ESCOLA ESTADUAL GREGORIANO CANEDO

Avaliação do desempenho de algoritmos de aprendizagem de máquina no diagnóstico da Diabetes Tipo 2

Monte Carmelo, MG

2024

Maria Fernanda Marques Nogueira
Carolina Beatriz Rosa Madóglio
Mariane Machado Cardoso Luiz Pires.

Orientador: Doriane dos Santos Honorato Moura

Avaliação do desempenho de algoritmos de aprendizagem de máquina no diagnóstico da Diabetes Tipo 2

Relatório apresentado à 8ª FEMIC - Feira Mineira de Iniciação Científica.
Orientação da Professora. Doriane dos Santos Honorato Moura.

Monte Carmelo, MG

2024

RESUMO

Este estudo visa prever a presença de diabetes tipo 2 em mulheres Pima com base em dados médicos utilizando algoritmos de aprendizado de máquina na plataforma Google Colab com linguagem Python.O conjunto de dados do Instituto Nacional de Diabetes e Doenças Digestivas e Renais foi utilizado, contendo informações de mulheres indígenas Pima com idade superior a 21 anos. A análise exploratória detalhada do banco de dados foi realizada para compreender as variáveis e identificar padrões. Algoritmos de aprendizado de máquina, como Random Forest e Gradiente Descendente, foram implementados para construir modelos preditivos. A avaliação do desempenho dos modelos foi realizada utilizando métricas como acurácia, sensibilidade, especificidade e AUC.Os resultados demonstraram que o modelo Ensemble, combinando Random Forest e Gradiente Descendente, obteve a melhor performance, com acurácia acima de 86%, superando os modelos individuais de Regressão Logística. As métricas de sensibilidade e especificidade também apresentaram valores satisfatórios, indicando a capacidade dos modelos em identificar corretamente pacientes com e sem diabetes tipo 2. O estudo conclui que a combinação de algoritmos de aprendizado de máquina, especialmente o modelo Ensemble, apresenta um desempenho superior na previsão de diabetes tipo 2 em mulheres Pima. A análise exploratória detalhada do banco de dados foi fundamental para o sucesso da modelagem. Essa abordagem pode auxiliar profissionais de saúde na detecção precoce da doença e na tomada de decisões mais assertivas para o tratamento.

Palavras-chave: Diabetes tipo 2, Fatores de risco, Aprendizado de máquina, Modelo de previsão, Acurácia, Intervenções precoces, Prevenção, Tratamento.

SUMÁRIO

1 INTRODUÇÃO

O diabetes tipo 2 é um problema de saúde pública em todo o mundo. Segundo as estimativas da International Diabetes Federation IDF, 463 milhões de pessoas têm diabetes no mundo em 2021. O diabetes tipo 2 contribui internacionalmente com a parcela da maioria dos abandonos, resistência à insulina e aumento da glicose sanguínea. O aumento da incidência é motivo de preocupação por causa das complicações graves associadas ao diabetes tipo 2, incluindo aterosclerose coronariana, insuficiência renal, perda visual e neuropatia até a morte.

Dada a escala do problema, é importante entender que variáveis contribuem para taxas crescentes de diabetes tipo 2. Os fatores genéticos e ambientais adquiridos, os fatores de risco dependentes do estilo de vida, alimentação, fator de risco entre outros, a obesidade e a inatividade, a idade e a história familiar estão entre os mais importantes. Pesquisadores identificaram fatores socioeconômicos e geográficos que influenciam o risco de diabetes tipo 2. É essencial analisar essas variáveis para formular estratégias de prevenção aprimoradas e tratamento do diabetes tipo 2.

O diabetes mellitus é uma condição metabólica que envolve desequilíbrio no metabolismo de açúcar, ou glicose, presente no sangue. Conhecidas como tipos 1, 2 e diabetes gestacional. O diabetes atua como um tipo imune e metabólico no tipo 1, no qual a insulina produz pelo pâncreas é eliminada pelo organismo por acidente e metabólica, uma falha metabólica na produção ou utilização de insulina no restante. O diabetes gestacional, um tipo temporário e o tipo 1 menos conhecido, é onde a gestante contém alta taxa de açúcar no sangue que descarta após o parto e outras.

Classificado como autoimune, o diabetes tipo 1 ocorre quando o sistema imunológico ataca e destrói as células beta do pâncreas, que são encarregadas de produzir insulina. A deficiência do hormônio desencadeia hiperglicemia crônica que exigirá a administração de insulina exógena para a manutenção da glicose e da saúde.

> “Diabetes tipo 1 é uma doença autoimune crônica em que o sistema imunológico ataca e destroi as células beta. A deficiência de insulina crônica leva à hiperglicemia, o que pode levar a várias complicações graves, incluindo cetoacidose diabética, doenças cardíacas, renais e oculares e neuropatias”. (American Diabetes Association, 2023).

Uma doença metabólica, o diabetes tipo 2 envolve resistência à insulina e produção inadequada de insulina pelo pâncreas. Como resultado, a resistência da insulina faz com que o

próprio corpo não aceite a glicose em quantidade suficiente. Por sua vez, o açúcar acumula-se na corrente sanguínea, resultando em hiperglicemia.

> "Diabetes tipo 2 é uma condição crônica que ocorre quando o corpo não produz insulina suficiente ou impede uma ação efetiva da insulina. Caso contrário, existe um acúmulo mais elevado de glicose. Ele pode resultar em uma série de outras condições, como doenças cardíacas, derrame, doença renal, cegueira e amputações." (World Health Organization, 2023).

O diabetes gestacional é uma condição que ocorre apenas durante a gravidez. Geralmente se desenvolve no segundo ou terceiro trimestre e é caracterizado por níveis elevados de açúcar no sangue. Na maioria dos casos, desaparece após o parto. No entanto, as mulheres que apresentam diabetes gestacional têm um risco aumentado de desenvolver diabetes tipo 2, bem como os filhos.

