Hot Encoder codifica as variáveis categóricas para que possam ser usadas em modelos de aprendizado de máquina que podem aceitar variáveis categóricas ou variáveis numéricas.

Outra diferença é que o Label Encoder pode levar a perda de informação, pois as categorias são codificadas como valores numéricos exclusivos. Por exemplo, se a variável experience for codificada usando Label Encoder, o modelo de aprendizado de máquina não poderá aprender que a categoria senior é mais experiente do que a categoria mid-level.

O One Hot Encoder não leva a perda de informação, pois as categorias são codificadas criando uma nova variável binária para cada categoria. Isso permite que o modelo de aprendizado de máquina aprenda as relações entre as categorias.

IV) Pré-Processamento

O código cria três novos conjuntos de dados para os modelos de regressão:

StandardScaler: Este conjunto de dados é criado usando o escalador StandardScaler. O escalador *StandardScaler* remove a média e escala os dados para unit variance. Isso pode ajudar os modelos de regressão a aprender padrões mais complexos nos dados e a evitar o overfitting.

MinMaxScaler: Este conjunto de dados é criado usando o escalador MinMaxScaler. O escalador *MinMaxScaler* escala os dados para um intervalo específico, geralmente entre 0 e 1. Isso pode ajudar a melhorar o desempenho dos modelos de regressão que são sensíveis à escala dos dados.

Normalizer: Este conjunto de dados é criado usando o escalador Normalizer. O escalador Normalizer normaliza os dados para que cada feature tenha uma magnitude de

1. Isso pode ajudar a melhorar o desempenho dos modelos de regressão que são sensíveis à magnitude das features.

V) Redução de dimensionalidade

O código cria novas features para os modelos de regressão usando PCA, KernelPCA e LDA.

O PCA (Análise de Componentes Principais) é uma técnica de redução de dimensionalidade que transforma um conjunto de dados de alta dimensão em um conjunto de dados de baixa dimensão, preservando a maior parte da informação original.

O código usa a PCA para criar novas features para os modelos de regressão com 1/3 e 1/2 do número de features originais. Isso pode ajudar os modelos de regressão a aprender padrões mais complexos nos dados e a evitar o overfitting.

O KernelPCA é uma extensão da PCA que pode ser usada para reduzir a dimensionalidade de dados não lineares.

O código usa a *KernelPCA* para criar novas features para os modelos de regressão com 1/3 e 1/2 do número de features originais. Isso pode ajudar os modelos de regressão a aprender padrões mais complexos nos dados e a evitar o overfitting.

Já a LDA (Análise Discriminante Linear) é uma técnica de redução de dimensionalidade que é usada para discriminar entre duas ou mais classes.

O código usa a LDA para criar novas features para os modelos de regressão com 1/3 e 1/2 do número de features originais. Isso pode ajudar os modelos de regressão a aprender padrões mais complexos nos dados e a evitar o overfitting.

Normalização

O código também normaliza os dados antes de aplicar a *PCA, KernelPCA e LDA*. Isso é importante para garantir que todas as features tenham a mesma importância.

As novas features criadas pelo código podem ser usadas para treinar os modelos de regressão da mesma forma que as features originais. Para fazer isso, você pode usar a seguinte abordagem:

Carregue os dados com as novas features.

Divida os dados em conjuntos de treinamento e teste.

Treine os modelos de regressão no conjunto de treinamento.

Avalie os modelos de regressão no conjunto de testes.

Evitando overfitting

É importante notar que as novas features criadas pelo código podem aumentar o risco de overfitting. Overfitting é quando os modelos de regressão aprendem os dados de treinamento muito bem e não são capazes de generalizar para novos dados.

Para evitar o overfitting, é importante usar validação cruzada ao treinar os modelos de regressão. A validação cruzada envolve dividir os dados de treinamento em vários conjuntos e treinar os modelos de regressão em cada conjunto. Os modelos de regressão são então avaliados em um conjunto de dados de teste que não foi usado para treinar os modelos.

VI) Clusterização

O código cria novas features para os modelos de regressão usando KMeans. O KMeans é um algoritmo de clustering que agrupa os dados em um determinado número de clusters (neste caso, 25 clusters).

Para criar as novas features, o código primeiro treina um modelo KMeans nos dados. Em seguida, o código prevê o cluster a que cada ponto de dados pertence. As novas features são então criadas adicionando uma coluna aos dados que contém o cluster a que cada ponto de dados pertence.

VII) Algoritmos baseado em árvore de decisão (regressão)

Os modelos são divididos em cinco categorias:

Árvore de decisão

Random Forest

Bagging

Extra Trees

AdaBoost

XGBoost

Cada categoria contém vários modelos com diferentes parâmetros. Por exemplo, a categoria tree contém modelos com diferentes critérios de divisão (criterion), diferentes profundidades máximas (max_depth) e diferentes divisores (splitter).

