```
import pandas as pd
import numpy as np
```

```
data = pd.read_csv("diabetes.csv")
data.head()
```

	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

```
data.isnull().any()
```

```
Pregnancies                 False
Glucose                     False
BloodPressure               False
SkinThickness               False
Insulin                     False
BMI                         False
DiabetesPedigreeFunction    False
Age                         False
Outcome                     False
dtype: bool
```

```
data.describe().T
```

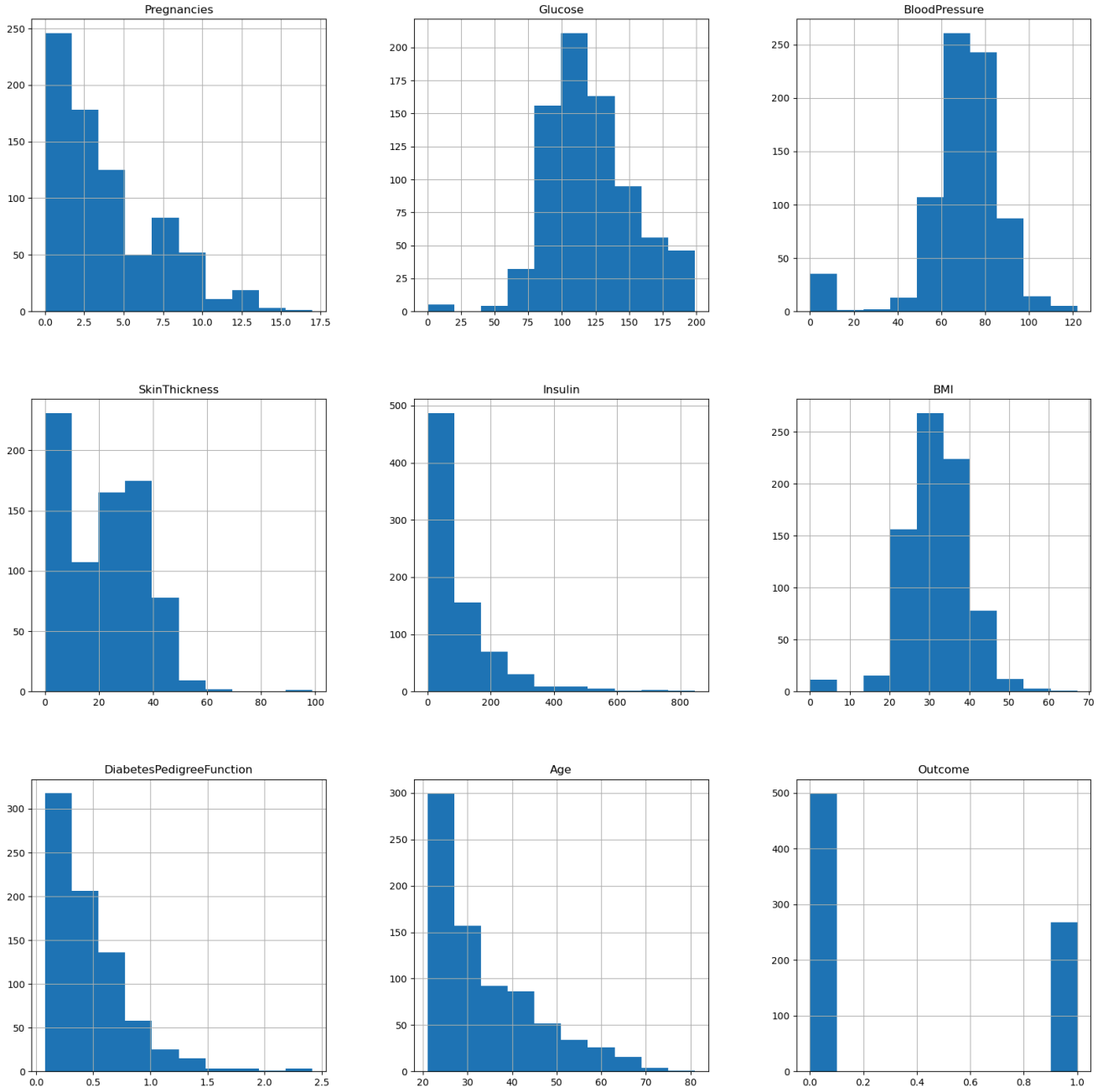
	count	mean	std	min	25%	50%	75%	max
Pregnancies	768.0	3.845052	3.369578	0.000	1.00000	3.0000	6.00000	17.00
Glucose	768.0	120.894531	31.972618	0.000	99.00000	117.0000	140.25000	199.00
BloodPressure	768.0	69.105469	19.355807	0.000	62.00000	72.0000	80.00000	122.00
SkinThickness	768.0	20.536458	15.952218	0.000	0.00000	23.0000	32.00000	99.00
