CENTRO UNIVERSITÁRIO SALESIANO DE SÃO PAULO

Engenharia de Computação

Kawã Sampaio

Murilo Ribeiro

Rafael Freitas

INTELIGENCIA ARTIFICIAL

Prof. Me. José Walmir G. Duque

ÁRVORE DE DECISÃO / REGRESSÃO LINEAR

Lorena, SP

2023

Sumário

1. INTRODUÇÃO

Neste trabalho, exploraremos dois métodos fundamentais de análise de dados, a regressão linear e a árvore de decisão, aplicados a um conjunto de dados relacionado a acidentes vasculares cerebrais (AVCs). Os AVCs são eventos de saúde crítica que têm um impacto significativo na qualidade de vida das pessoas e representam um desafio importante para a medicina preventiva e o sistema de saúde como um todo.

Exploraremos em detalhes esses dois métodos, descrevendo suas aplicações específicas no contexto de dados de AVC. Vamos considerar como eles podem ser utilizados para analisar e compreender as relações entre variáveis independentes e a ocorrência de AVCs, bem como avaliar a eficácia de cada método na modelagem desse problema de saúde crítico. O objetivo é fornecer uma visão abrangente das técnicas de regressão linear e árvore de decisão aplicadas a um conjunto de dados de AVC, destacando seu potencial na identificação de fatores de risco e tomada de decisões informadas na área da saúde.

2. BUSSINES UNDERSTANDING

Objetivos de Negócio:

O principal objetivo de negócio para este projeto de Data Science é desenvolver um modelo de previsão de AVC que possa identificar indivíduos em risco de sofrer um acidente vascular cerebral com base em dados médicos e demográficos. Isso permitirá a implementação de medidas preventivas e tratamentos mais eficazes, reduzindo assim a incidência de AVC e melhorando a saúde pública.

Tarefas de Data Science Elegíveis:

Com base nos objetivos de negócio, as tarefas de Data Science elegíveis a priori incluem:

- Análise exploratória de dados: Explorar o dataset para entender suas características, distribuições, valores ausentes e possíveis relações entre variáveis.

- Pré-Processamento de Dados: Antes da análise, os dados coletados devem passar por etapas de limpeza, transformação e normalização. Isso inclui o tratamento de valores ausentes, a padronização de formatos e a identificação de possíveis outliers.

- Modelagem Preditiva: O sistema deve ser capaz de desenvolver um modelo de Machine Learning capaz de prever o risco de AVC com base nos dados disponíveis. Isso requer a seleção de algoritmos adequados, o treinamento do modelo e a otimização de seus parâmetros.

- Avaliação de Modelos: Os modelos desenvolvidos precisam ser avaliados quanto à sua precisão e desempenho. Isso envolve a aplicação de métricas adequadas, como acurácia, sensibilidade e especificidade, para determinar a eficácia do modelo.

- Implantação do modelo: Implementar o modelo de previsão em um ambiente de produção para uso prático na identificação de riscos de AVC.

Funcionalidades e Regras de Negócio:

- Coleta de Dados: O sistema deve ser capaz de coletar informações médicas e demográficas dos pacientes de forma segura e precisa. Isso envolve a integração de fontes de dados, validação de dados e garantia de que as informações estejam atualizadas.

- Implementação em Produção: Após o desenvolvimento e a validação do modelo, ele deve ser implementado em um ambiente de produção. Isso inclui a integração com sistemas de saúde existentes e a disponibilização do modelo para uso prático na identificação de riscos de AVC.

- Privacidade de Dados: É crucial garantir a privacidade e a segurança dos dados dos pacientes, cumprindo regulamentações de proteção de dados, como a Lei Geral de Proteção de Dados (LGPD).

- Comunicação com Profissionais de Saúde: Deve haver um mecanismo eficaz de comunicação para alertar profissionais de saúde sobre pacientes em risco, possibilitando intervenções oportunas.

- Essas funcionalidades e regras de negócio são fundamentais para o desenvolvimento de um sistema eficaz de previsão de AVC, visando à melhoria da saúde pública e à redução da incidência desse grave problema de saúde.

Atores Principais:

Os principais atores envolvidos neste projeto incluem:

- Cientistas de Dados: Responsáveis pela análise, modelagem e avaliação dos dados.

- Profissionais de Saúde: Usuários finais que usarão os resultados do modelo para tomar decisões clínicas.

- Pacientes: Indivíduos cujos dados médicos e demográficos estão sendo usados no modelo.

- Administradores de Sistema: Responsáveis pela implementação e manutenção do sistema.

Requisitos Não Funcionais:

Alguns requisitos não funcionais essenciais para este projeto incluem:

- Segurança dos Dados: Garantir a segurança e a confidencialidade dos dados dos pacientes.