O código pode ser usado para treinar e avaliar os diferentes modelos de regressão em um conjunto de dados.

VIII) Validação cruzada

O código cria primeiramente um dicionário chamado *scores* para armazenar os resultados da validação cruzada. As chaves do dicionário são Base, Modelo, Erro Quadrado Médio, Erro Absoluto e R2.

Em seguida, o código itera sobre todos os conjuntos de dados e modelos. Para cada conjunto de dados e modelo, o código faz o seguinte:

Executa a validação cruzada de 30 iterações.

Para cada iteração, o código divide os dados em conjuntos de treinamento e teste usando a função *train_test_split()*.

Treina o modelo nos dados de treinamento usando a função *fit()*.

Faz previsões nos dados de teste usando a função *predict()*.

Calcula o erro quadrático médio (MSE), o erro absoluto médio (MAE) e o R² para as previsões.

Após a conclusão da validação cruzada, o código calcula os valores médios de MSE, MAE e R² para todas as iterações.

Os resultados da validação cruzada são armazenados no dicionário "*scores*".

O código também inclui um bloco try/except para lidar com erros. Se um modelo não puder ser treinado em um conjunto de dados, o código armazenará valores NaN no dicionário scores.

No final do código, o código imprime os resultados da validação cruzada para cada conjunto de dados e modelo.

IX) Best Fits:

O código primeiro cria um modelo de regressão por ensacamento com 90 estimadores e as seguintes opções: *warm_start=True*: Isso permite que o modelo seja treinado incrementalmente, o que pode acelerar o processo de treinamento.

bootstrap=True: Isso indica que o modelo deve usar amostragem aleatória com reposição para criar os conjuntos de treinamento para cada estimador.

Em seguida, o código executa a validação cruzada de 200 iterações. Para cada iteração, o código faz o seguinte:

Divide os dados em conjuntos de treinamento e teste usando a função *train_test_split()*.

Treina o modelo nos dados de treinamento usando a função *fit()*. Faz previsões nos dados de teste usando a função *predict()*. Calcula o erro absoluto médio (MAE), o erro quadrático médio (MSE) e o R² para as previsões.

Após a conclusão da validação cruzada, o código calcula os valores médios de MAE, MSE e R² para todas as iterações.

Os resultados da validação cruzada são os seguintes:

Squared: 532.858.540,28, Absolute: 17.036,42, R²: 67,19%

Esses resultados indicam que o modelo de regressão por ensacamento está tendo um bom desempenho nos dados de teste. O R^2 de 67,19% indica que o modelo é capaz de explicar 67,19% da variação nos dados de teste.

Figura 6 - Best Fits

```python
# ACCURACY
r2 = []
absolute = []
squared = []
model = BaggingRegressor(n_estimators=90, warm_start=True, bootstrap=True)

# CROSS VALIDATION 200
for _ in range(200):

    # SPLIT TRAIN TEST
    x_train, x_test, y_train, y_test = train_test_split(datas['X'], y, test_size=.3, shuffle=True)

    # MODEL FIT
    y_pred = model.fit(X=x_train, y=y_train).predict(x_test)

    # SAVE ACCURACY
    r2.append(r2_score(y_test, y_pred))
    absolute.append(mean_absolute_error(y_test, y_pred))
    squared.append(mean_squared_error(y_test, y_pred))

print(f'Squared: {np.mean(squared):>16,.2f} | Absolute: {np.mean(absolute):>8,.2f} | R²: {np.mean(r2)*100:>6,.2f}')
```

```
Squared:   532,858,540.28 | Absolute: 17,036.42 | R²:  67.19
```

Fonte: os autores, 2023

2.4) DEFINIÇÃO E DESCRIÇÃO DAS BASES TEÓRICAS DOS MÉTODOS

A seleção de bibliotecas e ferramentas para a análise de tendências salariais em Ciência de Dados é fundamentada em sólidas considerações teóricas e práticas, visando fornecer uma base robusta para nosso projeto. As escolhas são guiadas pelos seguintes princípios:

Python como Linguagem de Programação

Reconhecemos Python como a linguagem de programação ideal para a condução da análise de dados em nosso projeto. Essa escolha é respaldada pela ampla comunidade de usuários, a presença de bibliotecas essenciais e sua flexibilidade para integração com outras tecnologias.

Pandas para Manipulação de Dados

O Pandas se destaca como uma biblioteca essencial projetada para lidar com

dados tabulares. Seu uso eficaz do Data-Frame simplifica a manipulação e transformação de dados, tornando-o uma escolha natural para a nossa análise de tendências salariais.

