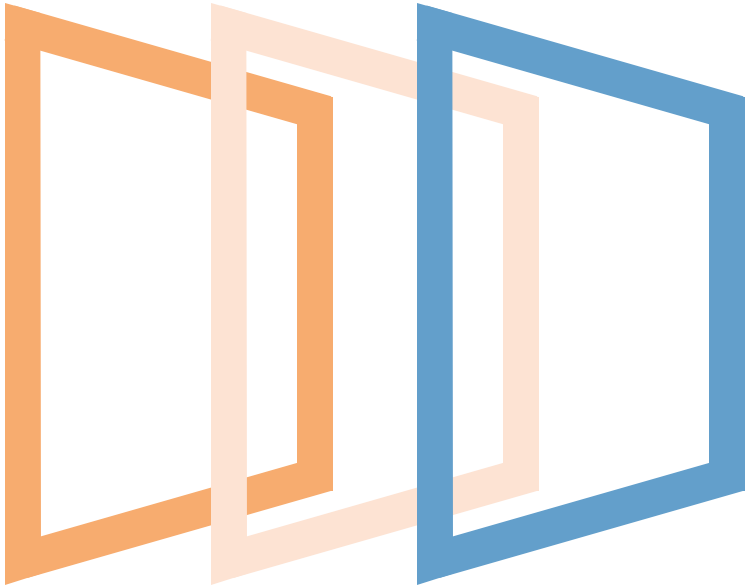
Capacitação Trilhando Caminhos em Ciência de Dados.

Projeto: Conjunto de Dados Avaliação de Riscos de Diabetes

Instrutora: Thaís Ratis

Autores:
- Flávio Silva
- Patrícia Lópes
- Pablo Veinberg



minsait

An Indra company

Definição do problema

O conjunto de dados contém uma gama diversificada de atributos relacionados à saúde, meticulosamente coletados para auxiliar no desenvolvimento de modelos preditivos para identificar indivíduos em risco de diabetes. Nosso objetivo é promover a colaboração e a inovação dentro da comunidade de ciência de dados, levando a um melhor diagnóstico precoce para o diabetes.

Amostra Inicial dos dados

	Id	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	1	6	148	72	35	0	33.6	0.627	50	1
1	2	1	85	66	29	0	26.6	0.351	31	0
2	3	8	183	64	0	0	23.3	0.672	32	1
3	4	1	89	66	23	94	28.1	0.167	21	0
4	5	0	137	40	35	168	43.1	2.288	33	1

Pré-Processamento: Alteração do nome das colunas e exclusão da coluna 'id'

```
data.columns = ['id', 'quant_gravidez', 'glicose', \
                'pressao_saguinea', 'espesura_pele', \
                'insulina', 'imc', 'diabetes_genetica', 'idade', 'target']
```

```
data.drop('id', axis=1, inplace=True)
```

	quant_gravidez	glicose	pressao_saguinea	espesura_pele	insulina	imc	diabetes_genetica	idade	target
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

Pré-Processamento: Duplicidades e valores nulos

```
# Duplicated
data.drop_duplicates(keep='first', inplace=True)
data.duplicated().sum()
```

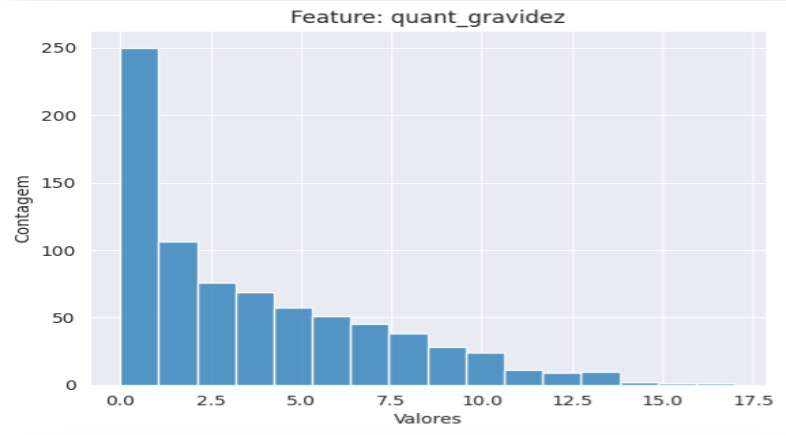
```
# Missing values
data.isnull().sum().sum()
```

