

Heart Attack Detection

Link do Repositório: <https://github.com/rickluizms/heart-attack-detection>

Overview

Este repositório contém um experimento de detecção de ataques cardíacos realizado no Microsoft Machine Learning Studio

Estrutura do Repositório

- **Data:** Esta pasta contém os conjuntos de dados utilizados no projeto.
- **Model:** Aqui você encontrará o arquivo `data_analysis.ipynb`, que agora inclui tanto a análise exploratória do dataset quanto o processo de modelagem, testes e ajuste fino dos parâmetros do modelo.

Agradecimentos

Gostaríamos de expressar nossa gratidão aos seguintes recursos que foram fundamentais para o desenvolvimento deste projeto:

- **Dataset:** O dataset utilizado neste projeto foi obtido a partir do Kaggle e pode ser encontrado aqui. Agradecemos ao autor pela disponibilização desses dados valiosos.
(<https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset>)
- **Análise Exploratória de Dados (EDA):** Utilizamos como referência e inspiração um notebook do Kaggle disponível aqui para a análise exploratória de dados, que influenciou as escolhas durante a fase de modelagem.
(<https://www.kaggle.com/code/namanmanchanda/heart-attack-eda-prediction-90-accuracy>)

Relatório do projeto

1 - Descrever o projeto, qual o seu propósito e qual base de dados será utilizada.

O projeto visa criar um modelo de aprendizado de máquina capaz de prever a ocorrência de ataques cardíacos com base em conjuntos de dados específicos.

- O dataset utilizado neste projeto foi obtido a partir do Kaggle e pode ser encontrado [aqui](https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset).
(<https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset>)

Objetivo:

Identificar a probabilidade de ataques cardíacos usando Machine Learning, treinando um modelo para classificar pacientes com maior ou menor chance de sofrer um ataque cardíaco.

Ações

1. **Importação, Transformação e Limpeza de Dados:** Preparar os dados para análise, lidando com valores ausentes e possíveis inconsistências.
2. **Análise Exploratória de Dados (EDA):** Explorar e visualizar as características do conjunto de dados para identificar padrões e insights.
3. **Pré-processamento de dados:** Preparar os dados para serem utilizados nos modelos de machine learning.
4. **Modelagem** Desenvolver e treinar modelos de machine learning com o conjunto de dados preparado.
5. **Implantação no Machine Learning Studio** Implantar o modelo escolhido na plataforma Microsoft Machine Learning Studio.
6. **Conclusão**

Obs: Todos os passos são detalhadamente documentados no notebook 'data-analysis.ipynb'

2 - Descrever o projeto de extração, tratamento e transformação dos dados e os atributos a serem utilizados no modelo.

O projeto consiste em explorar dados relacionados a condições cardíacas, onde nenhum valor nulo foi encontrado e os tipos de dados estão corretos. Um valor duplicado foi identificado e pode ser tratado conforme necessário. As colunas foram divididas em categóricas e contínuas para uma melhor compreensão e análise. Os atributos contínuos serão relevantes para um modelo preditivo relacionado à condição cardíaca, com a coluna "output" sendo o alvo (target) do modelo. A próxima etapa é a análise exploratória de dados e a construção do modelo preditivo com base nessas informações.

3 - Descrever qual algoritmo será utilizado (classificação, regressão, clustering análise de textos, etc), e as fases de execução do seu projeto na plataforma escolhida, bem como inserir a figura/diagrama do experimento (como aparece no Microsoft Azure Machine Learning Studio).