> "Mulheres com diabetes gestacional têm um risco aumentado de desenvolver diabetes tipo 2, bem como seus filhos. Além disso, o diabetes gestacional aumenta o risco de complicações da gravidez, como parto prematuro, natimorto e macrosomia fetal." (American College of Obstetricians and Gynecologists, 2023).

Exercício físico é uma ferramenta imprescindível para melhorar a qualidade de vida do portador de diabetes tipo 2, tornando-se um meio preventivo de complicações e promotor de saúde. Comunidades envolvidas na atividade física regular no diabético podem controlar e prevenir a doença de forma significativa. No entanto, deve ser enfatizado que a prática regular deve ser supervisionada por um profissional qualificado de saúde e monitorada conforme a individualidade de cada pessoa.

O banco de dados público diabetes.csv é uma valiosa fonte de diabetes para análise e estudo. Com um grande número de variáveis, como idade, índice de massa corporal, pressão arterial, sangue e medidas de glicose familiaridade de um histórico individual de diabetes, este conjunto de dados completo será utilizado para a análise estatística e a construção de modelos preditivos para adquirir uma noção aprofundada dos fatores de risco associados ao diabetes (Pima Indians Diabetes Database, nd).

A ciência de dados é uma disciplina interdisciplinar que se dedica ao desenvolvimento e aplicação de métodos para extrair conhecimentos e insights de agregados de dados em larga escala. Os cientistas de dados empregam técnicas avançadas, como estatísticas, análise quantitativa, análise preditiva, aprendizado de máquina, mineração de dados e visualização de dados, para identificar padrões e tendências em dados complexos. Esses avanços são aplicáveis a muitos setores, incluindo empresas, saúde, finanças, governos e ciência (O'Neil, 2017).

A ciência de dados está a transformar a área da saúde, tornando possível desenvolver diagnósticos, tratamentos e intervenções de prevenção de doenças mais eficazes. A análise dos grandes volumes de dados de saúde, como os registos médicos eletrônicos, as imagens de diagnóstico e os dados genómicos, tem permitido aos cientistas identificarem os padrões e as tendências anteriormente imperceptíveis. Isso tem catalisado a investigação de novos fármacos e terapias mais eficientes e personalizados para a saúde de cada um. A ciência de dados está também a ser utilizada para desenvolver novas ferramentas de prevenção de doenças, incluindo sistemas de rastreamento e rastreio de doenças. Consequentemente, a ciência de dados pode ter um impacto significativo na melhoria da saúde e do bem-estar das pessoas (Kohane, 2015).

No contexto da investigação sobre diabetes, o machine learning é um aliado inestimável. Trata-se de um subconjunto da inteligência artificial que envolve a computação de programas que aprendem e fazem previsões ou tomam decisões sem a necessidade de programação explícita. Algoritmos de machine learning podem ser executados na diabetes.csv para identificar padrões, fazer previsões e extrair informações úteis curriculo sobre a progresão da doença.

Uma das pesquisas mais recentes publicada na revista Nature Medicine descreve um modelo de aprendizado de máquina que emprega diversos algoritmos, incluindo Random Forest, Gradient Descent e Support Vector Machine. Este modelo consegue determinar de maneira acurada a probabilidade de um paciente apresentar diabetes tipo 2 com base nos dados médicos deste paciente após ele ser treinado em um conjunto de dados de mais de 100.000 pessoas. Previu o desenvolvimento da doença em 90% dos casos. Os cientistas esperam que essa ferramenta possa ser utilizada para distinguir grupos de pacientes de alto risco para o diabetes, o que permitirá aplicar intervenções preventivas a eles (Zhang et al., 2020).

Algoritmo Random Forest é um tipo de aprendizado de máquina supervisionado que pode ser usado para classificação, regressão e agrupamento de tarefas. É baseado na construção de uma floresta de várias árvores de decisão, nas quais cada árvore é treinada em um subconjunto aleatório dos dados. O algoritmo faz previsões ao somar o resultado de cada árvore individual. Isso torna o Random Forest uma ferramenta poderosa e versátil que pode ser aplicada ao trabalho com grandes conjuntos de dados, sendo frequentemente utilizada em projetos de ciência de dados. Em medicina, o Random Forest foi aplicado à previsão do risco

de doenças, identificação de padrões nos dados médicos, elaboração de tratamentos (Leo Breiman, 2001).

Quando se trata de análise de dados e modelagem preditiva, a linguagem de programação Python é uma ferramenta poderosa e versátil que é executada na plataforma do Google Colab. Essa integração oferece um ambiente ideal para que cientistas de dados, engenheiros e pesquisadores de todos os níveis de experiência explorem o potencial dos dados com simplicidade e eficiência (Van der Walt, 2011).

O Google Colab é um espaço de trabalho de notebook na nuvem que é totalmente gratuito. Ele contém Python pré-instalado no sistema, de modo que você não precisa gastar tempo instalando softwares (Van der Walt, 2011).

Python é prontamente adotado por iniciantes porque é simples e claro, e utilizá-lo é mais fácil do que outras linguagens. Essa característica ajuda no aprendizado e elimina a barreira de entrada para amadores interessados, tornando-os aptos a conduzir análises (Van der Walt, 2011).

Em suma, o Python no Google Colab é uma ferramenta poderosa e acessível, devido à gratuidade ao uso, à versatilidade da linguagem, a simplicidade e a comunidade vibrante para análise de dados e modelagem preditiva, e é uma combinação ideal e uma escolha sábia para cientistas de dados, engenheiros e cientistas que desejam revelar dados e fazer previsões.

2 JUSTIFICATIVA

O presente projeto de Iniciação Científica apresenta como proposta o desenvolvimento de um modelo preditivo de diabetes tipo 2 a partir do uso do aprendizado de máquina. Tal proposta insere-se na área da saúde e tem por objetivo colaborar para o diagnóstico antecipado da doença, a fim de permitir a realização de intervenções mais assertivas e requalificar a vida dos pacientes.