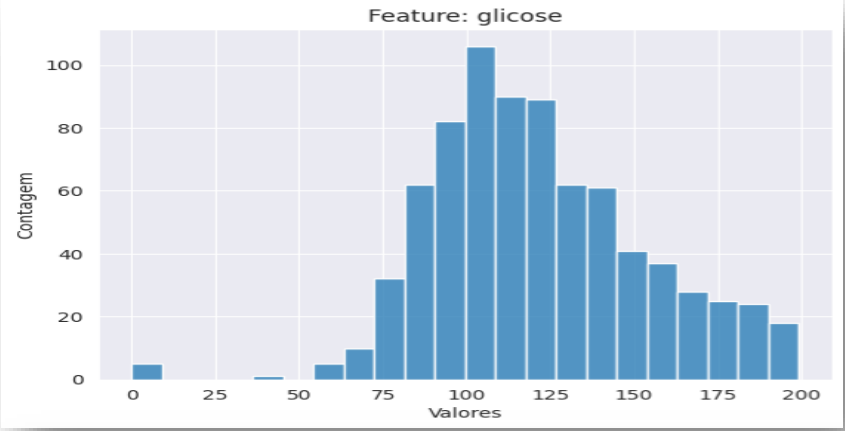
Análise de Dados: correlação entre as variáveis