Insulin	768.0	79.799479	115.244002	0.000	0.00000	30.5000	127.25000	846.00
BMI	768.0	31.992578	7.884160	0.000	27.30000	32.0000	36.60000	67.10
DiabetesPedigreeFunction	768.0	0.471876	0.331329	0.078	0.24375	0.3725	0.62625	2.42
Age	768.0	33.240885	11.760232	21.000	24.00000	29.0000	41.00000	81.00
Outcome	768.0	0.348958	0.476951	0.000	0.00000	0.0000	1.00000	1.00

```
data_copy = data.copy(deep = True)
data_copy[['Glucose','BloodPressure','SkinThickness','Insulin','BMI']] = data_copy[['Glucose','BloodPressure','SkinThickness','Insulin',
data_copy.isnull().sum()
```

```
Pregnancies                 0
Glucose                     5
BloodPressure              35
SkinThickness             227
Insulin                   374
BMI                        11
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

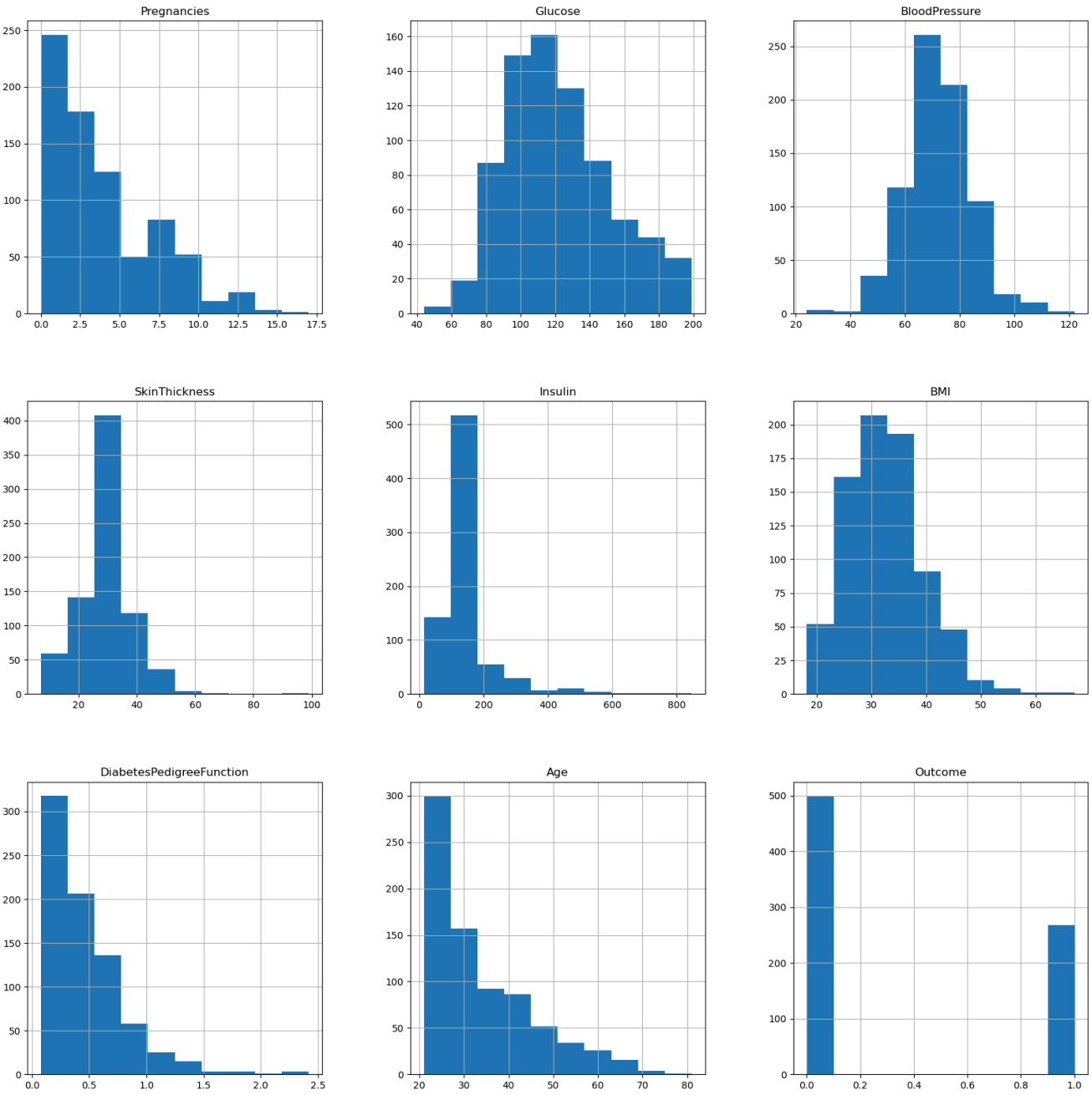
```
p = data.hist(figsize = (20,20))
```

