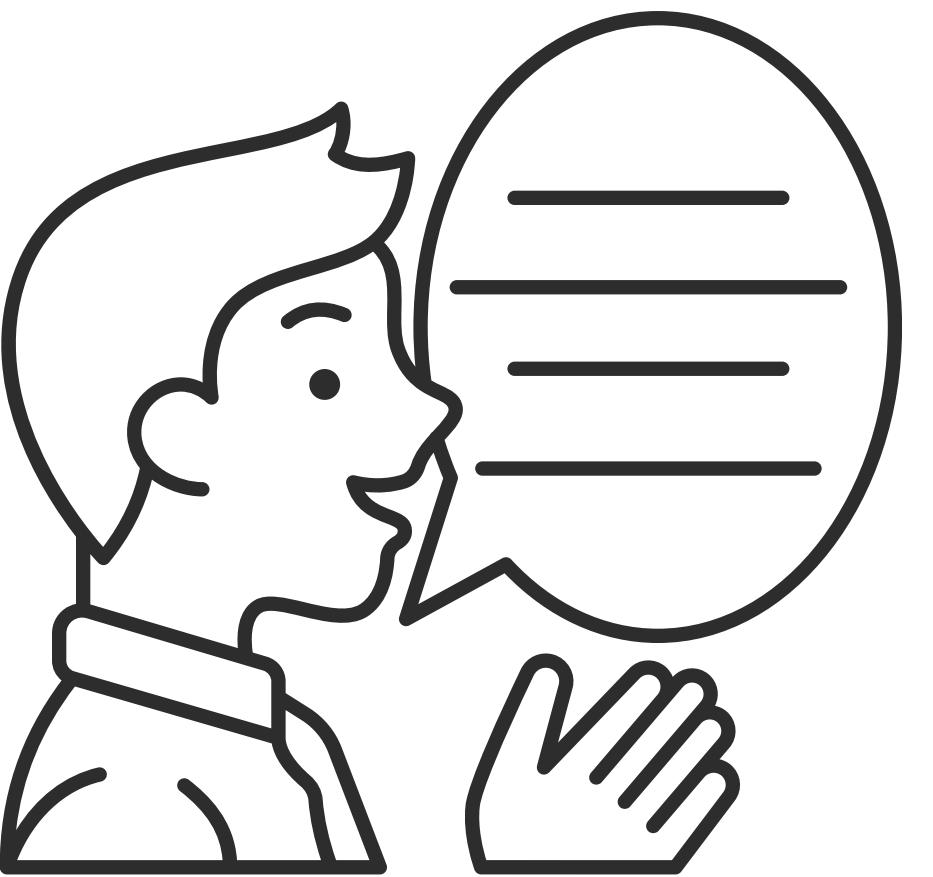


Análise da Doença de Parkinson: Sistema de Suporte ao Diagnóstico

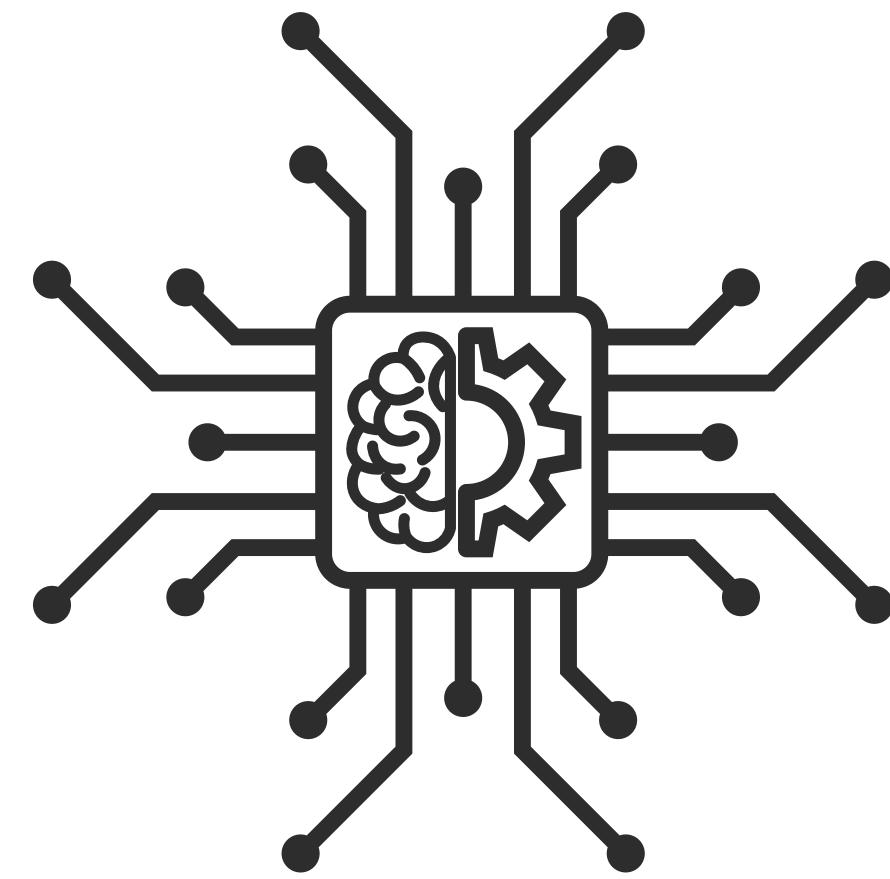
Tech Challenge IADT - Fase 1



Luana dos Santos Caldas
Elaine Martins da Silva



Será que a voz pode nos ajudar a detectar o Parkinson antes dos sintomas motores aparecerem?



Base de Dados e Variáveis

Dicionário de Variáveis

- name: Nome e número da gravação de voz do indivíduo.
- status: Variável-alvo (rótulo de classe), indicando o estado de saúde do sujeito:
 - 0: Indivíduo saudável
 - 1: Indivíduo com a doença de Parkinson
- MDVP:Fo(Hz): Frequência vocal fundamental média.
- MDVP:Fhi(Hz): Frequência vocal fundamental máxima.
- MDVP:Flo(Hz): Frequência vocal fundamental mínima.
- **Jitter (%)** / MDVP:Jitter(Abs), MDVP:RAP, MDVP:PPQ, Jitter:DDP: Várias medidas de variação na frequência fundamental da voz.
- **Shimmer (%)** / MDVP:Shimmer(dB), Shimmer:APQ3, Shimmer:APQ5, MDVP:APQ, Shimmer:DDA: Várias medidas de variação na amplitude da voz.
- NHR, HNR: Medidas da razão entre o ruído e os componentes tonais na voz.
- RPDE, D2: Duas medidas de complexidade dinâmica não-linear.
- DFA: Exponente de escala fractal do sinal.
- **spread1, spread2, PPE**: Três medidas não-lineares de variação da frequência fundamental.