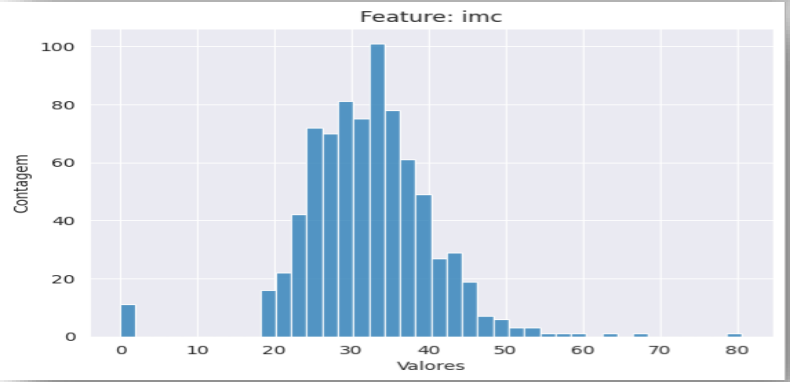
Correlação dos Dados

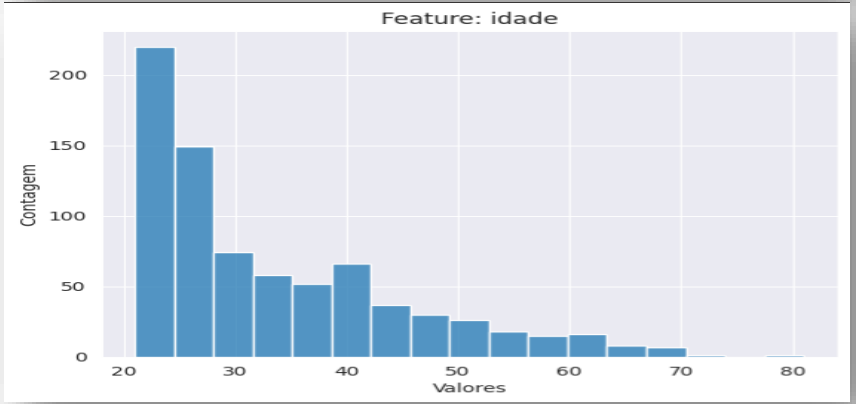
	quant_gravidez	glicose	pressao_saguinea	espesura_pele	insulina	imc	diabetes_genetica	idade	target
quant_gravidez	1.000000	0.124729	0.143599	-0.085663	-0.076876	0.010874	-0.034159	0.532993	0.220380
glicose	0.124729	1.000000	0.140420	0.067604	0.333652	0.231745	0.137337	0.262591	0.459152
pressao_saguinea	0.143599	0.140420	1.000000	0.178080	0.082516	0.249552	0.042145	0.243475	0.073921
espesura_pele	-0.085663	0.067604	0.178080	1.000000	0.434904	0.367135	0.182582	-0.101986	0.078016
insulina	-0.076876	0.333652	0.082516	0.434904	1.000000	0.195511	0.190193	-0.038262	0.127030
imc	0.010874	0.231745	0.249552	0.367135	0.195511	1.000000	0.130382	0.043150	0.264761
diabetes_genetica	-0.034159	0.137337	0.042145	0.182582	0.190193	0.130382	1.000000	0.034839	0.172160
idade	0.532993	0.262591	0.243475	-0.101986	-0.038262	0.043150	0.034839	1.000000	0.244260
target	0.220380	0.459152	0.073921	0.078016	0.127030	0.264761	0.172160	0.244260	1.000000