```
Matplotlib is building the font cache; this may take a moment.
```




```
data_copy['Glucose'].fillna(data_copy['Glucose'].mean(), inplace = True)
data_copy['BloodPressure'].fillna(data_copy['BloodPressure'].mean(), inplace = True)
data_copy['SkinThickness'].fillna(data_copy['SkinThickness'].median(), inplace = True)
data_copy['Insulin'].fillna(data_copy['Insulin'].median(), inplace = True)
data_copy['BMI'].fillna(data_copy['BMI'].median(), inplace = True)
```

```
p = data_copy.hist(figsize = (20,20))
```

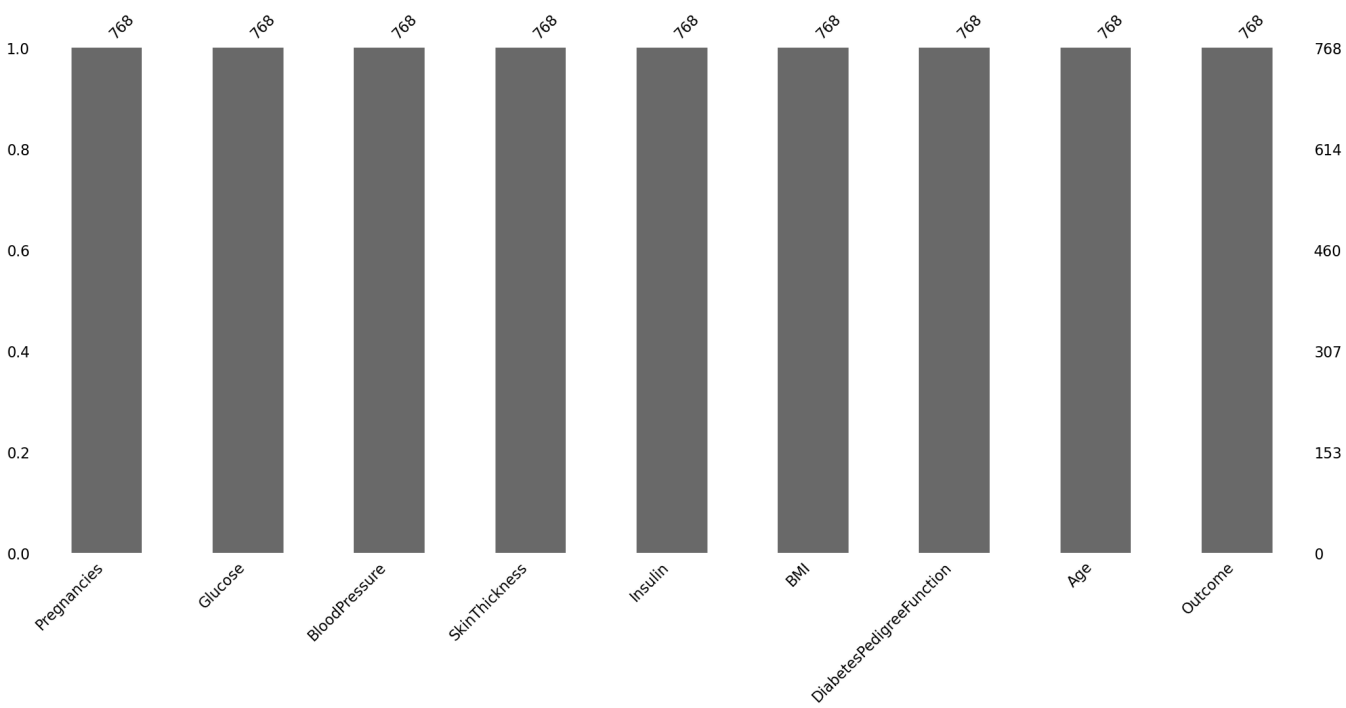



```
pip install missingno
```

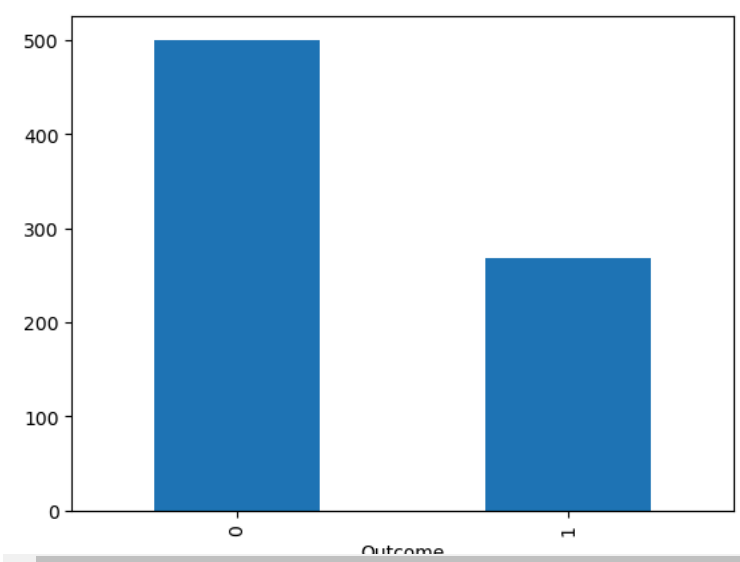
```
Collecting missingnoNote: you may need to restart the kernel to use updated packages.

  Downloading missingno-0.5.2-py3-none-any.whl.metadata (639 bytes)
Requirement already satisfied: numpy in c:\users\hp\anaconda3\anaconda\lib\site-packages (from missingno) (1.26.4)
Requirement already satisfied: matplotlib in c:\users\hp\anaconda3\anaconda\lib\site-packages (from missingno) (3.8.0)
Requirement already satisfied: scipy in c:\users\hp\anaconda3\anaconda\lib\site-packages (from missingno) (1.11.4)
Requirement already satisfied: seaborn in c:\users\hp\anaconda3\anaconda\lib\site-packages (from missingno) (0.12.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib->missingno) (1
Requirement already satisfied: cycler>=0.10 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib->missingno) (0.11.0
Requirement already satisfied: fonttools>=4.22.0 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib->missingno) (4
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib->missingno) (1
Requirement already satisfied: packaging>=20.0 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib->missingno) (23
Requirement already satisfied: pillow>=6.2.0 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib->missingno) (10.2
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib->missingno) (3
Requirement already satisfied: python-dateutil>=2.7 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib->missingno
Requirement already satisfied: pandas>=0.25 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from seaborn->missingno) (2.1.4)
Requirement already satisfied: pytz>=2020.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from pandas>=0.25->seaborn->missing
Requirement already satisfied: tzdata>=2022.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from pandas>=0.25->seaborn->missi
Requirement already satisfied: six>=1.5 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from python-dateutil>=2.7->matplotlib-
```

```
Downloading missingno-0.5.2-py3-none-any.whl (8.7 kB)
Installing collected packages: missingno
Successfully installed missingno-0.5.2
```

