# Machine Learning (P02)

Artificial Intelligence, 2022-23

**Marco Oliveira 16437, André Mandim 21160, Paulo Costa 16445**

## Introduction

Para a realização deste trabalho escolhemos o dataset “Pima-Indians-Diabetes”.

Referência: https://networkrepository.com/pima-indians-diabetes.php
Link GitHub: https://github.com/mandim02/TP2_AI

- André Mandim(21160)
- Marco Oliveira(16437)
- Paulo Costa(16445)

## Preparação de dados

Excerto de dados:

| | 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 |

*Figure 1 - Dados*

Valores em falta afetam o resultado e a taxa de sucesso dos modelos, e como tal, têm de ser tratados.

Alguns dos metodos usados são:

- Atribuir um valor por defeito a valores nulos.
- Remover colunas irrelevantes para o problema.
- Discretizar colunas com valores muito dispersos.
- Por analise dos dados, usando os histograma como apoio, percebemos como podemos tratar algumas das colunas, a seguinte figura mostra os histogramas gerados, conforme os dados de entrada.

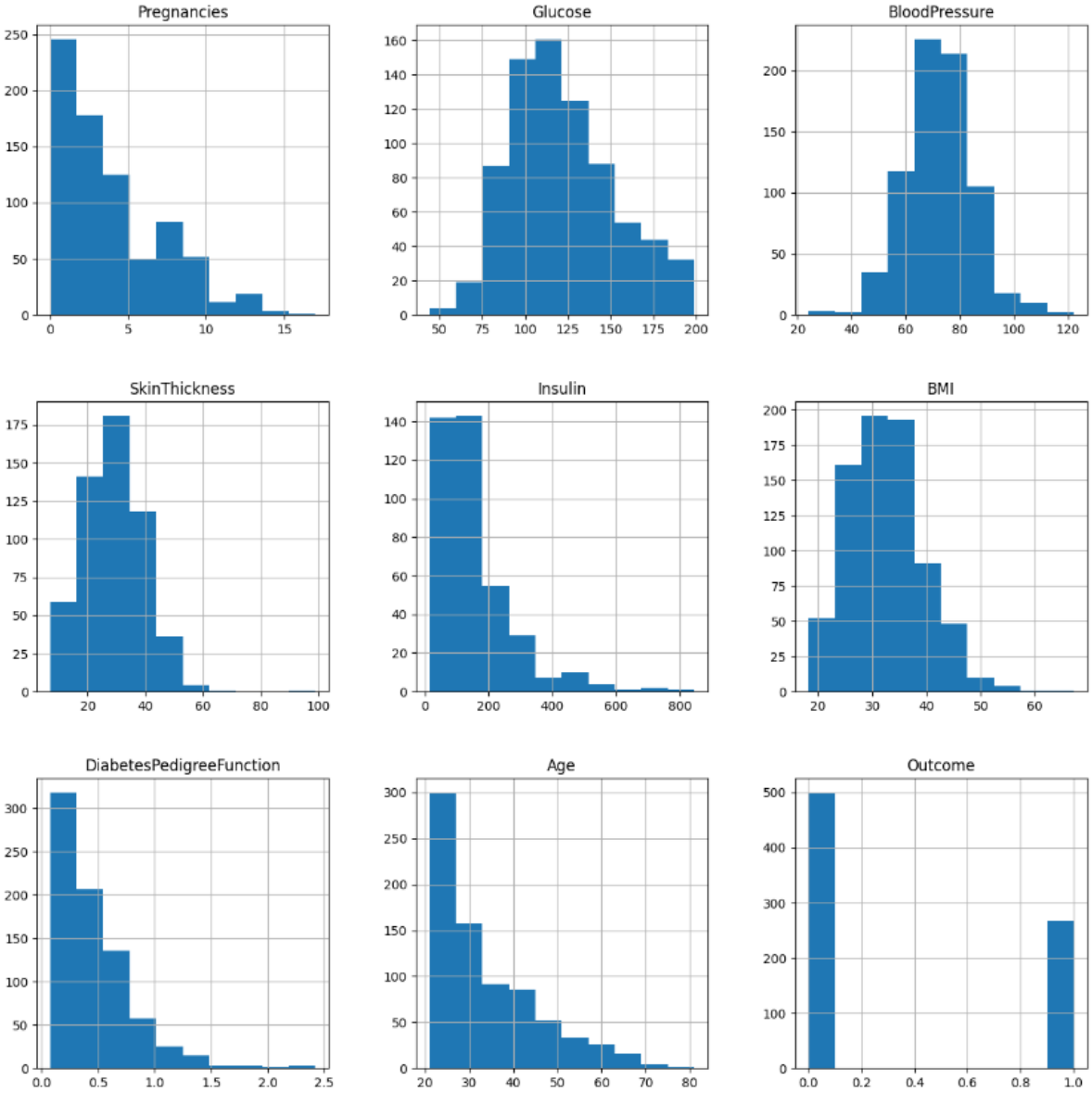


*Figure 2 - Histogramas dados*

A dispersão dos dados dos diffentes campos, afeta de forma significativa a performance dos modelos, e a apreciação dos mesmos, assim normalizamos convertendo os valores para um intervalo [0, 1].