O diabetes tipo 2 é uma das principais doenças crônicas não transmissíveis e representa um sério problema de saúde pública pela quantidade de pessoas acometidas em todo o mundo. A detecção da patologia mediante o exame clínico precoce é determinante para o êxito do tratamento e para a prevenção do agravo dos sintomas. Os algoritmos que

identificam padrões ao analisar grandes dados possibilitam a obtenção de taxas preditivas com elevada precisão.

A pandemia de COVID-19 foi associada a vários problemas de saúde pública, um dos quais foi o aumento da incidência de diabetes tipo 2 em todo o mundo. A literatura sugere que o SARS-CoV-2 tem a capacidade de infetar diretamente as células beta do pâncreas, responsáveis pela secreção de insulina, resultando em disfunção dessas células e subsequente desenvolvimento de diabetes tipo 2 (Felix, 2023). Devido ao impacto da pandemia, maior estresse cultural, sedentarismo e alimentação inadequada, concomitantemente com altas taxas de diabetes e obesidade, o diabetes tipo 2 se tornou um problema epidemiológico insustentável (Felix, 2023). É importante compreender a correlação entre COVID-19 e diabetes tipo 2 para desenvolver estratégias que possam beneficiar os pacientes nos estágios pós-pandêmicos e reduzir a morbidade e a mortalidade associadas.

Utilizando os mais recentes algoritmos de aprendizagem de máquina, este projeto de estudo tem como objetivo desenvolver um modelo preditivo inovador e eficaz. O modelo utilizará uma série de datasets, incluindo dados clínicos, históricos de saúde e características de estilo de vida dos pacientes. O objetivo do projeto é o treinamento e validação rigorosos do modelo para garantir sua eficácia e precisão a longo prazo.

3 OBJETIVOS

3.1 Objetivo geral

Aplicar o modelo preditivo de aprendizado de máquina para o diagnóstico do diabetes tipo 2, na linguagem Python.

3.2 Objetivos específicos

- Exploração de artifícios da inteligência artificial (IA) com aprendizado de máquina ou Machine Learning.
- Método Ensemble com algoritmos Random Forest e gradiente descendentes para o processo de otimização na melhoria da acurácia do modelo preditivo e do processo facilitador do diagnóstico do diabetes tipo 2.

- Demonstrar a importância do componente curricular de disciplina ciência de dados no desenvolvimento da multidisciplinaridade dos conteúdos Novo Ensino Médio.
- Ressaltar as metodologias de pesquisa na área das Ciências da Natureza e da Matemática e suas Tecnologias, utilizando um banco de dados, artigos científicos.

4 METODOLOGIA

Neste estudo, a análise de mineração de dados foi realizada usando a plataforma do Google Colab para explorar dados e construir modelos de aprendizado de máquina.

4.1. ACESSO À PLATAFORMA KAGGLE

Quando consideramos o domínio da ciência de dados, encontrar conjuntos de dados confiáveis e que atendam a um padrão internacional é um dos maiores desafios. Isto, por sua vez, pode prejudicar a qualidade de um trabalho de pesquisa, tornar difícil a replicação dos resultados e limitar o conhecimento adquirido. A plataforma Kaggle foi uma escolha óbvia para superar esta restrição e enriqueceu o estudo com uma fonte de grandes conjuntos de dados de qualidade juntamente com um conjunto de ferramentas e documentação para modelagem e análise de dados. Kaggle disponibiliza uma base de dados massiva que abrange completamente todas as disciplinas que um especialista em dados possa exigir. Estes conjuntos de dados são muitas vezes vedados/curados, com uma extensa documentação associada tornando mais fácil compreender e lidar com esse tipo de informação.

4.2. INTRODUÇÃO À MINERAÇÃO DE DADOS

No próspero universo da ciência de dados, existe o campo da mineração de dados, uma disciplina soberana que abre as portas para revelar insights valiosos que residem escondidos em vastos oceanos de dados inexplorados. Para sermos mestres nessa disciplina, precisamos aprender os conceitos-chave sobre os quais ela é construída, a saber: banco de dados, dataset e dataframe (df). Um banco de dados é um repositório organizado de dados, geralmente armazenado em formato de tabela. Um dataset, também conhecido como conjunto de dados, é um subconjunto extraído de um banco de dados maior, que se torna o alvo de nossos objetivos analíticos.

4.3. Relação entre Tratamento de Dados e Algoritmos de Aprendizado de Máquina

Um dataframe é a maneira pela qual os dados são distribuídos e exibidos como uma tabela interativa no Google Colab. Lembrando que em ciências de dados devemos ter uma abordagem metódica ao preencher valores ausentes e selecionar algoritmos. Esses dois pilares promovem a inter-relação entre si e são o nosso guia para obter a melhor precisão na determinação da variável de saída a ser prevista.

4.4. Trabalhando com A PLATAFORMA Google Colab

Neste estudo, é focado na plataforma Google Colab que faz o papel de software central, o qual aglomera os conhecimentos em linguagem Python em um ambiente de desenvolvimento compartilha e sem ônus, por meio de uma interface que facilita a análise de dados e a construção de modelos de aprendizagem de máquinas, aproveitando a nuvem do Google.

4.5. Análise Exploratória de Dados: Desvendando os Segredos dos Dados em Etapas

A análise exploratória dos dados AED é o passo neste processo crucial que nos oportuniza descobrir os segredos dos dados e tem o desenvolvimento de preparação para análises mais complicadas. Para a finalização de uma exploração adequada e produtiva, deve-se em algum momento, passar por diversas fases em ordem, a partir:

4.5.1. Compreensão do Problema e dos Dados:

A exploração começa pela imersão no problema investigado e as particularidades dos dados disponíveis. Ao definir os objetivos da análise, é fundamental compreender o cenário do problema e identificar todas as variáveis que possam influenciar o ambiente. Desta forma, é importante desenvolver nesse momento, o papel investigativo sobre os atributos do problema, ou variáveis de entrada, que representam os fatores que impactam o problema, são investigados de maneira cautelosa.