```
import missingno as msno
p = msno.bar(data)
```

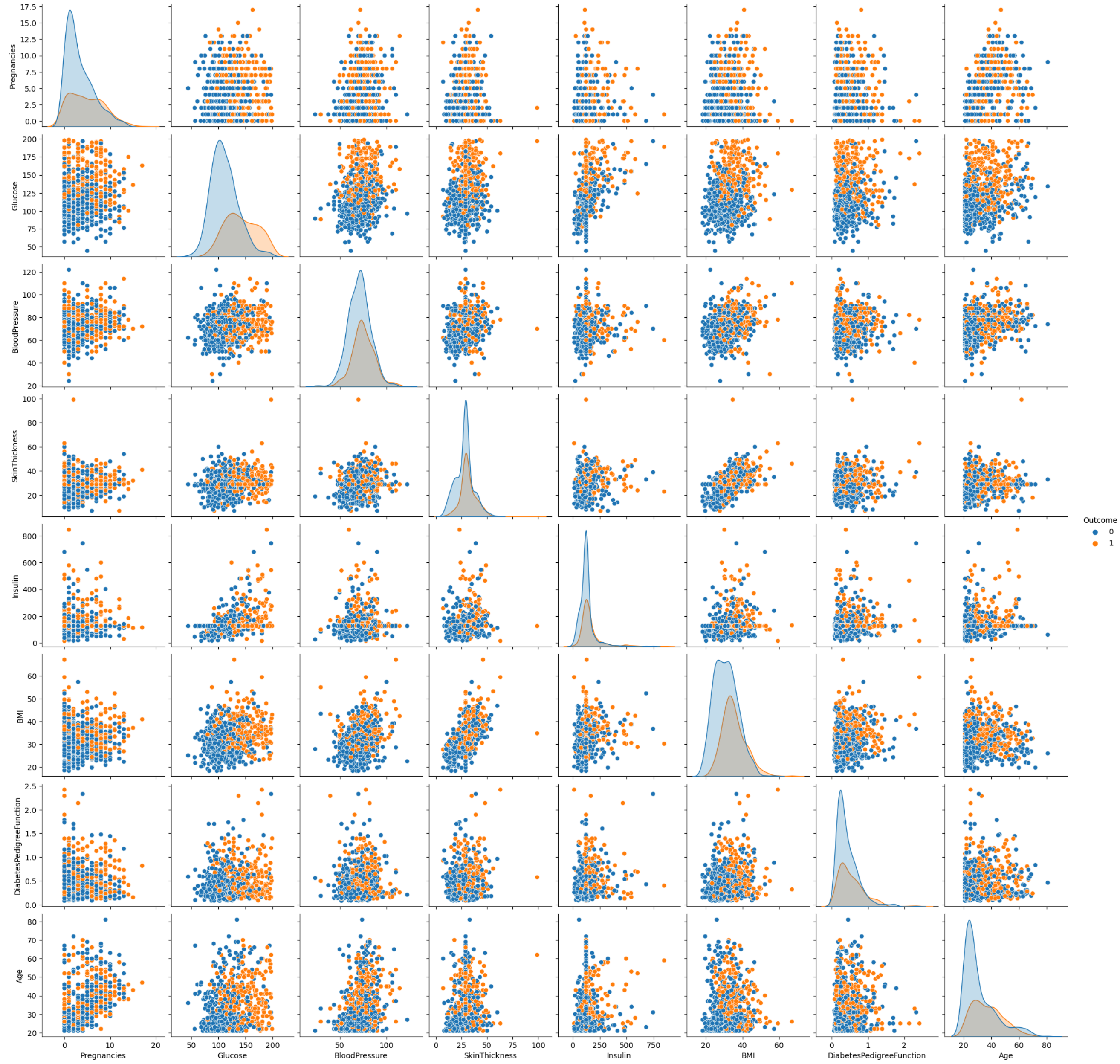



```
p=data.Outcome.value_counts().plot(kind="bar")
```

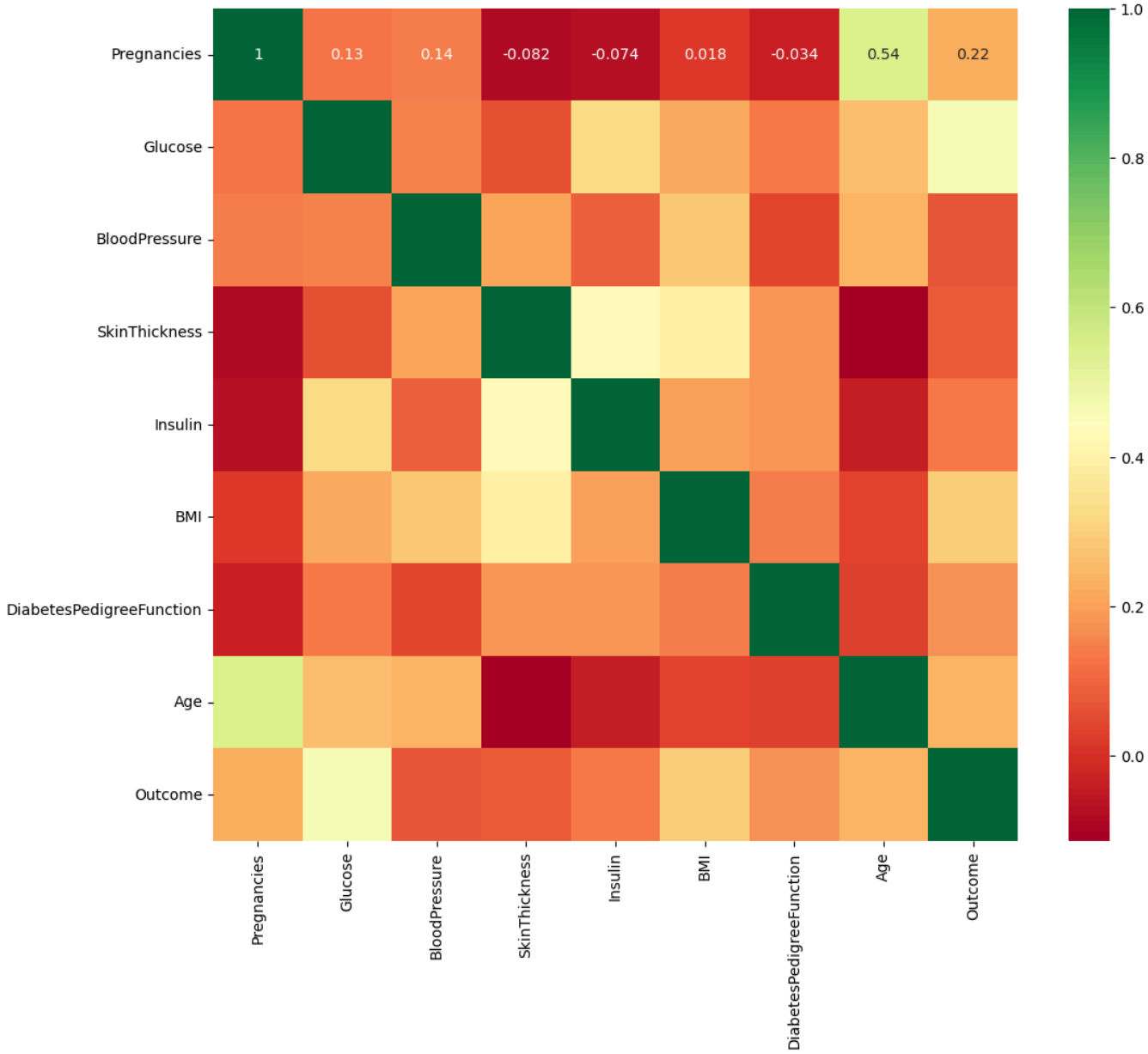