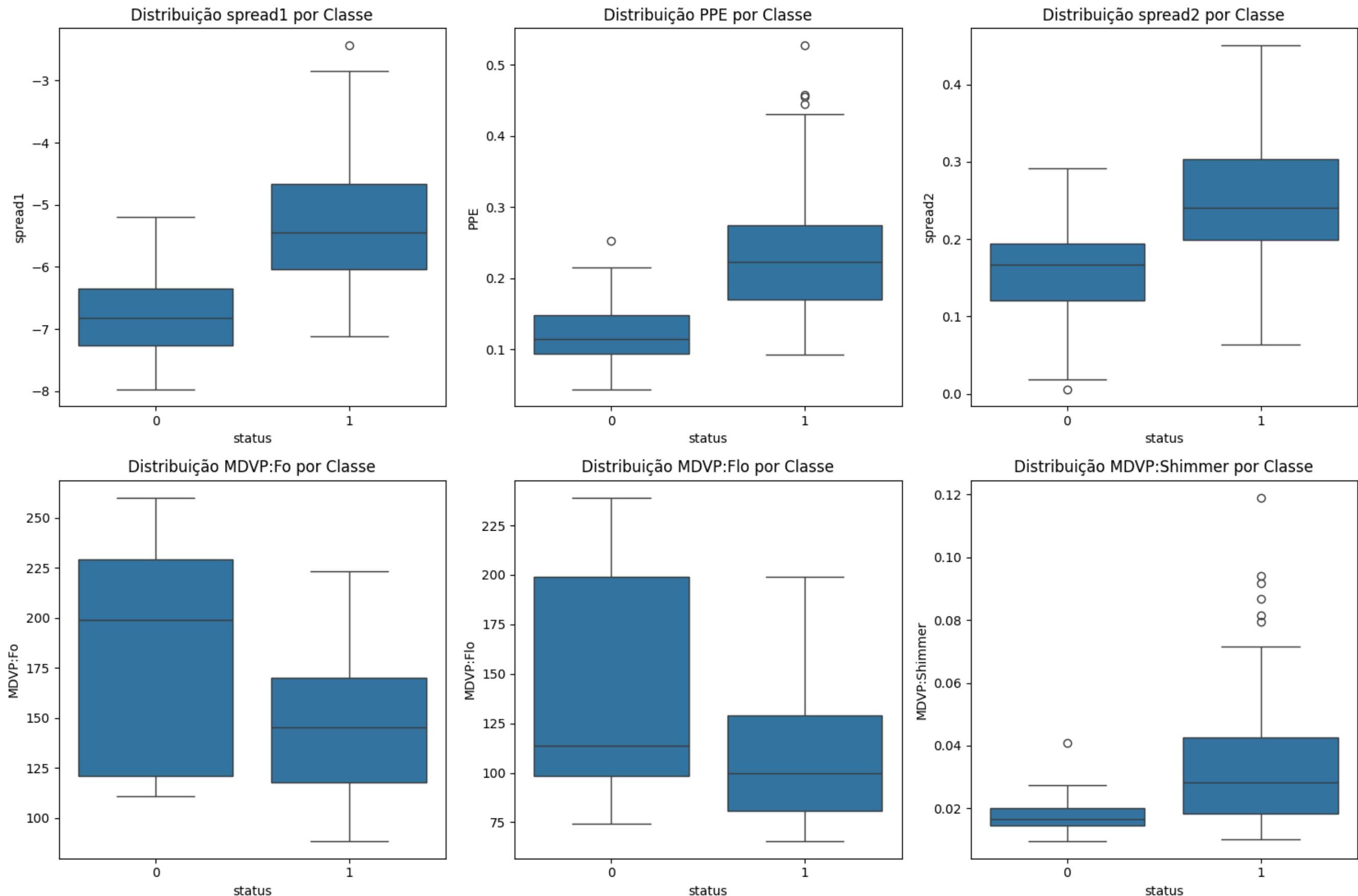
195 amostras

31 indivíduos

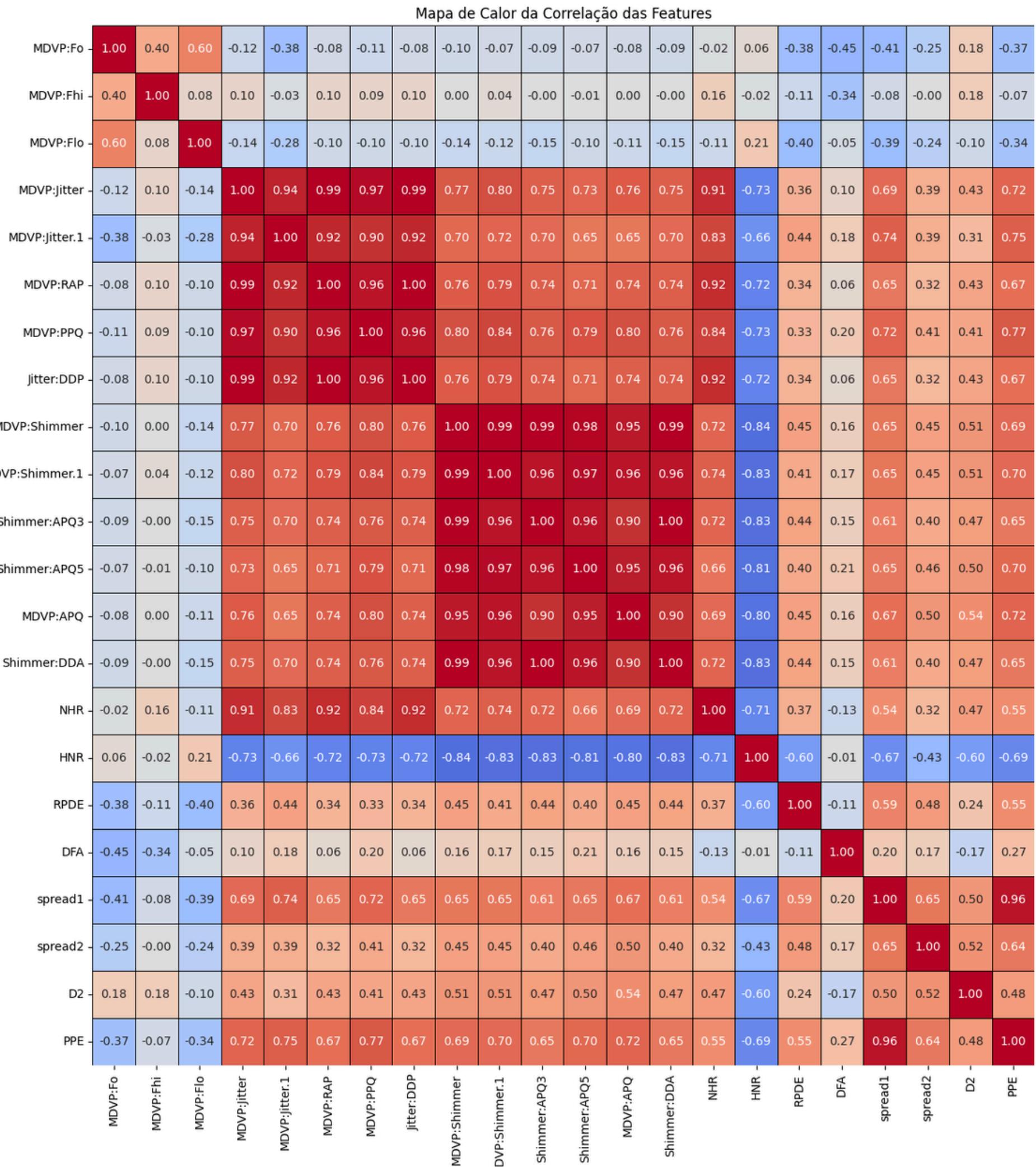
22 variáveis

Variável-alvo: status

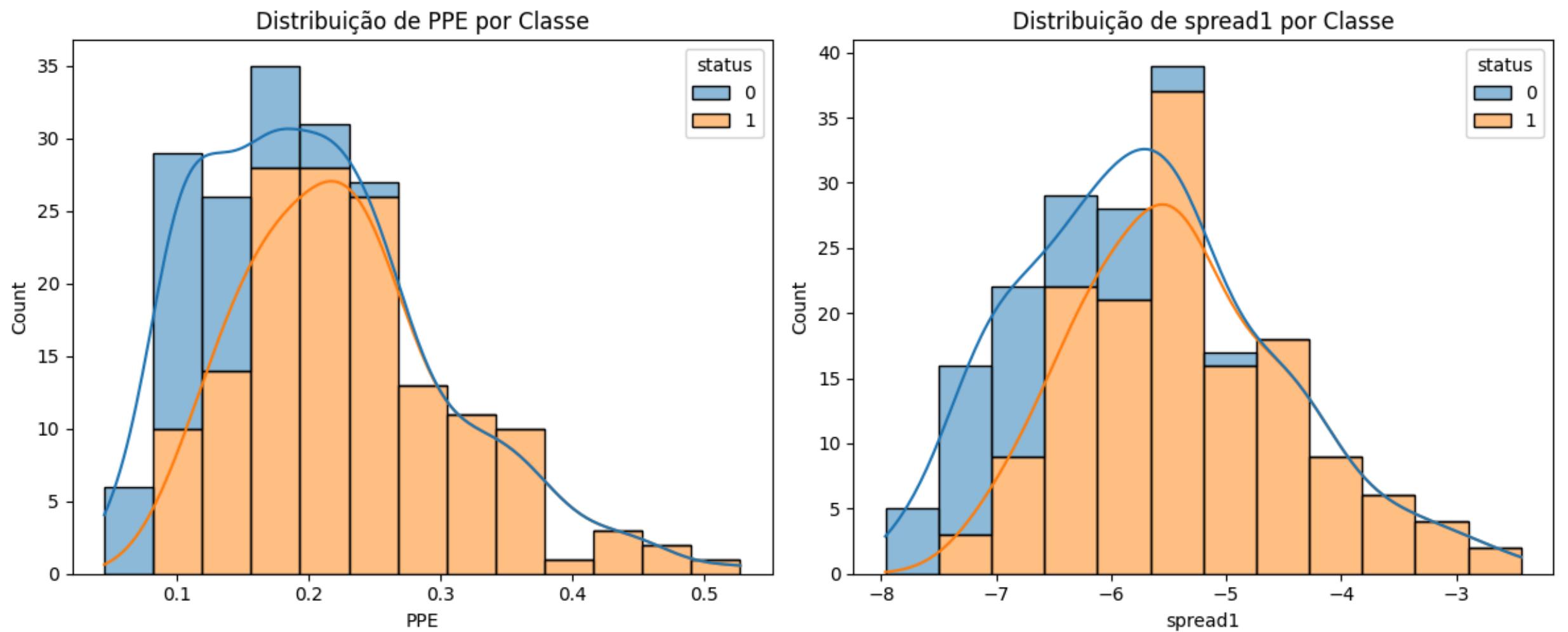
Análise Exploratória dos Dados (EDA)



Análise Exploratória dos Dados (EDA)



Análise Exploratória dos Dados (EDA)



Modelagem Preditiva e Pipeline

6.1 Remoção de Features Não-Preditoras

Removemos o identificador `name`.

```
# Remove a coluna 'name'  
df = df.drop('name', axis=1)  
  
# Exibe as colunas restantes  
print("Colunas após a remoção de 'name':")  