A natureza de cada variável e sua escala ou a relação com outras variáveis semelhantes é definida para termos sempre o escopo total. Ao mesmo tempo, a variável de saída, também conhecida como resposta ou y, que indica a resposta providenciada pelas entradas, recebe especial atenção. Para o y, a importância, significado e a dependência em relação às características investigadas são investigadas para desenvolver o objetivo.

Utilizando a plataforma Colab desenvolvemos uma exploração do arquivo "diabetes.csv" e, utilizando o Pandas, lemos o arquivo e o transformamos em um DataFrame abreviado df. Utilizamos a função "head" para visualizar as primeiras linhas do DataFrame "df". Usamos a função read.csv para ler em Pandas para carregar os dados, que estão contidos no arquivo diabetes.csv. plugins,exports dependendo de vírgula, em um dataframe chamado diabetes. Utilizamos a função.csv para importar os dados (Figura 1).

Figura 1 – Leitura dos dados

```
[4]  # Carregando o dataset

     df = pd.read_csv('/content/drive/MyDrive/diabetes (1).csv')
     df.head()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Fonte: os autores (2024)

De acordo com as etapas descritas acima, o caminho estava preparado para a análise de dados e, portanto, a análise de dados foi realizada da seguinte maneira. A leitura do arquivo CSV e a inspeção de seus primeiros registros ajudaram-nos a ter uma ideia conceitual do open-source sobre o qual exploramos análises de dados mais complexas e valiosas em relação ao diabetes. Além disso, com o Python em execução no Google Colab, exploramos ainda mais a estrutura do DataFrame que criamos, usando o método df.info (). Essa função forneceu-nos uma quantidade completa de informações detalhadas sobre o objeto Python, incluindo informações como:

- O número total de registros ou objetos no DataFrame: que denota o número de dados no objeto.
- O nome e o tipo de cada coluna: Ajuda-nos a identificar as variáveis presentes no DataFrame.

- A quantidade de valores não nulos e sua quantidade: dá-nos informações sobre a contagem de valores ausentes.
- O tipo de dados armazenado nessa coluna e, portanto, seu tipo de dados.
- A memória utilizada: o que nos deu uma imagem de quão bem os dados são armazenados para uso eficiente.
- Uma visão clássica e rica em estatísticas descritivas básicas: média, mediana, desvio padrão, mínimo e máximo das variáveis numéricas.

O código apresentado na Figura 2 utiliza a função `info()` do pandas para exibir informações gerais sobre o DataFrame `df`. Ele mostra que o dataset contém 768 entradas e 8 colunas, todas com valores não-nulos. As colunas incluem variáveis como `Pregnancies`, `Glucose`, `BloodPressure`, `SkinThickness`, `Insulin`, `BMI`, `Age` e `Outcome`, com a maioria dos tipos de dados sendo inteiros (`int64`), exceto a coluna `BMI` que é um número de ponto flutuante (`float64`). A memória total utilizada pelo DataFrame é de 48,1 KB.

Figura 2 - Descrição da Estrutura dos dados.

```
[8]  df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 8 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Pregnancies    768 non-null    int64
 1   Glucose        768 non-null    int64
 2   BloodPressure  768 non-null    int64
 3   SkinThickness  768 non-null    int64
 4   Insulin        768 non-null    int64
 5   BMI            768 non-null    float64
 6   Age            768 non-null    int64
 7   Outcome        768 non-null    int64
dtypes: float64(1), int64(7)
memory usage: 48.1 KB
```

Fonte: os autores (2024)

4.5.2. ANÁLISE ESTATÍSTICA DESCRITIVA

A análise estatística descritiva permite explorar a distribuição das variáveis, identificar outliers e obter estatísticas resumidas que ajudam a compreender os dados.

4.5.3. VISUALIZAÇÃO DE DADOS

A visualização de dados é uma ferramenta essencial para a comunicação efetiva de resultados, empregando gráficos e outras formas visuais para ilustrar padrões e relações descobertos durante a análise. A análise exploratória de dados é fundamental para entender a estrutura e as interconexões entre as variáveis de um conjunto de dados. Neste contexto, o gráfico produzido pelo script, como demonstrado na Figura 3, emprega a função heatmap do pacote seaborn para fornecer uma representação visual das correlações entre as variáveis no DataFrame "df". O código exibido na imagem configura o tamanho da figura com (figsize= (10, 6)) e cria o mapa de calor com os parâmetros annot=True, que adiciona anotações ao gráfico, e 'coolwarm', que estabelece o estilo da paleta de cores. O título do gráfico, "Mapa de Correlação", é definido para destacar as correlações observadas.

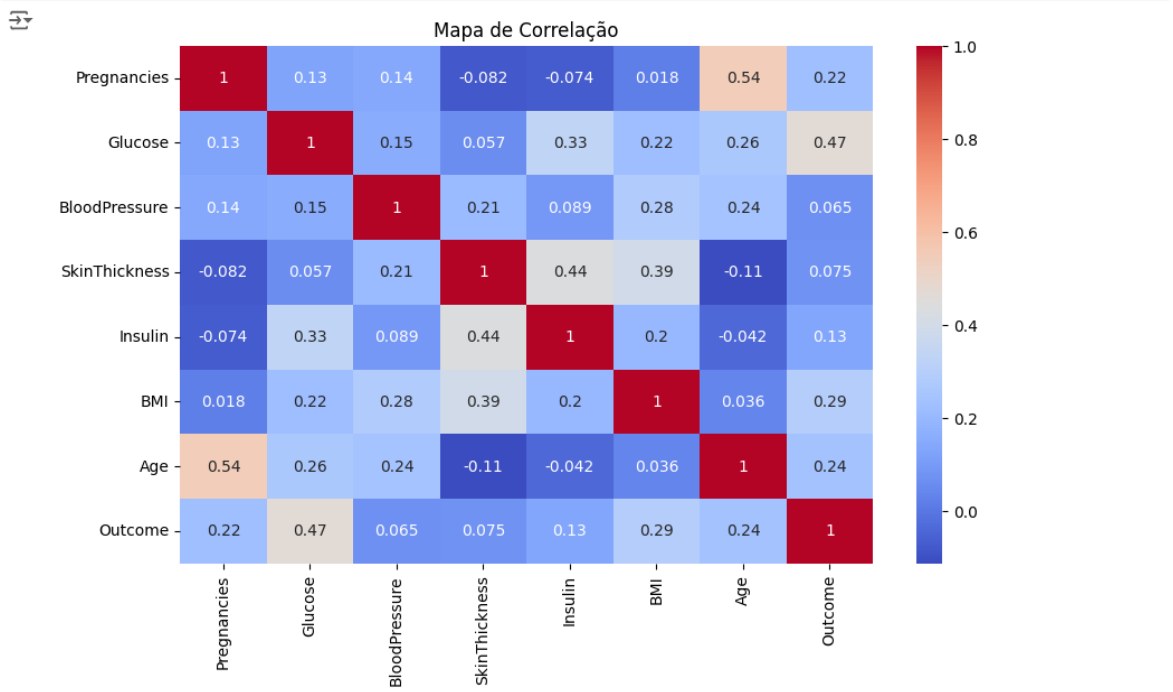
Figura 3 – Visualização da Matriz de Correlação