Matplotlib e Seaborn para Visualização

Utilizamos as bibliotecas Matplotlib e Seaborn para criar representações gráficas de alta qualidade. Matplotlib oferece versatilidade na visualização de dados, enquanto o Seaborn aprimora nossas capacidades de criação de visualizações estatísticas atraentes. Ambas são escolhas sólidas para apresentar nossos resultados de maneira informativa e impactante.

Adaptação às Necessidades do Projeto

É crucial personalizar a seleção de bibliotecas e ferramentas de acordo com as especificidades do projeto. Isso inclui considerar a natureza dos dados, os objetivos analíticos e a eficiência na realização de tarefas específicas. A adequação às necessidades é uma prioridade para garantir resultados precisos.

Documentação e Comunidade de Suporte

Garantir que as bibliotecas e ferramentas selecionadas possuam documentação detalhada e uma comunidade de suporte ativa é essencial. Esses fatores facilitam a solução de problemas e o aprendizado contínuo durante o desenvolvimento do projeto.

Revisão da Literatura

A revisão da literatura desempenha um papel crítico na escolha de bibliotecas e ferramentas, permitindo uma avaliação da relevância e eficácia das soluções em projetos semelhantes.

Em resumo, as escolhas das bibliotecas e ferramentas em nosso projeto são baseadas em uma análise aprofundada, considerando aspectos teóricos, práticos e a adaptabilidade às necessidades específicas de análise de tendências salariais em Ciência de Dados. Essas escolhas garantem a eficácia da análise e a qualidade dos resultados obtidos, contribuindo para o sucesso do projeto.

2.5) APRESENTAÇÃO DE PRODUTOS E STORYTELLING

Em projetos de Data Science, o storytelling desempenha um papel fundamental na comunicação eficaz de descobertas e insights. Ao transformar dados complexos em narrativas envolventes, o storytelling torna a análise acessível a uma variedade de públicos, inclusive aqueles sem conhecimento técnico aprofundado. A seguir, destacam-se as principais razões pelas quais o storytelling é essencial nesse contexto:

- Comunicação Eficaz:

Facilita a compreensão de insights complexos, tornando a informação acessível a públicos diversos.

- Engajamento do Público:

Cativa e mantém a atenção das partes interessadas, aumentando a receptividade às descobertas e recomendações.

- Contextualização dos Resultados:

Coloca os resultados em um contexto significativo, destacando a relevância dos padrões ou tendências identificadas.

- Geração de Empatia:

Utiliza histórias e exemplos para criar uma conexão emocional, relacionando-se com as necessidades e desafios do público.

- Influência nas Decisões:

Persuade apresentando dados de maneira convincente, orientando decisões com base em insights claros.

- Clareza na Narrativa:

Estrutura a análise de dados em uma narrativa coesa, proporcionando uma compreensão lógica e contínua.

- Orientação para Ação:

Direciona a atenção para ações recomendadas, transformando insights em iniciativas acionáveis.

- Memorabilidade:

Histórias são mais facilmente lembradas, contribuindo para uma retenção eficaz da informação ao longo do tempo.

- Construção de Confiança:

Reforça a credibilidade da análise de dados, construindo confiança nas conclusões apresentadas.

- Alinhamento com Objetivos Estratégicos:

Garante que a análise esteja alinhada com os objetivos estratégicos da empresa, conectando as descobertas à visão organizacional.

Em resumo, o storytelling não apenas simplifica a ciência de dados, mas também agrega valor ao transformar insights em ações tangíveis, influenciando positivamente a percepção e utilização das descobertas. O rascunho do modelo de storytelling se encontra no apêndice c deste documento.

2.6) O método analítico definido na etapa anterior aplicado à base de dados escolhida

Na etapa anterior, definimos que o método analítico seria a Análise Exploratória de Dados (AED), focando na análise estatística das tendências salariais em Ciência de Dados em 2023. Essa abordagem envolveu a coleta, limpeza e preparação dos dados, seguida por análises estatísticas e visualizações para entender a distribuição de salários, fatores que afetam os salários e identificação de tendências ao longo dos anos. A AED nos permitiu obter uma compreensão sólida da estrutura dos dados e extrair insights preliminares importantes para o projeto.

Medidas de acurácia, usando os métodos definidos na etapa anterior

Embora as medidas de acurácia não fossem o foco principal da Análise Exploratória de Dados, realizamos algumas verificações para garantir a qualidade dos dados. Verificamos a integridade dos valores nulos e validamos os tipos de dados das colunas. Utilizamos medidas de tendência central, como média, mediana e moda, para resumir a distribuição salarial. Além disso, calculamos o crescimento salarial ao longo dos anos e a média salarial com base em variáveis categóricas, como nível de experiência, tipo de emprego e localização.