print(df.columns.tolist())  
  
Colunas após a remoção de 'name':  
['MDVP:Fo', 'MDVP:Fhi', 'MDVP:Flo', 'MDVP:Jitter', 'MDVP:Jitter.1', 'MDVP:RAP',  
  
from sklearn.model_selection import train_test_split  
from sklearn.preprocessing import StandardScaler  
  
# Define as features (X) e a variável-alvo (y)  
X = df.drop(['status'], axis=1)  
y = df['status']  
  
# Divindo antes de padronizar para evitar data leakage  
# 1. Divide em 80% treino e 20% teste  
X_treino, X_teste, y_treino, y_teste = train_test_split(  
    X, y, test_size=0.2, random_state=42, stratify=y  
)  
  
# 2. Padroniza as features - fit apenas no treino  
scaler = StandardScaler()  
X_treino_scaled = scaler.fit_transform(X_treino)  
X_teste_scaled = scaler.transform(X_teste) # Apenas transform no teste  
  
# Imprimindo o formato dos conjuntos  
print(f"Shape do conjunto de treino: {X_treino_scaled.shape}")  
print(f"Shape do conjunto de teste: {X_teste_scaled.shape}")  
  
Shape do conjunto de treino: (156, 22)  
Shape do conjunto de teste: (39, 22)
```