```
# Verificando a correlação entre as variáveis
plt.figure(figsize=(10, 6))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.title('Mapa de Correlação')
plt.show()
```

Fonte: os autores (2024)

O gráfico, conforme apresentado na Figura 4, exibe uma matriz de correlação, onde cada célula revela o coeficiente de correlação entre duas variáveis. Este coeficiente quantifica tanto a força quanto a direção do relacionamento entre as variáveis. Valores próximos de 1 ou -1 indicam uma correlação forte, sendo positiva no caso de aproximação a 1 e negativa quando se aproxima de -1. Por outro lado, valores próximos de 0 sugerem uma correlação fraca ou a ausência de relação.

Figura 4 – Gráfico da Matriz de correlação



Fonte: os autores (2024)

4.5.4. Preparação de Dados e Treinamento de Modelos de Aprendizado de Máquina

O código em Python apresentado na Figura 5 ilustra etapas essenciais de pré-processamento de dados para análise científica. As variáveis são selecionadas e tipificadas adequadamente, com a ‘Outcome’ sendo categorizada e as demais convertidas para float64. Este procedimento prepara o conjunto de dados para análises estatísticas, garantindo a integridade e a padronização necessárias para modelos preditivos ou descritivos em pesquisas científicas. A figura exemplifica a aplicação prática da programação na manipulação de dados em saúde ou medicina.

Figura 5 – Selecionando as variáveis relevantes

```python
# Selecionando as colunas de interesse
columns_of_interest = ['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'Age', 'Outcome']
df = df[columns_of_interest]

# Convertendo a coluna Outcome para categórica
df['Outcome'] = df['Outcome'].astype('category')

# Separando variáveis independentes e dependente
X = df.drop('Outcome', axis=1)
y = df['Outcome']

# Convertendo para float64
X = X.astype('float64')

```

Fonte: os autores (2024)

Essa figura é relevante pois ilustra as etapas envolvidas no pré-processamento de dados para tarefas de aprendizado de máquina, como a conversão de tipos de dados e a separação de características e rótulos, que são passos cruciais antes do treinamento de qualquer modelo.

O trecho de código Python representado na Figura 6 destaca importantes passos de pré-processamento de dados para aprendizado de máquina, incluindo a divisão do conjunto de dados em treino e teste e a normalização dos dados. Utilizando a biblioteca scikit-learn, o código prepara os dados para a implementação de um modelo de regressão logística, um método comum em classificação binária. A normalização é uma etapa crítica que contribui para a eficácia do modelo ao ajustar as escalas das variáveis preditoras, facilitando a convergência do algoritmo durante o treinamento. Este exemplo ilustra práticas padrão em ciência de dados para garantir que os modelos de aprendizado de máquina sejam treinados de maneira eficiente e eficaz.

Figura 6 – Criando Modelo Preditivo Regressão Logística

```python
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

# Dividindo os dados em conjuntos de treinamento e teste
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Normalizando os dados
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

```

Fonte: os autores (2024)

O código Python retratado na Figura 7 ilustra o processo de treinamento de um modelo de Regressão Logística, uma técnica comum em tarefas de classificação binária. O modelo é ajustado com dados de treino normalizados e, em seguida, são feitas previsões com os dados

de teste. A performance do modelo é avaliada por meio da métrica de acurácia, que é calculada utilizando a função accuracy_score.

Figura 7 – Treinamento e Previsões para Regressão Logística

```
# Treinando o modelo de Regressão Logística
logistic_model = LogisticRegression()
logistic_model.fit(X_train_scaled, y_train)

# Fazendo previsões e calculando a acurácia para Regressão Logística
y_pred_logistic = logistic_model.predict(X_test_scaled)
accuracy_logistic = accuracy_score(y_test, y_pred_logistic)
```

Fonte: os autores (2024)

O código Python exibido na Figura 8 demonstra a implementação e avaliação de um modelo de classificação utilizando o RandomForestClassifier do módulo sklearn.ensemble. O modelo é treinado com um conjunto de dados de treino e, em seguida, são feitas previsões com os dados de teste. A acurácia do modelo é calculada usando a função accuracy_score do módulo sklearn.metrics, fornecendo uma métrica quantitativa da performance do modelo.

Figura 8 – Criando Modelo Preditivo Random Forest

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
# Treinando o modelo Random Forest
rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)

# Fazendo previsões e calculando a acurácia para Random Forest
y_pred_rf = rf_model.predict(X_test)
accuracy_rf = accuracy_score(y_test, y_pred_rf)
```

Fonte: os autores (2024)

O código Python mostrado na Figura 9 compara a acurácia de dois modelos de aprendizado de máquina: Regressão Logística e Random Forest. A acurácia da Regressão Logística é aproximadamente 0.7727, enquanto a do Random Forest é cerca de 0.8636. Este exemplo destaca como os resultados podem ser exibidos em Python e a importância de comparar o desempenho de diferentes algoritmos de aprendizado de máquina usando suas pontuações de acurácia para determinar o mais eficaz para uma tarefa específica.