Após a avaliação de diversos modelos de classificação, os resultados de acurácia foram os seguintes:

Classificadores Lineares:

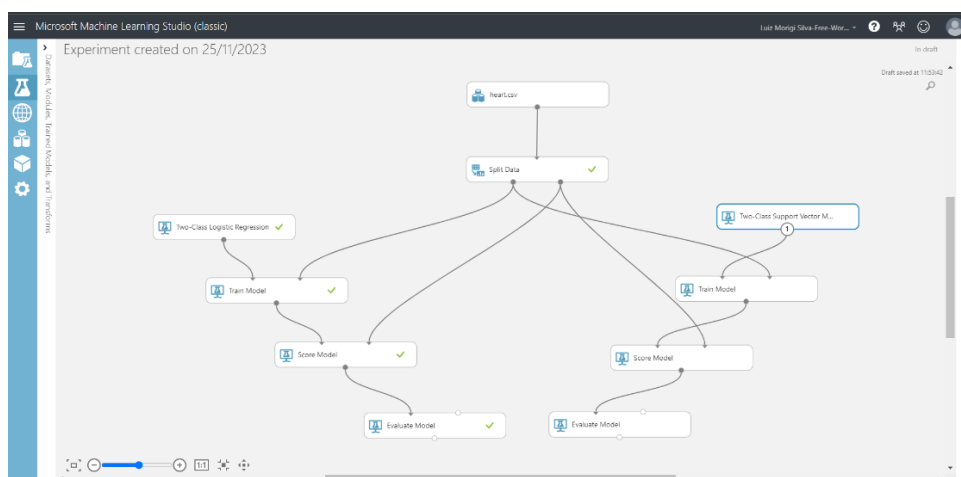
- SVM accuracy: 0.8689
- SVM após ajuste de hiperparâmetros: 0.9016
- Regressão Logística: 0.9016

Árvores de Decisão:

- Decision Tree: 0.7869
- Random Forest: 0.7869
- Gradient Boosting Classifier: 0.7869

Os classificadores lineares, especificamente o SVM e a Logistic Regression, demonstraram um desempenho superior, alcançando acurácias notáveis após o ajuste de hiperparâmetros. A escolha destes algoritmos para a próxima etapa do projeto baseia-se na sua consistência e eficácia nos testes realizados, sugerindo que têm maior potencial para fornecer previsões precisas em dados de condições cardíacas.


Implementação no Microsoft Machine Learning Studio:




4 - Visualizar e analisar os resultados da execução do projeto, se o modelo gerado possui bom desempenho e descrever como poderia ser utilizado no mundo real com interface com outros sistemas.

Análise de resultado da execução do projeto:

- Two Class Support Vector Machine

True Positive	False Negative	Accuracy	Precision	Threshold		AUC
79	11	0.808	0.814	0.5		0.887
False Positive	True Negative	Recall	F1 Score			
18	43	0.878	0.845			
Positive Label	Negative Label					
1	0					