- Desempenho: O sistema deve ser capaz de responder rapidamente, especialmente na implementação em tempo real.

- Escalabilidade: Deve ser capaz de lidar com grandes volumes de dados à medida que o sistema é adotado em larga escala.

- Interpretabilidade do Modelo: O modelo deve ser interpretável para que os profissionais de saúde possam entender e confiar em suas previsões.

- Conformidade Legal: Cumprir todas as regulamentações e leis de proteção de dados relevantes.

- Facilidade de Manutenção: O sistema deve ser de fácil manutenção e atualização.

3. Data Understanding e Data Preparation

Features presentes no Dataset

VARIÁVEL	TIPO	DOMÍNIO DE VALORES
gender	Categórica	"Masculino", "Feminino"
age	Numérica	Valor positivo representando idade
hypertension	Binária	0 (Não possui hipertensão) ou 1 (Possui hipertensão)
heart_disease	Binária	0 (Não possui doença cardíaca) ou 1 (Possui doença cardíaca)
ever_married	Categórica	"Sim" ou "Não"
work_type	Categórica	"Privado", "Autônomo", "Emprego Público", "Criança", "Nunca Trabalhou"
Residence_type	Categórica	"Urbano" ou "Rural"
avg_glucose_level	Numérica	Valores numéricos
bmi	Numérica	Valores numéricos
smoking_status	Categórica	"Fumava anteriormente", "Nunca fumou", "Fuma", "Desconhecido"
stroke	Binária	0 (Não teve AVC) ou 1 (Teve AVC)

Primeiro Target (BMI): Preenchimento de Valores Faltantes no BMI com Regressão Linear

O BMI é uma medida importante da saúde, mas frequentemente, em dados do mundo real, alguns valores podem estar ausentes. Para resolver esse problema, foi utilizado um modelo de regressão linear. Nesse modelo, o BMI foi escolhido como a variável alvo (target), enquanto outras features foram usadas como preditores para estimar o BMI ausente. Isso permitiu preencher os valores faltantes de BMI de forma apropriada, aumentando a integridade do conjunto de dados e garantindo que ele estivesse completo para análises posteriores.

Segundo Target (Stroke): Previsão de AVC com Árvore de Decisão

Após o preenchimento bem-sucedido dos valores faltantes do BMI, a análise de dados avançou para o próximo objetivo: prever a ocorrência de acidente vascular cerebral (stroke). Para isso, foi aplicada uma técnica de aprendizado de máquina, conhecida como árvore de decisão. A variável alvo agora era a presença ou ausência de AVC, enquanto outras features foram usadas como atributos para a árvore de decisão.

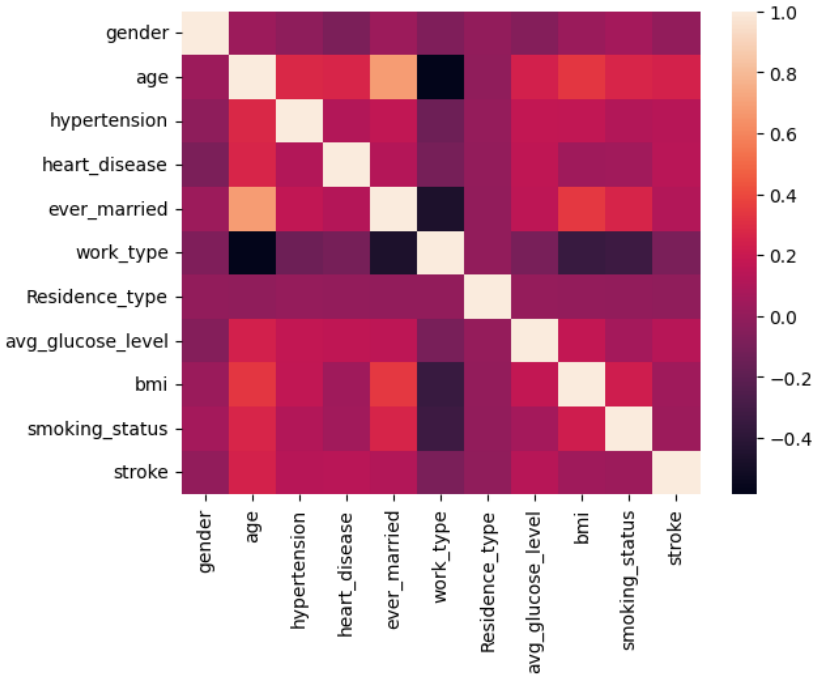
Preparação dos Dados

A maioria dos algoritmos de aprendizado de máquina, como regressão linear e árvores de decisão, requer que as entradas sejam numéricas. Portanto, as variáveis categóricas precisam ser convertidas para que esses modelos possam operar.

```
df.gender = df.gender.replace({'Male':0,'Female':1,'Other':-1})
df.ever_married = df.ever_married.replace({'Yes':1,'No':0})
df.work_type = df.work_type.replace({'Self-
employed':0,'Private':1,'Govt_job':2,'children':3, 'Never_worked':4})
df.Residence_type = df.Residence_type.replace({'Rural':1,'Urban':0})
df.smoking_status = df.smoking_status.replace({'smokes':1,'never smoked':0,
'formerly smoked':-1, 'Unknown': -2})
```