7.1 Validação Cruzada Estratificada

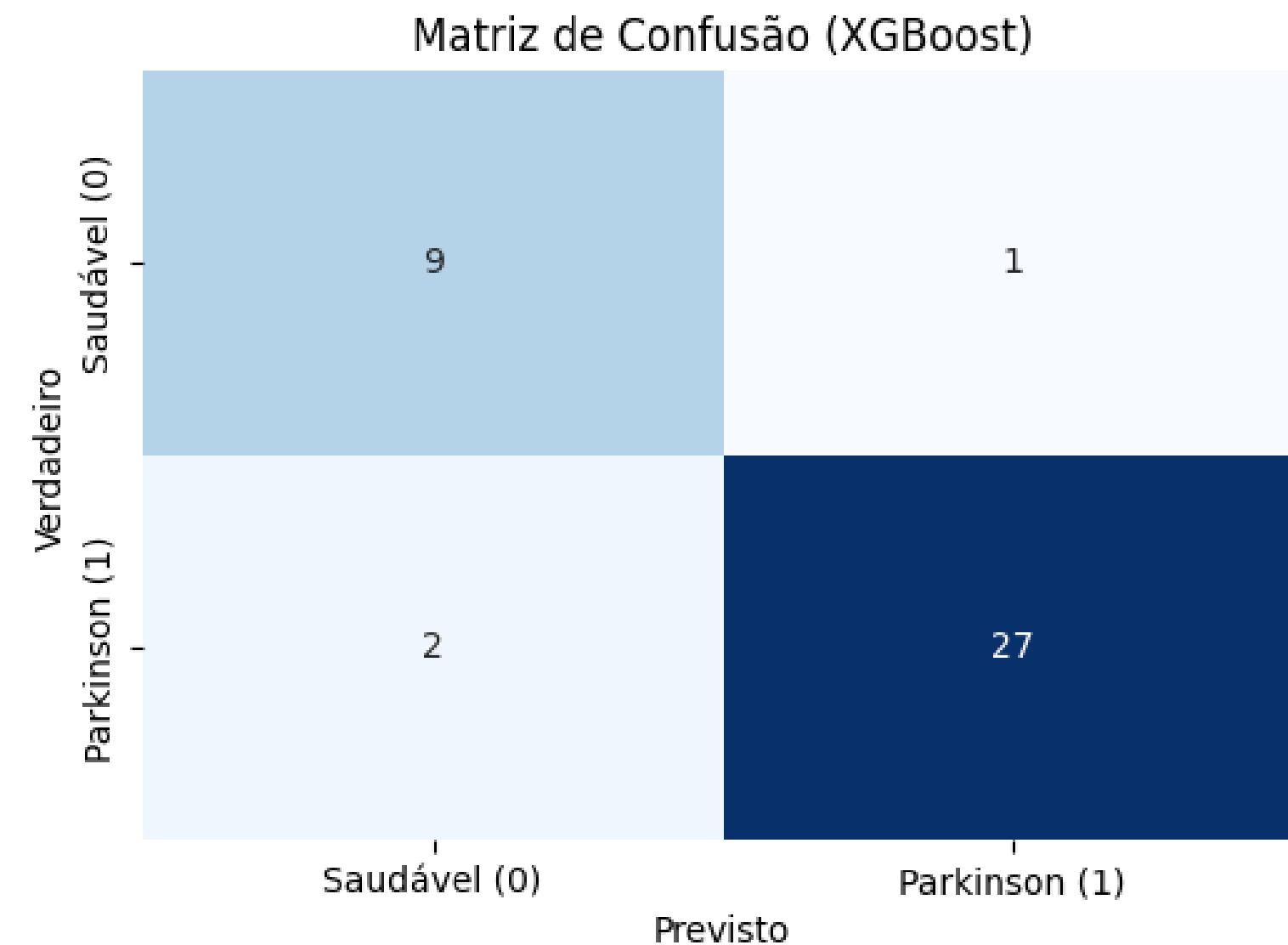
Antes de avaliar no conjunto de teste, vamos usar validação cruzada para obter uma estimativa mais confiável da performance do modelo.

```
from sklearn.model_selection import StratifiedKFold, cross_validate  
import numpy as np  
  
# Definir estratégia de validação cruzada  
cv_strategy = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)  
  
# Definir métricas de interesse  
scoring = {  
    'accuracy': 'accuracy',  
    'recall': 'recall',  
    'f1': 'f1',  
    'roc_auc': 'roc_auc'  
}  
  
print("==== Validação Cruzada Estratificada (5-Fold) ====\n")  
  
==== Validação Cruzada Estratificada (5-Fold) ===
```