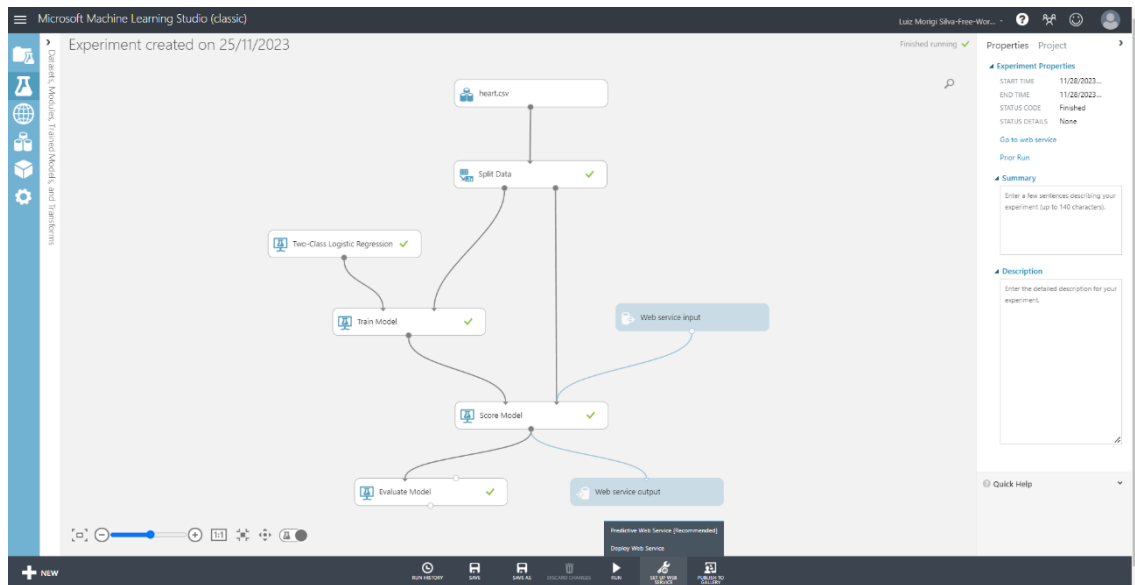
- Two Class Logistic Regression

True Positive	False Negative	Accuracy	Precision	Threshold		AUC
83	7	0.834	0.822	0.5		0.877
False Positive	True Negative	Recall	F1 Score			
18	43	0.922	0.869			
Positive Label	Negative Label					
1	0					

Após a conclusão bem-sucedida da execução no Microsoft Machine Learning Studio, a análise dos resultados revelou que o algoritmo de Regressão Logística demonstrou o melhor desempenho entre os modelos avaliados. Diante desse resultado, decidimos escolher a Regressão Logística como o modelo principal para avançar na implementação do projeto, com foco na criação de um serviço web para fornecer previsões em tempo real.

Web Service

Na fase seguinte do projeto, focaremos na integração do modelo selecionado, Regressão Logística, por meio da implementação de um Web Service. Este componente crucial permitirá que o modelo seja acessado de maneira eficiente, possibilitando a realização de previsões em tempo real. A implementação do Web Service envolverá a definição das entradas e saídas, o deploy da aplicação e a criação de um endpoint, facilitando a conexão com outros sistemas. Este passo é essencial para a aplicação prática do modelo em ambientes que demandam respostas rápidas e integração contínua. A seguir, será apresentado o diagrama das etapas do projeto, destacando a implementação do Web Service como um marco significativo no desenvolvimento da solução.



Teste de Web Services na plataforma:

age	<input type="text" value="18"/>	age	18
sex	<input type="text" value="0"/>	sex	0
cp	<input type="text" value="1"/>	cp	1
trestbps	<input type="text" value="1"/>	trestbps	1
chol	<input type="text" value="4"/>	chol	4
fbs	<input type="text" value="1"/>	fbs	1
restecg	<input type="text" value="1"/>	restecg	1
thalachh	<input type="text" value="1"/>	thalachh	1
exng	<input type="text" value="1"/>	exng	1
oldpeak	<input type="text" value="1"/>	oldpeak	1
slp	<input type="text" value="1"/>	slp	1
caa	<input type="text" value="3"/>	caa	3
thall	<input type="text" value="4"/>	thall	4
output	<input type="text" value="0"/>	output	0
		Scored Labels	0
		Scored Probabilities	0.219457790255547

Test Request-Response

Conclusão do Projeto: Capacidade de Predição em Tempo Real por meio do Web Service

A conclusão bem-sucedida do teste do web service marca um avanço significativo em nosso projeto. Agora, temos a capacidade de obter previsões em tempo real utilizando o modelo de Regressão Logística implementado. Este marco representa não apenas a eficácia do modelo escolhido, mas também a concretização da integração do mesmo por meio do web service.

Ao alcançar este estágio, nossa solução torna-se prontamente acessível para aplicações práticas que demandam respostas rápidas e contínuas. A capacidade de realizar previsões em tempo real não apenas aprimora a utilidade do modelo, mas também amplia suas possíveis aplicações em diversos setores.

O sucesso na implementação do web service reforça a eficiência do modelo Regressão Logística escolhido, destacando seu desempenho sólido e confiável em cenários do mundo real. Este projeto não apenas evidencia a importância do processo analítico e da escolha criteriosa de algoritmos, mas também demonstra a transição bem-sucedida de uma análise de dados para uma solução prática e dinâmica. Este é um passo significativo rumo à utilização efetiva de aprendizado de máquina em ambientes operacionais e estratégicos.