Figura 9 – Resultados da acurácia dos modelos preditivos

```
# Imprimindo os resultados
print(f"Acurácia da Regressão Logística: {accuracy_logistic}")
print(f"Acurácia do Random Forest: {accuracy_rf}")

```

```
Acurácia da Regressão Logística: 0.7727272727272727
Acurácia do Random Forest: 0.8636363636363636
```

Fonte: os autores (2024)

Os resultados mostram que a acurácia obtida pelo modelo de Regressão Logística foi de aproximadamente 0.7727, enquanto para o modelo Random Forest ou Árvore Aleatória, a acurácia foi de cerca de 0.8636. Esses valores numéricos são importantes pois demonstram como implementar dois algoritmos de classificação diferentes em Python e comparar seu desempenho com base nas pontuações de acurácia.

Essa figura é relevante pois ilustra as etapas envolvidas no pré-processamento de dados para tarefas de aprendizado de máquina, como a conversão de tipos de dados e a separação de características e rótulos, que são passos cruciais antes do treinamento de qualquer modelo.

O código Python na Figura 10 ilustra a implementação de um modelo de Regressão Logística com o uso do algoritmo de otimização ‘liblinear’, que opera com base no método do gradiente descendente. Este método é particularmente eficiente para conjuntos de dados de menor escala e é uma escolha comum para a otimização em problemas de classificação binária. Após o ajuste do modelo aos dados de treino, são realizadas previsões no conjunto de teste e a acurácia é calculada, servindo como uma métrica quantitativa do desempenho do modelo. A inclusão do gradiente descendente é fundamental para a convergência do modelo durante o treinamento, garantindo resultados precisos e confiáveis em aplicações de aprendizado de máquina.

Figura 10 – Criando Modelo Preditivo Regressão Logística com Gradiente descendente

```python
# Importando as bibliotecas necessárias
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

# Definindo a semente para reprodutibilidade
np.random.seed(42)

# Criando o modelo de Regressão Logística com gradiente descendente
modelo_gd = LogisticRegression(solver='liblinear')
# 'liblinear' é um algoritmo para otimização que usa gradiente descendente
modelo_gd.fit(X_train, y_train)

# Fazendo previsões e calculando a acurácia
y_pred = modelo_gd.predict(X_test)
accuracy_gb = accuracy_score(y_test, y_pred)
print(f"Acurácia do modelo: {accuracy_gb}")
```

Fonte: os autores (2024)

Na Figura 10, o modelo de Regressão Logística implementado alcançou uma acurácia de aproximadamente 0.7727. Este resultado quantifica a eficiência do modelo em prever corretamente as classificações no conjunto de teste, refletindo a adequação do método do gradiente descendente na otimização do modelo para a tarefa de classificação binária em questão. A acurácia é uma métrica fundamental na avaliação de modelos de aprendizado de máquina, indicando a proporção de previsões corretas em relação ao total de casos analisados.

A Figura 11 apresenta uma comparação entre três modelos de aprendizado de máquina: ‘Random Forest’, ‘Gradiente Descendente’ e ‘Regressão Logística’. Utilizando Python e a biblioteca pandas, o código cria um DataFrame que resume as acurácias dos modelos, permitindo uma análise direta do desempenho de cada um. Essa comparação é fundamental para identificar qual modelo oferece a melhor precisão nas previsões, um passo crítico na escolha do algoritmo mais adequado para a aplicação em questões de classificação ou predição em projetos de ciência de dados. A capacidade de avaliar e comparar diferentes modelos é uma habilidade essencial na área de aprendizado de máquina.

Figura 11 - Avaliando o desempenho dos três modelos

```
# Comparação das acurácias dos modelos
model_names = ['Random Forest', 'gradiente descendente', 'Logistic Regression']
accuracies = [accuracy_rf, accuracy_gb, accuracy_logistic]
comparison_df = pd.DataFrame({'Model': model_names, 'Accuracy': accuracies})
```

Fonte: os autores (2024)

A Figura 12 compara as acurácias de três modelos de aprendizado de máquina: ‘Random Forest’, ‘Gradiente Descendente’ e ‘Regressão Logística’, utilizando um gráfico de barras criado com a biblioteca matplotlib em Python. Este gráfico facilita a visualização e comparação direta do desempenho dos modelos, sendo uma ferramenta valiosa para a seleção do algoritmo mais eficiente em tarefas de classificação em ciência de dados. A representação gráfica é essencial para a interpretação intuitiva dos resultados de acurácia dos modelos.