Dois datasets sao necessarios para criar, treinar e validar os modelos. \

Assim sendo, decidimos separar os dados em dois conjuntos, 70% para treino e 30% para teste:

- x_train => dados de treino
- y_train => dados de validação de treino
- x_test => dados de teste
- y_test => dados de validação de teste

## Automatic classification

Prever com base num dataset a probabilidade de alguem ter ou não diabetes.

### K-NN (Nearby Neighbors)

Segue excerto de código do algoritmo implementado:

```
from sklearn.neighbors import KNeighborsClassifier
train_score_list=[]
test_score_list=[]

for each in range (2,31):
    knn=KNeighborsClassifier(n_neighbors=each)
    knn.fit(x_train,y_train)
    test_score_list.append(knn.score(x_test,y_test)*100)
    train_score_list.append(knn.score(x_train,y_train)*100)


print("Melhor precisao(teste):{:.3f} % para K = {}".format(np.max(test_score_list),test_score_list.index(np.max(test_score_list))+ 2))
print("Melhor precisao(treino) is {:.3f}% para K = {}".format(np.max(train_score_list),train_score_list.index(np.max(train_score_list))+2))
```

```
Melhor precisao(teste):78.355 % para K = 23
Melhor precisao(treino) is 87.523% para K = 3
```