Desempenho, Métricas e Interpretação

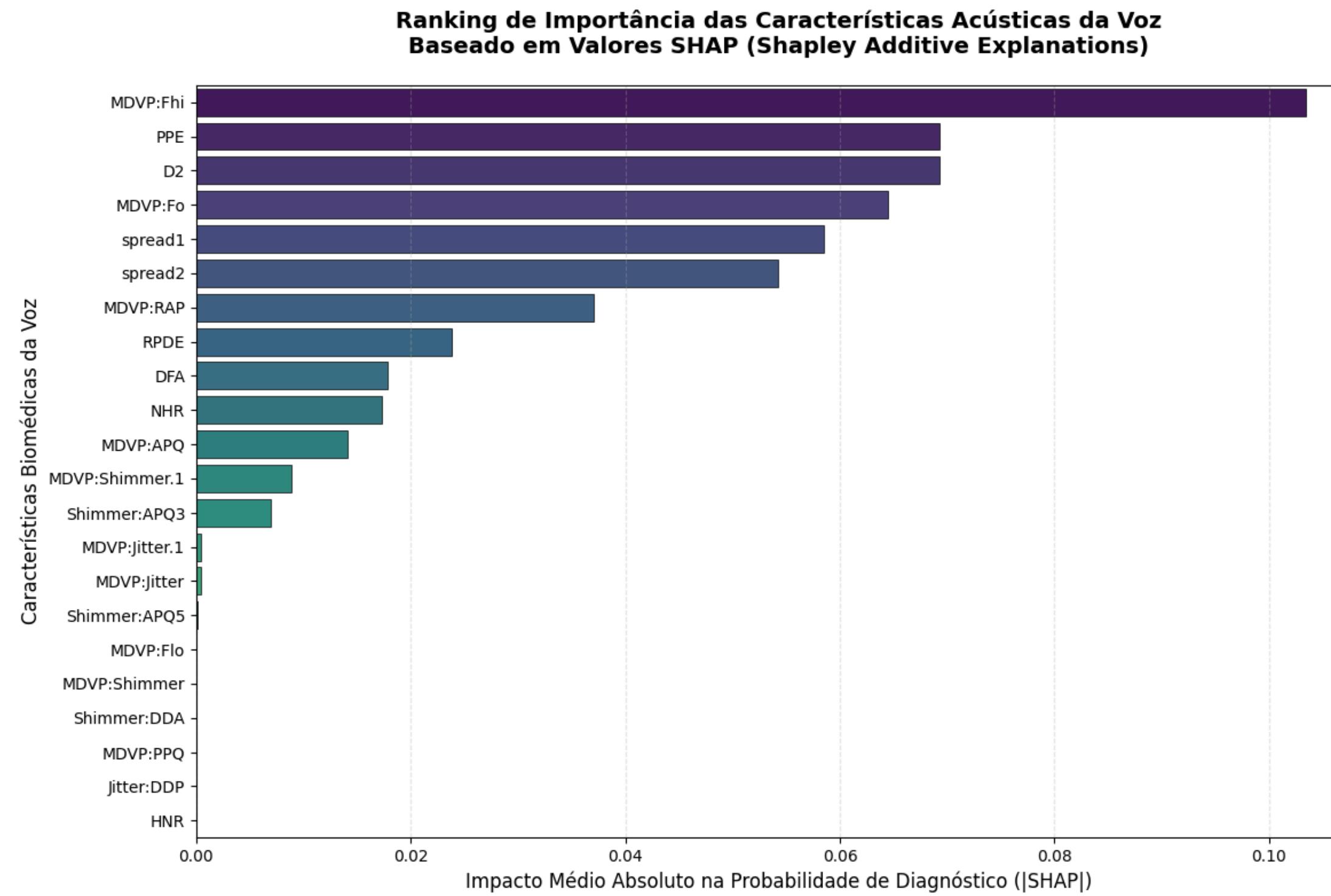
Modelo	Acurácia	Recall	F1-Score
Regr. Logística	0.77	0.72	0.82
Random Forest	0.92	0.96	0.94
XGBoost	0.92	0.93	0.93