I) Descrição dos resultados preliminares, apresentando um produto gerado e rascunhando um possível modelo de negócios:

Os resultados preliminares da Análise Exploratória de Dados forneceram uma visão abrangente das tendências salariais em Ciência de Dados em 2023. Identificamos padrões de crescimento nos salários, evidenciamos a influência da experiência e de variáveis categóricas nos salários e destacamos diferenças significativas com base na localização. Além disso, exploramos o impacto da imigração na mobilidade dos profissionais e na remuneração.

Como produto da análise exploratória, geramos gráficos e visualizações que facilitam a comunicação dos resultados e insights para as partes interessadas. Em relação a um possível modelo de negócios, consideramos a oferta de serviços de consultoria em remuneração personalizados com base nas tendências salariais e nas estratégias de retenção identificadas. Além disso, a empresa Data-Trend Insights pode desenvolver soluções de software especializadas para a gestão de remuneração em Ciência de Dados, fornecendo ferramentas de análise de dados, modelagem de salários e previsão de tendências salariais.

3. CONCLUSÃO

Em meio ao desafio constante de reter talentos em Ciência de Dados, a jornada da DataTrend Insights foi muito mais do que uma análise de tendências salariais. Fomos além dos números, mergulhando em histórias que os dados contavam.

A Análise Exploratória de Dados não foi apenas uma busca por padrões; foi uma narrativa visual de crescimento salarial, influência da experiência e nuances geográficas. Os gráficos não eram apenas representações; eram capítulos de uma história que revelava a complexidade das dinâmicas salariais em 2023.

Com a metodologia Random Forest, transformamos árvores de decisão em raízes de conhecimento, explorando as relações entre títulos de emprego e salários de maneira inovadora. O aprendizado de máquina não era apenas código; era a arte de prever futuros salariais com base em um passado minuciosamente compreendido.

Ao enviar o Relatório Inicial, não apenas compartilhamos descobertas, mas entregamos um enredo de insights que orientará estrategistas e líderes na tomada de decisões. A oferta de consultoria personalizada em remuneração e soluções de software não é apenas um modelo de negócios; é o compromisso de continuar a escrever capítulos significativos na história da gestão salarial em Ciência de Dados.

A apresentação em vídeo do projeto baseado no storytelling encontra-se no apêndice d, em formato não listado na plataforma youtube.

4. REFERÊNCIAS BIBLIOGRÁFICAS

I) KUMARDATALAB, harish. **Data Science Salary 2021 to 2023**. Kaggle, 2023. Disponível em: https://www.kaggle.com/datasets/harishkumardatalab/data-sciencesalary-2021-to-2023. Acesso em: 10 set. 2023.

III) BEHESHTI, Nima. **Random Forest Regression**. Medium, 2022. Disponível em: https://towardsdatascience.com/random-forest-regression-5f605132d19d. Acesso em: 11 out. 2023

IV) **What is a Decision Tree?.** IBM. Disponível em: https://www.ibm.com/topics/decisiontrees. Acesso em: 11 out. 2023

V) OLIVEIRA MELLO, LEONARDO. ***Ciência de Dados Aplicada a Gestão de Projetos de Quality Assurance***. 2021. 39 p. Dissertação de especialista — UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, Porto Alegre, 2021. Disponível em: https://lume.ufrgs.br/bitstream/handle/10183/231570/001133187.pdf?sequence=1. Acesso em: 3 nov. 2023.

VI) MARTINS DE PAULA, Lauro Cássio. **ESTUDO E APLICAÇÃO DE BIG DATA E MACHINE LEARNING EM CIÊNCIA DE DADOS**. 2019. 5 p. TCC — INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA BAHIA, Santo Antônio de Jesus, 2019. Disponível em: https://portal.ifba.edu.br/santoantonio/pesquisa/arquivos/ProjetoESTUDOEAPLICAODEBIGDATAEMACHINELEARNINGEMCINCIADEDADOS2.pdf. Acesso em: 3 nov. 2023.

VII) FERREIRA DE PAIVA, ATILAS. **APLICAÇÃO DE DATA SCIENCE COMO FERRAMENTA DE APOIO A TOMADA DE DECISÃO ORIENTADA POR DADOS**. 2018. 24 p. MONOGRAFIA DE ESPECIALIZAÇÃO — UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ, PATO BRANCO, 2018. Disponível em: https://repositorio.utfpr.edu.br/jspui/bitstream/1/25166/1/PB_ESEP_III_2018_5.pdf. Acesso em: 4 nov. 2023.

5. APÊNDICES

APÊNDICE A - CÓDIGO PYTHON USADO PARA A ANÁLISE EXPLORATÓRIA DE DADOS.

APÊNDICE B - APRENDIZADO DE MÁQUINA: A METODOLOGIA RANDOM FOREST

APÊNDICE C - REPOSITÓRIO: GITHUB

APÊNDICE D - APRESENTAÇÃO EM VÍDEO NO YOUTUBE: VÍDEO