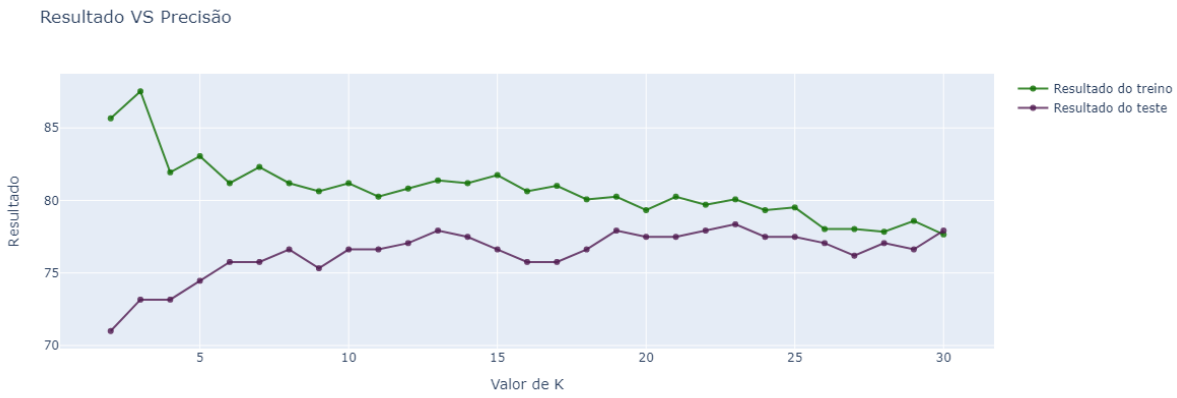


*Figure 3 - KNN Resultados*

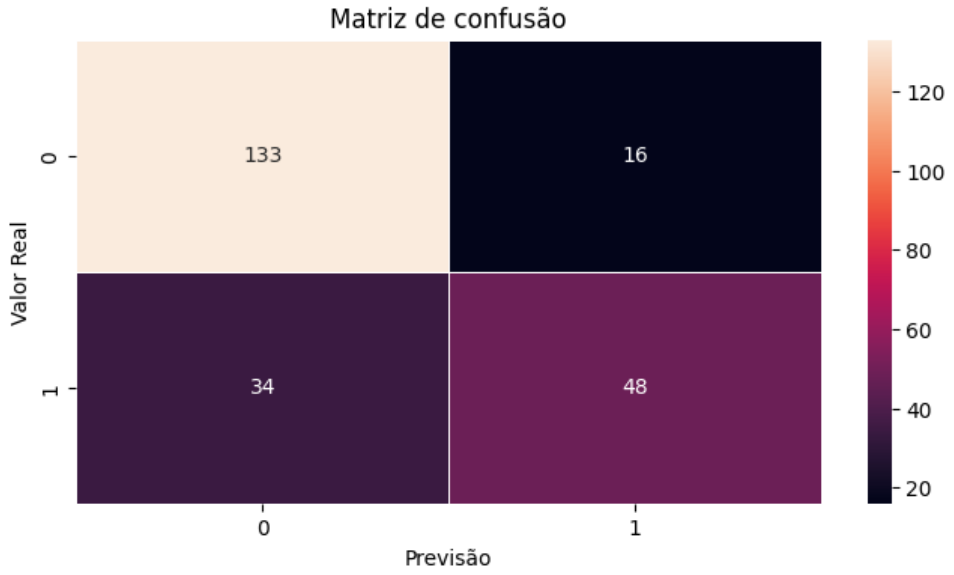


*Figure 4 - KNN Matriz de confusão*

Conforme a figura anterior, Podemos verificar os seguintes resultados:

- Dos 149 pacientes sem diabetes, a previsão acertou em 133 e errou em 16.
- Dos restantes 82 com diabetes, a previsão acertou em 48 e errou em 34.
- Segundo este resultados é seguro afirmar que é mais facil prever pacientes sem diabetes do que pacientes com diabetes.

## Decision Tree

Cada atributo é representado por um "node". Os ramos e as folhas são a estrutura da arvore. O topo é chamado raiz.

Alguns dos algoritmos usados sao o ID3 e IC4.5.

A medida de incerteza num sistema é designada de entropia.

Esta biblioteca permite defenir varios tipos de critério para a decisão das árvores, dos quais decidimos comparar:

- Entropy: indica a variancia de cada elemento em relação com o padrão encontrado,
- Gini: indica a frequência com que um elemento é mal classificado quando lhe é atribuida uma categoria aleatoriamente.