Desempenho, Métricas e Interpretação

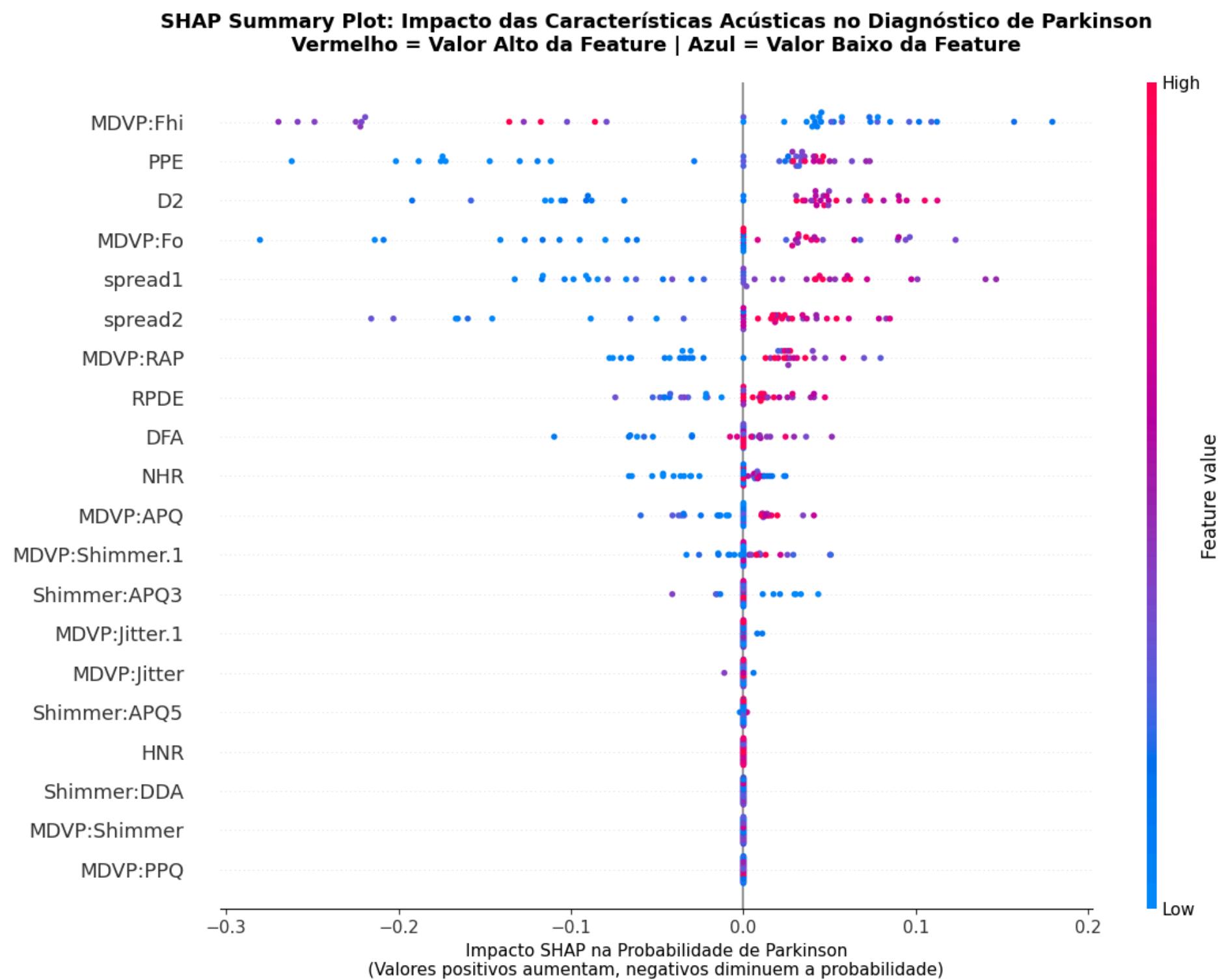


Falsos Negativos: mínimos

Interpretabilidade e SHAP



Interpretabilidade e SHAP



Demonstração e Conclusão

Análise da Doença de Parkinson: Sistema de Suporte ao Diagnóstico

Tech Challenge IADT - Fase 1



Luana dos Santos Caldas
Elaine Martins da Silva