4. Modeling e Evaluation – Regressão Linear (Preenchimento dos BMI's vazios)

Modeling




```
# Selecionando as variáveis explanatórias para o modelo (conjunto X)
#Filtrado apenas as linhas com BMI != NaN para treinamento do modelo
filteredDF = df[~df.bmi.isna()]

# age e ever_married parece, os mais promissores pois a correlação é maior
X = filteredDF[['age']]
# Selecionando a variável dependente (target) para o modelo (conjunto Y)
Y = filteredDF[['bmi']]
```

```
#Modeling
# ------------------------------------------------
model = LinearRegression()
model.fit(X,Y)
# Cálculo do Coeficiente de Determinação R2 (quanto mais próximo de 1, melhor o modelo!)
print('Coeficiente de Determinação:', model.score(X, Y))


# Admitindo a Equação da Regressão Linear: f(x) = b₀ + b₁x
print('b0 ou intercept:', model.intercept_)
print('b1 ou coeficiente (slope):', model.coef_)


# Aplicando o modelo construído nos dados (conjunto X)
```

```
predictions = model.predict(X)


# Cálculo de MSE (Mean Squared Error), MAE (Mean Absolute Error) e RMSE (Root Mean Square Error)
print('MAE:', metrics.mean_absolute_error(Y, predictions))
print('MSE:', metrics.mean_squared_error(Y, predictions))
print('RMSE:', np.sqrt(metrics.mean_squared_error(Y, predictions)))
```

Coeficiente de Determinação: 0.11115422317700019

b0 ou intercept: [23.91679268]

b1 ou coeficiente (slope): [[0.11609474]]

MAE: 5.560063542208871

MSE: 54.81849508911304

RMSE: 7.403951315960488

```
# Utilizando os dados do modelo para predizer os valores vazios de BMI baseados em AGE

toPredict = df[df.bmi.isna()].copy()['age'] # Copiando apenas os dados de AGE dos valores com BMI == NaN

newPredictions = []

# Predizendo os valores de BMI
for age in toPredict:
    newPredictions.append((model.intercept_ + model.coef_* age))


bmi_empty = df[df.bmi.isna()] # Criado dataframe apenas com BMI === NaN (201 linhas)
bmi_empty['bmi'] = newPredictions # adicionamos a coluna predicted BMI ao dataframe de teste
dfWithoutBMI = df[~df.bmi.isna()] # filtrando apenas as linhas sem NaN do DF principal

bmi_empty = bmi_empty.explode('bmi').explode('bmi')
predictedDataframe = pd.concat([bmi_empty, dfWithoutBMI]) # concatenando os DF (4909 + 201)

predictedDataframe # Dataframe completo
```

Os resultados apresentados indicam um coeficiente de determinação (R^2) baixo, erro absoluto médio (MAE) e raiz do erro quadrático médio (RMSE) relativamente altos em um modelo de regressão linear. Isso significa que o modelo não se ajusta bem aos dados e tem dificuldade em fazer previsões precisas. No entanto, há situações em que esses resultados podem ser aceitáveis.

Por exemplo, em algumas áreas de pesquisa ou negócios, os dados podem ser intrinsecamente ruidosos ou altamente variáveis, tornando difícil obter um ajuste perfeito. Nesses casos, o modelo pode servir como uma primeira aproximação ou como uma ferramenta exploratória para entender a relação entre as variáveis. É importante ressaltar que, durante a análise, foi observado que a feature "age" apresentou a melhor correlação possível com o BMI, com um coeficiente de correlação de 0.3. Isso sugere que a idade é a variável mais relevante disponível para explicar a variação no BMI.

Portanto, embora esses resultados possam não ser ideais, considerando a limitação das variáveis disponíveis e a correlação encontrada com a idade, eles podem ser a melhor opção disponível. Desde que sejam interpretados com cautela e que suas limitações sejam reconhecidas, esses resultados ainda podem fornecer insights valiosos, mesmo que não atinjam altos padrões de precisão.