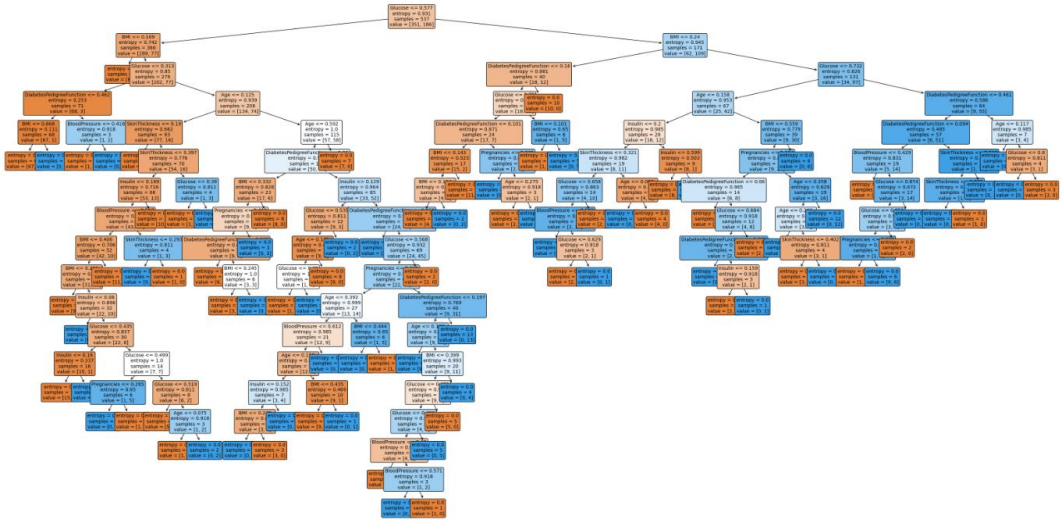

*Figure 5 - Decision Tree - Entropy*

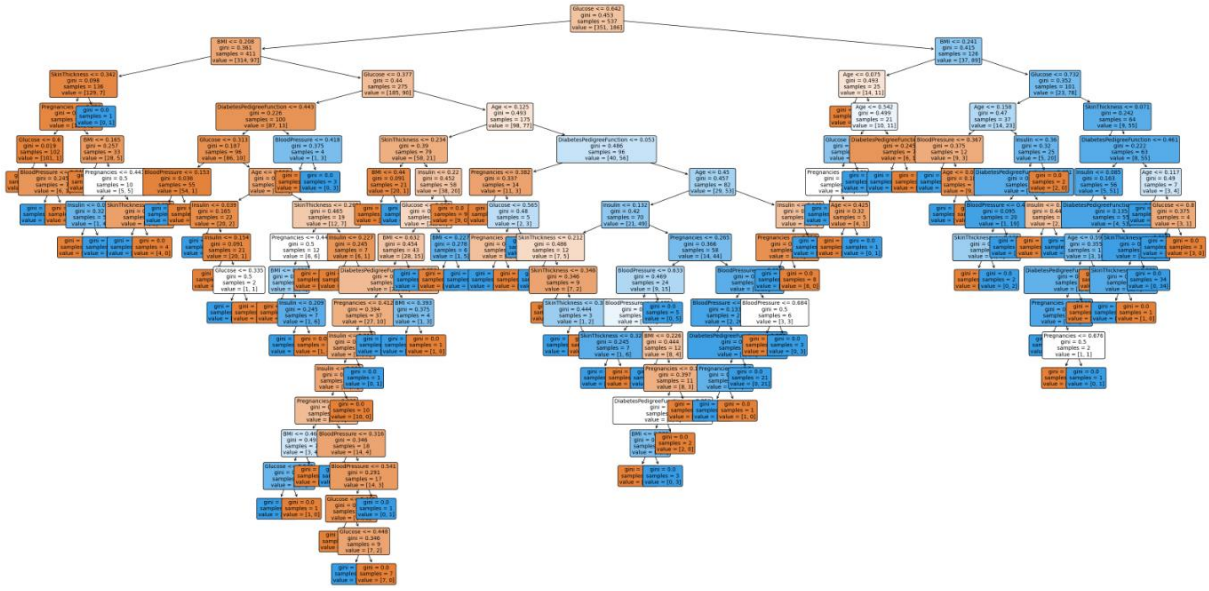

*Figure 6 - Decision Tree - Gini*

Tal como anteriormente, gerando a respetiva matriz de confusão, chegamos aos seguintes valores:

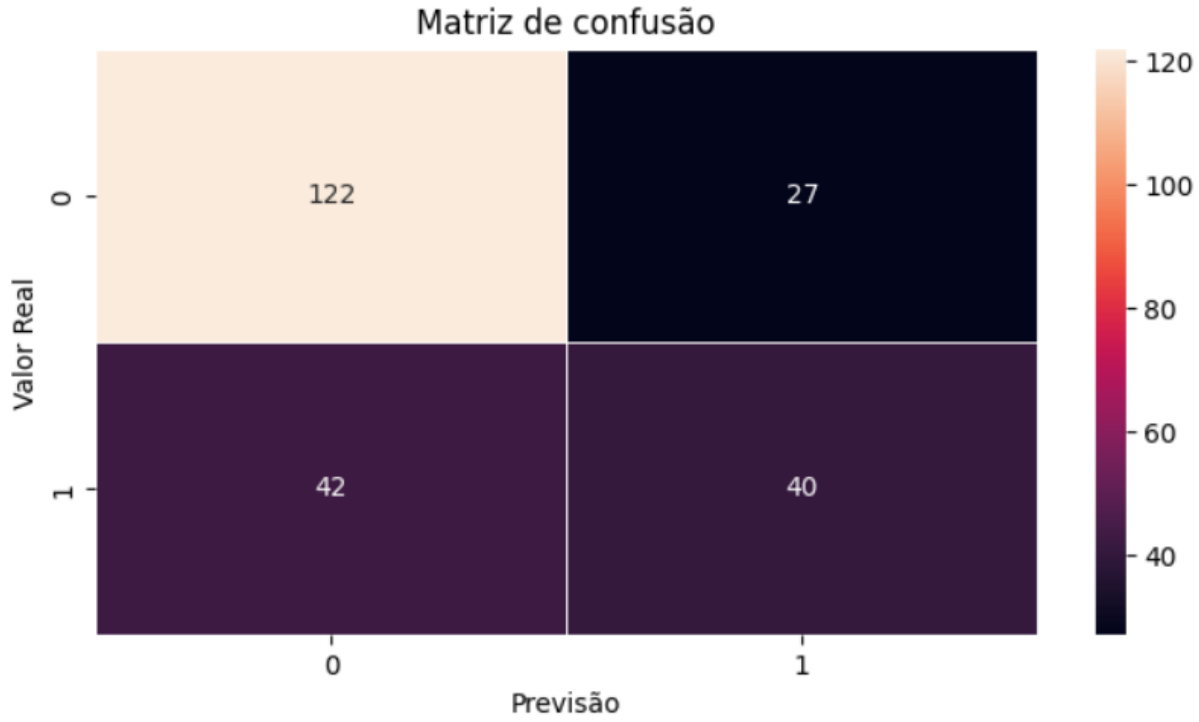


*Figure 7 - Decision Tree - Matriz de Confusão – Entropia*

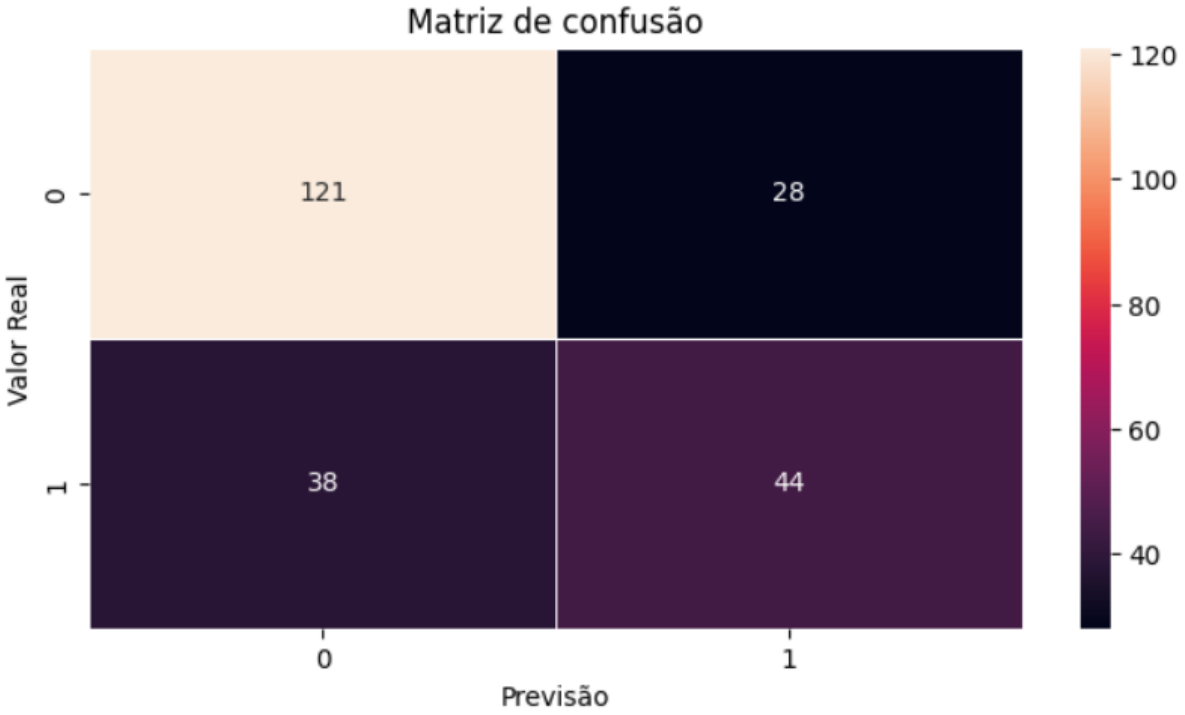


*Figure 8 - Decision Tree - Matriz de Confusão – Gini*

Com base nestes resultados Podemos afirmar, que o critério de entropia foi o mais certeiro nos valores que preveu.