```
import seaborn as sns
p=sns.pairplot(data_copy, hue = 'Outcome')
```

```
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and
  with pd.option_context('mode.use_inf_as_na', True):
```

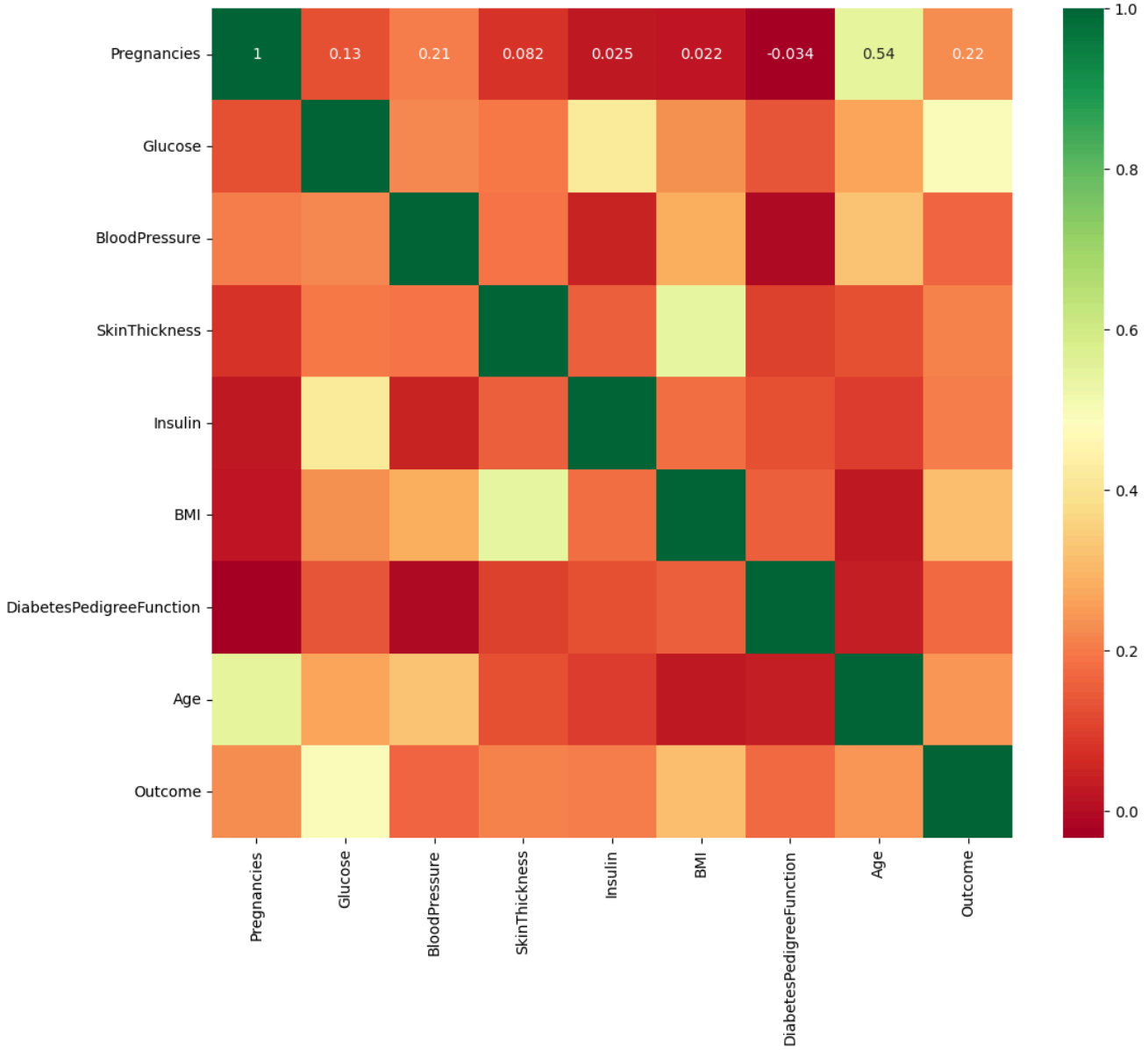