5. Modeling e Evaluation – Árvore de Decisão (Previsão de Stroke)

Modeling

```
# Decision Tree - Modeling - Stroke
# ------------------------------------------------

X_train = predictedDataframe[['age','gender','bmi','heart_disease',
'avg_glucose_level', 'bmi']].copy() # Selecionado parametros
Y_train = predictedDataframe[['stroke']].copy() # Selecionado BMI como target

X_train, X_test, Y_train, Y_test = train_test_split(X_train, Y_train,
test_size=0.3, random_state=1) #Aplicando a técnica de “Percentage Split

clf = DecisionTreeClassifier(criterion='entropy') # declarado nosso modelo
clf = decisionTreeClassifier.fit(X_train, Y_train) # treinando o modelo
baseado nas linhas com BMI != NaN (4909 linhas)

Y_pred = decisionTreeClassifier.predict(X_test)
```

Evaluation

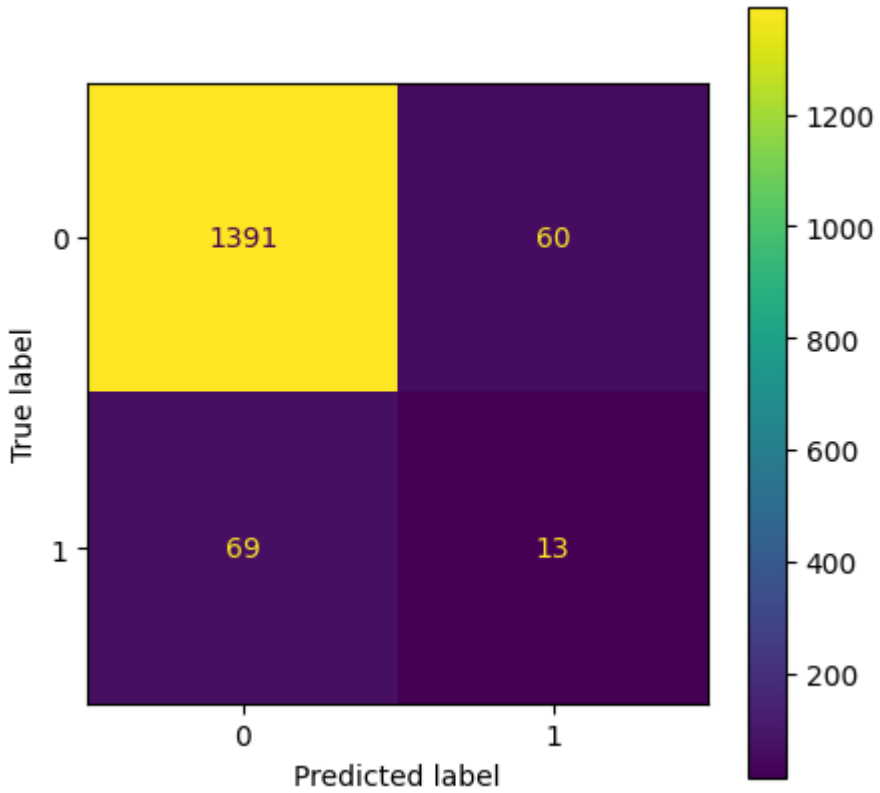
```
# Decision Tree - Stroke - Evaluation
# ------------------------------------------------

accuracy = metrics.accuracy_score(Y_pred, Y_test)
accuracy
print('Acurácia: {:.2f}'.format(accuracy))

cm = confusion_matrix(Y_test, Y_pred, labels=decisionTreeClassifier.classes_)
disp = ConfusionMatrixDisplay(confusion_matrix=cm,
display_labels=decisionTreeClassifier.classes_)

fig, ax = plt.subplots(figsize=(5,5))
disp.plot(ax=ax)
Acurácia: 0.92
```

Matriz de Confusão



A acurácia da árvore de decisão é de 0,92, o que significa que o modelo está correto em suas previsões em aproximadamente 92% dos casos. Isso é um bom indicativo de que o modelo está desempenhando bem na tarefa de classificação.

6. CONCLUSÃO

Nosso trabalho ilustrou como a análise de dados em saúde pode ser um processo interativo e abrangente. Começando com o tratamento de valores faltantes usando regressão linear e depois aplicando modelos de aprendizado de máquina, como a árvore de decisão, podemos melhorar a qualidade dos dados e obter insights valiosos sobre questões de saúde críticas, como a previsão de AVC. Essa abordagem demonstra a importância de utilizar uma variedade de técnicas analíticas para abordar os desafios complexos associados aos dados, contribuindo para uma tomada de decisão mais informada e eficaz na área de aplicação.

No entanto, é fundamental reconhecer que, embora tenhamos alcançado resultados promissores, a dificuldade em obter um ajuste perfeito dos dados foi uma realidade que acompanhou nosso desenvolvimento. Portanto, é crucial que os resultados obtidos sejam interpretados com cautela e que sejam realizadas validações adicionais.