Documentação Técnica

Link git hub - https://github.com/amandazevedo01/Pos_FIAP/tree/main

Projeto: Classificação de Espécies de Flores com Spark e scikit-learn
Curso: Machine Learning Engineering

Disciplina: Fase 3 - Arquitetura ML e Aprendizado
Aluno(a): Amanda

Data: Outubro de 2025

1. Objetivo

Este projeto tem como finalidade desenvolver um modelo de classificação supervisionada capaz de prever a espécie de uma flor com base em atributos morfológicos. A solução utiliza dados armazenados em formato Parquet no Amazon S3, processados inicialmente com Apache Spark e posteriormente modelados com scikit-learn.

2. Tecnologias Utilizadas

- **Apache Spark**: Leitura e manipulação de dados em ambiente distribuído
- **Amazon S3**: Armazenamento de dados em nuvem
- **Python (Pandas, scikit-learn)**: Modelagem estatística e avaliação
- **Jupyter Notebook / Ambiente compatível**: Execução dos scripts

3. Pipeline de Desenvolvimento

3.1 Leitura dos dados com Spark

```
df = spark.read.parquet("s3://fiapflores/refined/flores.parquet")

df.createOrReplaceTempView("flores")

spark.sql("SELECT * FROM flores").show()
```

3.2 Conversão para Pandas

```
df_pd = df.toPandas()
```

A conversão para Pandas foi necessária para aplicar algoritmos de machine learning com scikit-learn, que não são suportados nativamente no ambiente Spark utilizado.

3.3 Modelagem com scikit-learn

```
from sklearn.ensemble import RandomForestClassifier

from sklearn.model_selection import train_test_split

from sklearn.metrics import classification_report

X = df_pd[["comprimento_sepala", "largura_sepala", "comprimento_petala", "largura_petala"]]

y = df_pd["classe"]
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)

model = RandomForestClassifier()

model.fit(X_train, y_train)


preds = model.predict(X_test)

print(classification_report(y_test, preds))
```

Foi utilizado o algoritmo Random Forest para realizar a classificação. A divisão dos dados em treino e teste seguiu a proporção de 70/30. A avaliação foi feita com métricas de desempenho como acurácia, precisão, recall e F1-score.

4. Estrutura dos Dados

Coluna	Tipo	Descrição
comprimento_sepala	float	Comprimento da sépala
largura_sepala	float	Largura da sépala
comprimento_petala	float	Comprimento da pétala
largura_petala	float	Largura da pétala
classe	string	Espécie da flor (variável alvo)

5. Resultados

O modelo apresentou desempenho satisfatório na tarefa de classificação, com métricas consistentes entre as classes. A Random Forest demonstrou robustez mesmo sem ajustes de hiperparâmetros.

6. Conclusão

O projeto cumpriu seu objetivo ao integrar tecnologias de processamento distribuído (Spark) com modelagem estatística (scikit-learn), utilizando dados reais em ambiente de nuvem. A abordagem adotada é compatível com práticas de mercado e atende aos critérios acadêmicos exigidos pela FIAP.

Progress: 100% elapsed time = 00:08s, DPU counts active/requested = 0/0

Calculation completed.

comprimento_sepala	largura_sepala	comprimento_petala	largura_petala	classe
5.1	3.5	1.4	0.2	Iris-setosa
4.9	3.0	1.4	0.2	Iris-setosa
4.7	3.2	1.3	0.2	Iris-setosa
4.6	3.1	1.5	0.2	Iris-setosa
5.0	3.6	1.4	0.2	Iris-setosa
5.4	3.9	1.7	0.4	Iris-setosa
4.6	3.4	1.4	0.3	Iris-setosa
5.0	3.4	1.5	0.2	Iris-setosa
4.4	2.9	1.4	0.2	Iris-setosa
4.9	3.1	1.5	0.1	Iris-setosa
5.4	3.7	1.5	0.2	Iris-setosa
4.8	3.4	1.6	0.2	Iris-setosa
4.8	3.0	1.4	0.1	Iris-setosa
4.3	3.0	1.1	0.1	Iris-setosa
5.8	4.0	1.2	0.2	Iris-setosa
5.7	4.4	1.5	0.4	Iris-setosa
5.4	3.9	1.3	0.4	Iris-setosa
5.1	3.5	1.4	0.3	Iris-setosa
5.7	3.8	1.7	0.3	Iris-setosa
5.1	3.8	1.5	0.3	Iris-setosa

only showing top 20 rows

	precision	recall	f1-score	support
Iris-setosa	1.00	1.00	1.00	13
Iris-versicolor	0.94	0.94	0.94	17
Iris-virginica	0.93	0.93	0.93	15
accuracy			0.96	45
macro avg	0.96	0.96	0.96	45
weighted avg	0.96	0.96	0.96	45