```
import matplotlib.pyplot as plt
plt.figure(figsize=(12,10))
p=sns.heatmap(data.corr(), annot=True,cmap ='RdYlGn')
```



+ Code + Text

```
plt.figure(figsize=(12,10))
p=sns.heatmap(data_copy.corr(), annot=True,cmap ='RdYlGn')
```




```
from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
X = pd.DataFrame(sc_X.fit_transform(data_copy.drop(["Outcome"], axis =1),),columns=['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThic
       'BMI', 'DiabetesPedigreeFunction', 'Age'])

X.head()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age
0	0.639947	0.865108	-0.033518	0.670643	-0.181541	0.166619	0.468492	1.425995
1	-0.844885	-1.206162	-0.529859	-0.012301	-0.181541	-0.852200	-0.365061	-0.190672
2	1.233880	2.015813	-0.695306	-0.012301	-0.181541	-1.332500	0.604397	-0.105584
3	-0.844885	-1.074652	-0.529859	-0.695245	-0.540642	-0.633881	-0.920763	-1.041549
4	1.141852	0.503458	-2.680669	0.670643	0.316566	1.549303	5.484909	0.020496

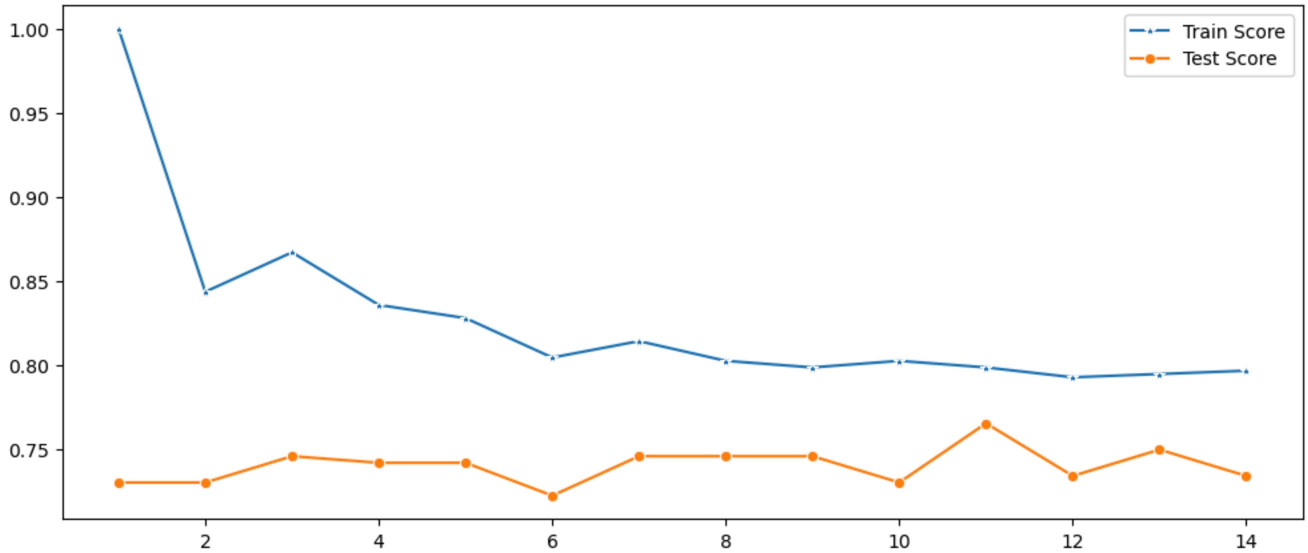
```
y =data_copy.Outcome


from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 1/3, random_state = 42, stratify=y)


from sklearn.neighbors import KNeighborsClassifier

train_scores = []
```

```
test_scores = []

for i in range(1,15):
    knn = KNeighborsClassifier(i)
    knn.fit(X_train, y_train)
    train_scores.append(knn.score(X_train, y_train))
    test_scores.append(knn.score(X_test, y_test))


max_test_score =max(test_scores)


test_score_index = [i for i, v in enumerate(test_scores) if v== max_test_score]

print('Max test score {} % and k = {}'.format(max_test_score*100,list(map(lambda x: x+1, test_score_index))))
```

```
Max test score 76.5625 % and k = [11]
```

```
plt.figure(figsize=(12,5))
p = sns.lineplot(x=range(1,15), y=train_scores, marker='*', label='Train Score')
p = sns.lineplot(x=range(1,15), y=test_scores, marker='o', label='Test Score')

plt.legend()
plt.show()
```

```
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and wil
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and wil
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and wil
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\HP\anaconda3\Anaconda\Lib\site-packages\seaborn\_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and wil
  with pd.option_context('mode.use_inf_as_na', True):
```




```
knn = KNeighborsClassifier(11)

knn.fit(X_train,y_train)
knn.score(X_test,y_test)
```

```
0.765625
```

```
pip install mlxtend
```