# Clustering

**K-Means**

Colocamos o dataset inicial e dividindo em 2 clusters, numa nova variável.

Para gerar o seguinte gráfico, relacionamos os seguintes dados:

BMI, Glucose e a Função de Diabetes Pedigree:

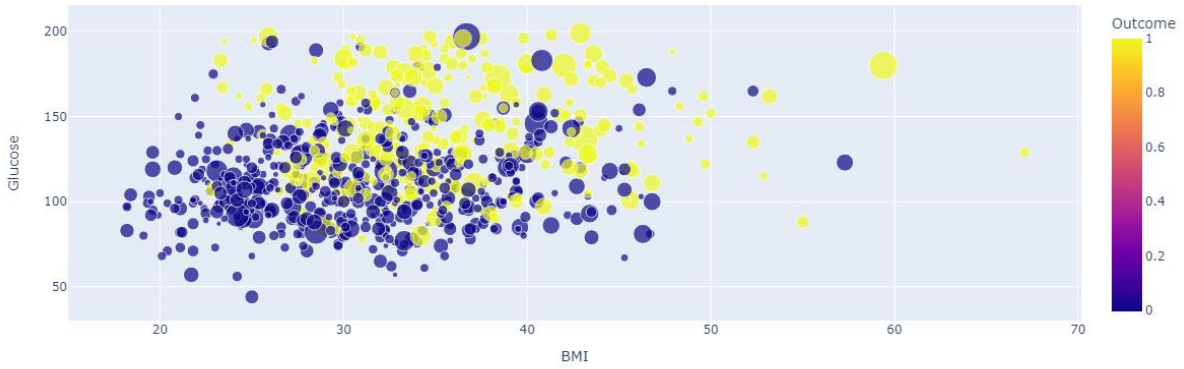


*Figure 9 - K means dados*

Com base nisto, Podemos verificar a seguinte matriz de confusão:

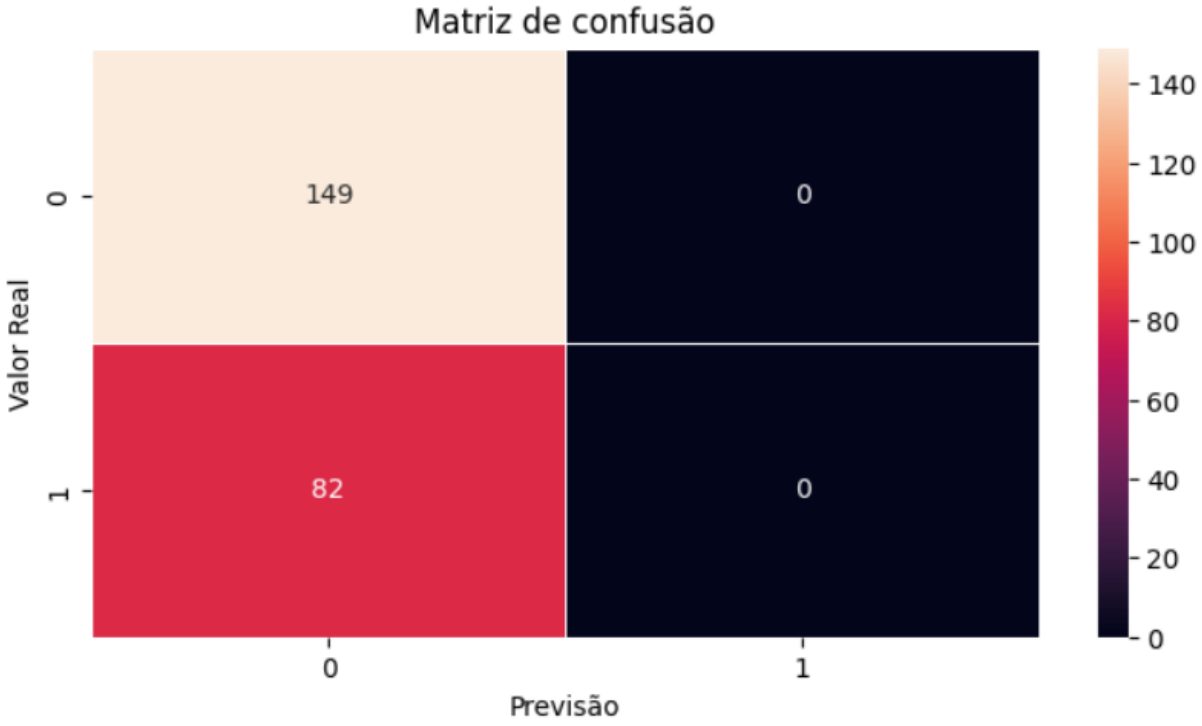


*Figure 10 - K means - matriz de confusão*

## Association rules

Pretende-se descobrir regras de associação entre os vários dados de forma a saber o que influencia e de que forma influenciam o resultado.

Tratamento de dados:

- Remoção dos valores nulos para valores de media e mediana.
- Convertemos os valores discretos em intervaloes de forma a ser mais amigavel a interpretação dos valores
- Para cada um dos campos consideramos quatro intervalos de valores, um para cada quartil, com excepção do campo “Outcome” em que apenas classificamos como diabético ou não diabético

Aplicação do algoritmo:

- Aplicando as seguintes regras:
    - Suporte Diabetes : (numero real de diabeticos) / (numero total de registos)
    - Confiança: (Suporte de A U B) / (Suporte de B)
        - Taxa de, por exemplo, probabilidade de ter diabetes sabendo que tem Glucose >X e Insulina < Y
    - Para os restantes parâmetros consideramos valores irrisórios porque existem muitos atributos

Após aplicação de algoritmo, procuramos as transações que contêm nelas a palavra Diabetic. Ficando então com os seguintes resultados(excerto):

| | 0 | 1 | 2 | 0 | 1 |
|---|---|---|---|---|---|
| **7** | BMI <= 27.5 | Age <= 24.0 | | Not Diabetic | SkinThickness <= 25.0 |
| **14** | Age <= 29.0 | Not Diabetic | SkinThickness <= 25.0 | Insulin <= 121.5 | |
| **17** | Age <= 81.0 | Diabetic | | Pregnancies <= 17.0 | DiabetesPedigreeFunction <= 0.62625 |
| **18** | Age <= 81.0 | Diabetic | | Pregnancies <= 17.0 | Glucose <= 199.0 |
| **20** | BMI <= 27.5 | Glucose <= 99.75 | | Not Diabetic | SkinThickness <= 25.0 |
| **21** | SkinThickness <= 25.0 | | | Insulin <= 121.5 | Not Diabetic |
| **22** | Pregnancies <= 1.0 | BMI <= 27.5 | | Not Diabetic | SkinThickness <= 25.0 |

*Figure 11 - Apriori Results*