Distribuição dos dados – Alguns Histogramas

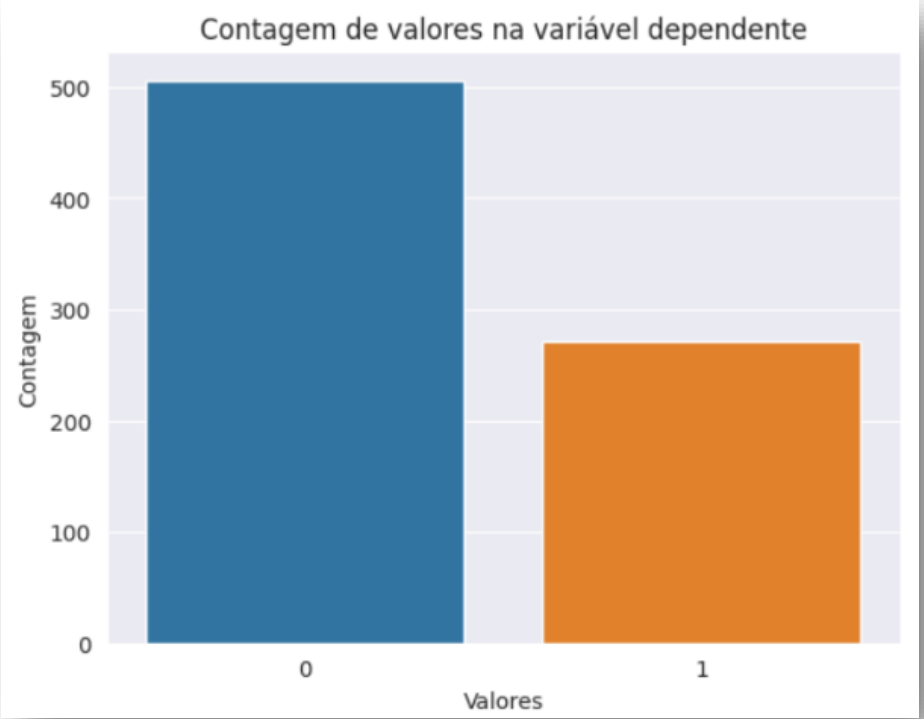








Valores na Variável Dependente



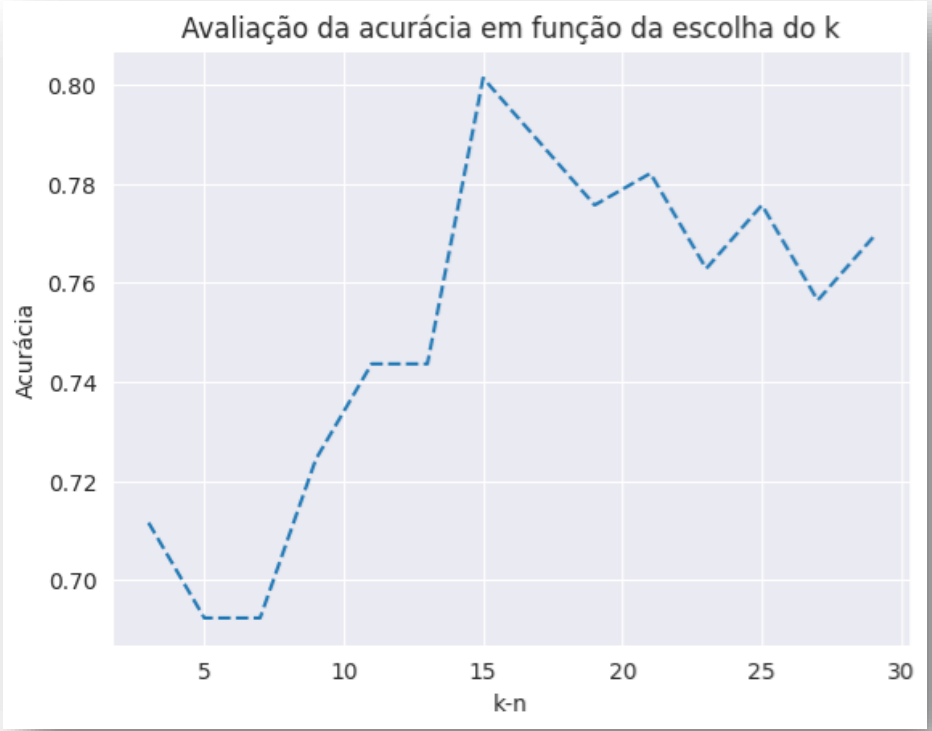
Seleção de Variáveis Dependentes e Independentes

```
1 X = data.drop('target', axis=1) # variáveis indepentes
2 y = data.target # Variável dependente
```

Calculando Vizinhos mais Próximos (KNN)

```python
def results(y_test,y_pred):
    results = confusion_matrix(y_test, y_pred)
    print ('Confusion Matrix :')
    print(results)
    accuracy = accuracy_score(y_test, y_pred)
    print("Accuracy: %.2f%%" % (accuracy * 100.0))
    print ('Report : ')
    print (classification_report(y_test, y_pred))
    return accuracy

def compute_knn(X, y, k, print = False):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    model = KNeighborsClassifier(n_neighbors=k, metric='euclidean', algorithm='auto')
    model.fit(X_train, y_train)

    y_pred = model.predict(X_test)

    accuracy = results(y_test, y_pred)

    return accuracy
```

Calculando Resultados com *K* entre 3 e 30



	k	accuracy
0	3	0.711538
1	5	0.692308
2	7	0.692308
3	9	0.724359
4	11	0.743590
5	13	0.743590
6	15	0.801282
7	17	0.788462
8	19	0.775641
9	21	0.782051
10	23	0.762821
11	25	0.775641
12	27	0.756410
13	29	0.769231

Calculando Resultado com K = 15

```
Confusion Matrix :
[[99 11]
 [20 26]]
Accuracy: 80.13%
Report :
              precision    recall  f1-score   support

           0       0.83      0.90      0.86       110
           1       0.70      0.57      0.63        46

    accuracy                           0.80       156
   macro avg       0.77      0.73      0.75       156
weighted avg       0.79      0.80      0.79       156
```

Mark Making the way forward

An Indra company