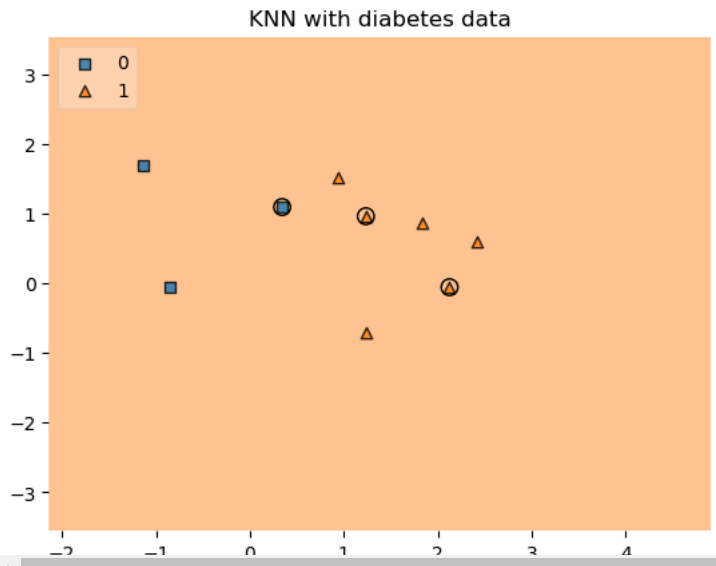
```
Collecting mlxtend
  Downloading mlxtend-0.23.1-py3-none-any.whl.metadata (7.3 kB)
Requirement already satisfied: scipy>=1.2.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from mlxtend) (1.11.4)
Requirement already satisfied: numpy>=1.16.2 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from mlxtend) (1.26.4)
Requirement already satisfied: pandas>=0.24.2 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from mlxtend) (2.1.4)
Requirement already satisfied: scikit-learn>=1.0.2 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from mlxtend) (1.2.2)
Requirement already satisfied: matplotlib>=3.0.0 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from mlxtend) (3.8.0)
Requirement already satisfied: joblib>=0.13.2 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from mlxtend) (1.2.0)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib>=3.0.0->mlxtenc
Requirement already satisfied: cycler>=0.10 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib>=3.0.0->mlxtend) (6
Requirement already satisfied: fonttools>=4.22.0 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib>=3.0.0->mlxter
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib>=3.0.0->mlxter
Requirement already satisfied: packaging>=20.0 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib>=3.0.0->mlxtend
Requirement already satisfied: pillow>=6.2.0 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib>=3.0.0->mlxtend)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib>=3.0.0->mlxtenc
Requirement already satisfied: python-dateutil>=2.7 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from matplotlib>=3.0.0->ml>
```

```
Requirement already satisfied: pytz>=2020.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from pandas>=0.24.2->mlxtend) (2023
Requirement already satisfied: tzdata>=2022.1 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from pandas>=0.24.2->mlxtend) (20
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from scikit-learn>=1.0.2->m
Requirement already satisfied: six>=1.5 in c:\users\hp\anaconda3\anaconda\lib\site-packages (from python-dateutil>=2.7->matplotlib>=
Downloading mlxtend-0.23.1-py3-none-any.whl (1.4 MB)
   ---------------------------------------- 0.0/1.4 MB ? eta -:--:--
   ---------------------------------------- 0.0/1.4 MB ? eta -:--:--
   ---------------------------------------- 0.0/1.4 MB ? eta -:--:--
   - -------------------------------------- 0.1/1.4 MB 544.7 kB/s eta 0:00:03
   ----- ---------------------------------- 0.2/1.4 MB 1.4 MB/s eta 0:00:01
   ------------ --------------------------- 0.5/1.4 MB 2.4 MB/s eta 0:00:01
   -------------------- ------------------- 0.7/1.4 MB 3.2 MB/s eta 0:00:01
   ----------------------- ---------------- 0.8/1.4 MB 3.4 MB/s eta 0:00:01
   ------------------------------ --------- 1.1/1.4 MB 3.5 MB/s eta 0:00:01
   ---------------------------------- ----- 1.2/1.4 MB 3.4 MB/s eta 0:00:01
   ---------------------------------------- - 1.4/1.4 MB 3.4 MB/s eta 0:00:01
   ---------------------------------------- 1.4/1.4 MB 3.2 MB/s eta 0:00:00
Installing collected packages: mlxtend
Successfully installed mlxtend-0.23.1
Note: you may need to restart the kernel to use updated packages.
```

```
# Convert X and X_test to NumPy arrays before fitting the classifier
X_np = X.values
X_test_np = X_test.values

# Train the KNN model using NumPy arrays
knn.fit(X_np, y)
```