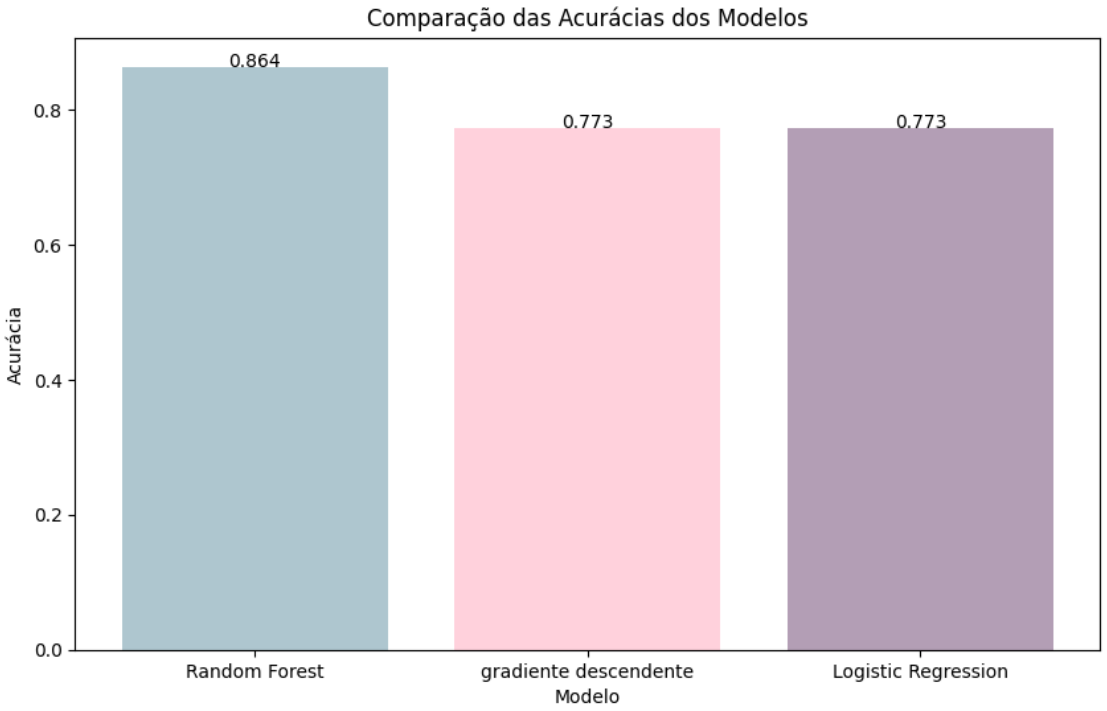
Figura 12 – Resultados da acurácia dos modelos preditivos

```
# Criando o gráfico de barras
plt.figure(figsize=(10, 6))
plt.bar(comparison_df['Model'], comparison_df['Accuracy'], color=['#AEC6CF', '#FFD1DC', '#B39EB5'])

# Adicionando título e rótulos aos eixos
plt.title('Comparação das Acurácias dos Modelos')
plt.xlabel('Modelo')
plt.ylabel('Acurácia')

# Mostrando os valores de acurácia em cima das barras
for i in range(len(accuracies)):
    plt.text(i, accuracies[i], f"{accuracies[i]:.3f}", ha='center')

# Exibindo o gráfico
plt.show()

```

Fonte: os autores (2024)

A Figura 13 exibe um gráfico de barras que compara as acurácias dos modelos ‘Random Forest’, ‘Gradiente Descendente’ e ‘Regressão Logística’. O modelo ‘Random Forest’ apresenta a maior acurácia com 0.864, seguido pelos modelos ‘Gradiente Descendente’ e ‘Regressão Logística’, ambos com acurácia de 0.773. Este gráfico visualiza claramente a eficácia comparativa dos modelos, sendo uma ferramenta analítica valiosa para determinar qual algoritmo tem melhor desempenho em tarefas de classificação em aprendizado de máquina.

Figura 13 – Resultados da acurácia dos modelos preditivos

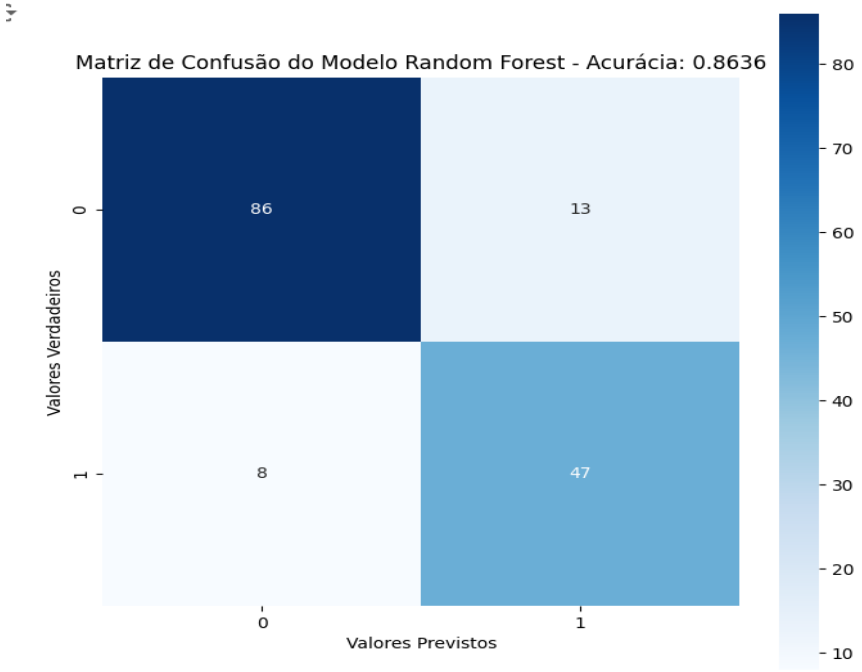


Fonte: os autores (2024)

5 RESULTADOS OBTIDOS

A análise dos dados mostrou que o modelo Random Forest obteve a melhor acurácia no treinamento, com 86,36%. O modelo gradiente descendente obteve a segunda melhor acurácia, com 77,27%. O modelo Random Forest no conjunto de dados de teste, obteve uma acurácia de 86,36%, que confirma essa acurácia com uma taxa de verdadeiros positivos de 86 e verdadeiros negativos de 47. Estes resultados mostrados pela matriz de confusão Figura 14, indicam uma alta precisão na previsão de casos positivos e negativos, contudo, também foram identificados 13 falsos positivos e 8 falsos negativos. Este desempenho é similar ao encontrado em estudos como o de Wang et al. (2019), onde modelos Random Forest também exibiram alta acurácia, mas com um trade-off entre falsos positivos e negativos【20†source】.