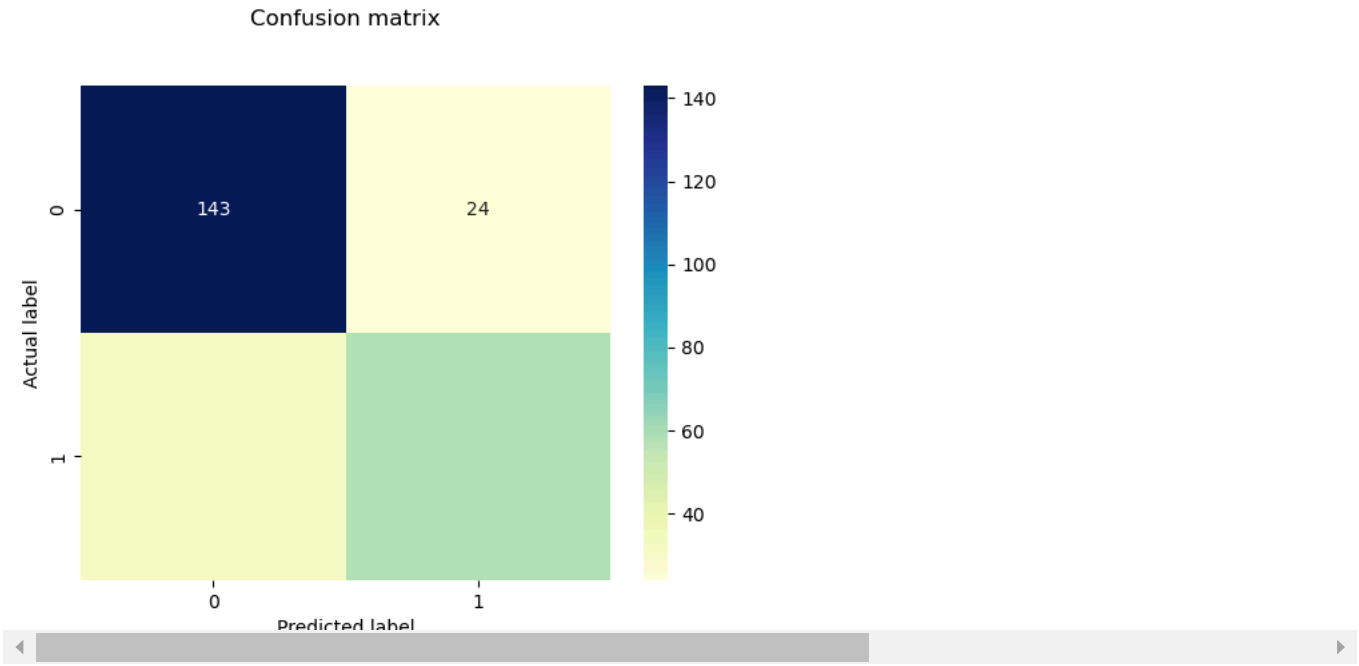
```
KNeighborsClassifier
KNeighborsClassifier(n_neighbors=11)
```

```
from mlxtend.plotting import plot_decision_regions

value = 20000
width = 20000

# Use NumPy arrays for plotting
plot_decision_regions(X_np, y.values, clf=knn, legend=2,
                      filler_feature_values={2: value, 3: value, 4: value, 5: value, 6: value, 7: value},
                      filler_feature_ranges={2: width, 3: width, 4: width, 5: width, 6: width, 7: width},
                      X_highlight=X_test_np)
plt.title("KNN with diabetes data")
plt.show()
```



```
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, fbeta_score
y_pred = knn.predict(X_test)

cnf_matrix = confusion_matrix(y_test, y_pred)



p = sns.heatmap(pd.DataFrame(cnf_matrix), annot=True, cmap="YlGnBu" ,fmt='g')
plt.title('Confusion matrix', y=1.1)
plt.ylabel('Actual label')
plt.xlabel('Predicted label')
```

```
Text(0.5, 23.52222222222222, 'Predicted label')
```




```
def model_evaluation(y_test, y_pred, model_name):
    acc = accuracy_score(y_test, y_pred)
    prec = precision_score(y_test, y_pred)
    rec = recall_score(y_test, y_pred)
    f1 = f1_score(y_test, y_pred)
    f2 = fbeta_score(y_test, y_pred, beta = 2.0)

    results = pd.DataFrame([[model_name, acc, prec, rec, f1, f2]],
                      columns = ["Model", "Accuracy", "Precision", "Recall",
                                 "F1 SCore", "F2 Score"])
    results = results.sort_values(["Precision", "Recall", "F2 Score"], ascending = False)
    return results



model_evaluation(y_test, y_pred, "KNN")
```

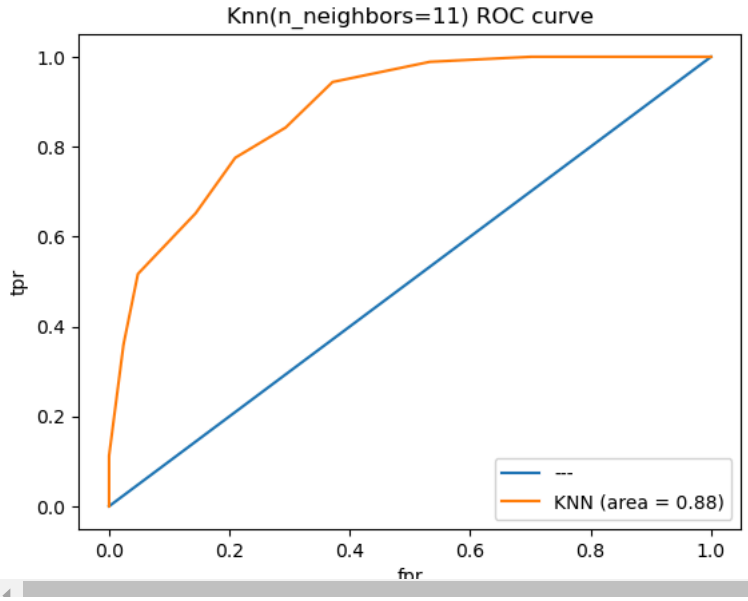
	Model	Accuracy	Precision	Recall	F1 SCore	F2 Score
0	KNN	0.785156	0.707317	0.651685	0.678363	0.6621

```
from sklearn.metrics import classification_report
print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.82      0.86      0.84       167
           1       0.71      0.65      0.68        89

    accuracy                           0.79       256
   macro avg       0.76      0.75      0.76       256
weighted avg       0.78      0.79      0.78       256
```

```
from sklearn.metrics import auc, roc_auc_score, roc_curve

y_pred_proba = knn.predict_proba(X_test)[:,-1]
fpr, tpr, threshold = roc_curve(y_test, y_pred_proba)
```

```
classifier_roc_auc = roc_auc_score(y_test, y_pred_proba)
plt.plot([0,1],[0,1], label = "---")

plt.plot(fpr, tpr, label ='KNN (area = %0.2f)' % classifier_roc_auc)
plt.xlabel("fpr")
plt.ylabel("tpr")
plt.title('Knn(n_neighbors=11) ROC curve')
plt.legend(loc="lower right", fontsize = "medium")
plt.xticks(rotation=0, horizontalalignment="center")
plt.yticks(rotation=0, horizontalalignment="right")
plt.show()
```




```
from sklearn.model_selection import GridSearchCV
parameters_grid = {"n_neighbors": np.arange(0,50)}
knn= KNeighborsClassifier()
knn_GSV = GridSearchCV(knn, param_grid=parameters_grid, cv = 5)
knn_GSV.fit(X, y)
```