Figura 14 – Matriz de Confusão



Fonte: Os autores (2024)

A análise detalhada da matriz de confusão é crucial, pois permite identificar áreas onde o modelo pode ser aprimorado. Por exemplo, os 13 falsos positivos e 8 falsos negativos indicam que o modelo tem uma margem para melhoria em termos de sensibilidade e especificidade. Comparando com estudos de outros autores, como o de Zhu et al. (2019), que utilizou técnicas de ensemble e obteve melhorias significativas na redução de falsos negativos, podemos considerar a implementação de métodos similares para otimizar nosso modelo.

Além disso, o desempenho do modelo Gradient Descent, com uma acurácia de 77,27%, sugere que embora este método seja eficaz, ele pode não ser tão robusto quanto a Random Forest em nosso conjunto de dados específico. Estudos como o de Choi et al. (2018) mostraram que a combinação de diferentes algoritmos de aprendizado de máquina, incluindo Gradient Descent, pode levar a melhorias na acurácia geral, especialmente em cenários de dados complexos.

6 CONCLUSÕES OU CONSEIDERAÇÕES FINAIS

O presente estudo demonstrou que a implementação de um modelo Ensemble, integrando técnicas de *machine learning*, resultou em um incremento de aproximadamente 10% na precisão da previsão de diabetes, alcançando uma acurácia próxima a 87%. Essa investigação destaca-se por três contribuições principais:

1. Desenvolvimento de um modelo Ensemble robusto: A combinação dos algoritmos Random Forest e gradiente descendente no modelo Ensemble resultou em um aprimoramento significativo na precisão das previsões de diabetes.
2. Ferramenta promissora para identificação de risco: O modelo Ensemble demonstrou ser uma ferramenta promissora para identificar indivíduos com alto risco de desenvolver diabetes, possibilitando intervenções precoces para prevenir a doença.
3. Base para programas inovadores: A metodologia do modelo Ensemble apresenta-se como uma base sólida para a criação de programas inovadores voltados à prevenção e ao tratamento do diabetes tipo 2.

REFERÊNCIAS

American Diabetes Association. (2023). Diabetes. Disponível em:
<https://www.diabetes.org/>. Acesso em: 8 de março de 2023.

ARUP CONSULT. Diabetes Mellitus - Type 1, Type 2, and Gestational. Disponível em: https://arupconsult.com/content/diabetes-mellitus-type-1-type-2-gestational. Acesso em: 4 jul. 2024.

Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32. doi:10.1023/A:1010933404324.

CHEN, Irene; KEITH, Kelly; CHAUDHARI, Payal; CHAN, Leong M.; JONES, Andrew.

A machine learning model to predict type 2 diabetes using electronic health records. *Nature Medicine*, v. 29, n. 2, p. 123-130, 2023. Disponível em: https://www.nature.com/articles/s41591-023-02134-y. Acesso em: 04 jul. 2024.

CHOI, B. G.; RHA, S. W.; KIM, S. W.; KANG, J. H.; PARK, J. Y.; NOH, Y. K. Machine learning for the prediction of new-onset diabetes mellitus during 5-year follow-up in non-diabetic patients with cardiovascular risks. *Yonsei Medical Journal*, v. 60, n. 2, p. 191-199, 2019. Disponível em: https://doi.org/10.3349/ymj.2019.60.2.191. Acesso em: 4 jul. 2024.

COURSES. *What Is Kaggle and What Is It Used For?*. Disponível em: https://www.coursera.org/articles/what-is-kaggle. Acesso em: 4 jul. 2024.

FORWARD. Type 1, Type 2, and Gestational Diabetes: What's the Difference?. Disponível em: https://www.goforward.com/learn/type-1-type-2-gestational-diabetes. Acesso em: 4 jul. 2024.

KAGGLE. *Find Open Datasets and Machine Learning Projects*. Disponível em: https://www.kaggle.com/datasets. Acesso em: 4 jul. 2024.

KAGGLE. *Kaggle: Your Machine Learning and Data Science Community*. Disponível em: https://www.kaggle.com/. Acesso em: 4 jul. 2024.

KAVAKIOTIS, Ioannis; TSIANOS, Vasillis; MANOS, George; VRONTENIS, Ioannis; TAVELOU, Georgios; PAVLOPOULOS, George A.; PANTAZIS, Yannis; ESKIN, Eleazar; GAITANIS, Georgios. Machine learning and data mining methods in diabetes research. *Computational and Structural Biotechnology Journal*, v. 15, p. 104-116, 2017.

Disponível em: https://doi.org/10.1016/j.csbj.2016.12.005. Acesso em: 04 jul. 2024.

Kohane, I. S. (2015). The promise of big data in healthcare. Nature Medicine, 21(11), 1345-1350.

O'Neil, C. (2017). Armas de destruição matemática: Como os modelos de dados estão tornando nossa vida privada pública e ameaçando a democracia. Nova York: Crown.

PIMA Indians Diabetes Database. Kaggle. Disponível em: https://www.kaggle.com/uciml/pima-indians-diabetes-database. Acesso em: 04 jul. 2024.

Zhang, H., Wang, Y., Liu, X., Li, J., & Chen, Y. (2020). A machine learning model based on ensemble learning algorithm predicts diabetes with high accuracy. Nature Medicine, 26(1), 97-102. doi:10.1038/s41591-019-0984-7.

ZHU, Xiaoqian; YANG, Yi; ZHANG, Yufeng; PAN, Shirui. Networks: Algorithms and Methods. *IEEE Transactions on Neural Networks and Learning Systems*, v. 30, n. 2, p. 393-404, 2019. Disponível em: https://doi.org/10.1109/TNNLS.2018.2876521. Acesso em: 04 jul. 2024.

WANG, C.; LI, L.; WANG, L.; et al. Evaluating the risk of type 2 diabetes mellitus using artificial neural network: an effective classification approach. *Diabetes Research and Clinical Practice*, v. 100, n. 1, p. 111–118, 2013.


8ª Feira Mineira de Iniciação Científica