```
%matplotlib inline
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
diabetes = pd.read_csv('/content/diabetes2.csv')
diabetes.columns
```

```
Index(['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',
       'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome'],
      dtype='object')
```

```
diabetes.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

```
print("Diabetes data set dimensions : {}".format(diabetes.shape))
```

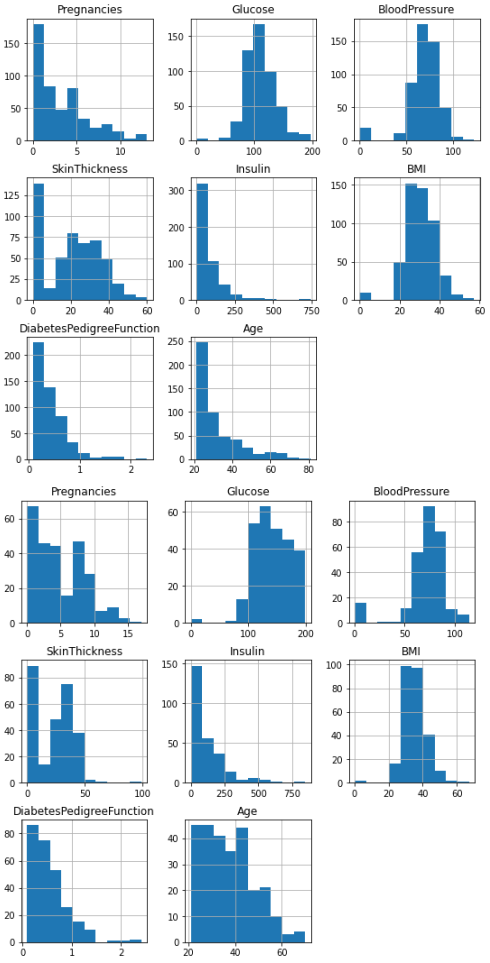
```
Diabetes data set dimensions : (768, 9)
```

```
diabetes.groupby('Outcome').size()
```

```
Outcome
0    500
1    268
dtype: int64
```

```
diabetes.groupby('Outcome').hist(figsize=(9, 9))
```

```
Outcome
0    [[AxesSubplot(0.125,0.670278;0.215278x0.209722...
1    [[AxesSubplot(0.125,0.670278;0.215278x0.209722...
dtype: object
```




```
diabetes.isnull().sum()
diabetes.isna().sum()
```

```
Pregnancies                 0
Glucose                     0
BloodPressure               0
SkinThickness               0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
```

```
Outcome                   0
dtype: int64
```

```
print("Total : ", diabetes[diabetes.BloodPressure == 0].shape[0])
```

```
Total :  35
```

```
print(diabetes[diabetes.BloodPressure == 0].groupby('Outcome')['Age'].count())
```

```
Outcome
0    19
1    16
Name: Age, dtype: int64
```

```
print("Total : ", diabetes[diabetes.Glucose == 0].shape[0])
```

```
Total :  5
```

```
diabetes_mod = diabetes[(diabetes.BloodPressure != 0) & (diabetes.BMI != 0) & (diabetes.Glucose != 0)]
print(diabetes_mod.shape)
```

```
(724, 9)
```

```
feature_names = ['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age']
X = diabetes_mod[feature_names]
y = diabetes_mod.Outcome
```

1. Comparing classification models

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import GradientBoostingClassifier
```

```
models = []
models.append(('KNN', KNeighborsClassifier()))
models.append(('SVC', SVC()))
models.append(('LR', LogisticRegression()))
models.append(('DT', DecisionTreeClassifier()))
models.append(('GNB', GaussianNB()))
models.append(('RF', RandomForestClassifier()))
models.append(('GB', GradientBoostingClassifier()))
```

```
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.metrics import accuracy_score
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify = diabetes_mod.Outcome, random_state=0)
```

```
names = []
scores = []
for name, model in models:
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    scores.append(accuracy_score(y_test, y_pred))
    names.append(name)
tr_split = pd.DataFrame({'Name': names, 'Score': scores})
print(tr_split)
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
  Name     Score
0  KNN  0.729282
1  SVC  0.740331
2   LR  0.762431
3   DT  0.740331
4  GNB  0.734807
5   RF  0.756906
6   GB  0.773481
```

```
from sklearn.model_selection import KFold
names = []
scores = []
for name, model in models:

    kfold = KFold(n_splits=10, random_state=10)
    score = cross_val_score(model, X, y, cv=kfold, scoring='accuracy').mean()

    names.append(name)
    scores.append(score)
kf_cross_val = pd.DataFrame({'Name': names, 'Score': scores})
print(kf_cross_val)
```

```
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
```

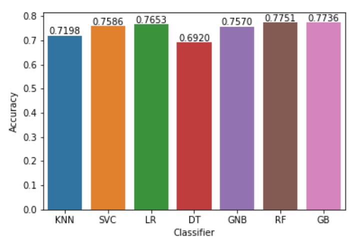
```
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/dist-packages/sklearn/linear_model/_logistic.py:940: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/dist-packages/sklearn/model_selection/_split.py:296: FutureWarning: Setting a random_state has no effect since shuffle is False. This will raise an error in 0.24. You should
  FutureWarning
/usr/local/lib/python3.7/dist-packages/sklearn/model_selection/_split.py:296: FutureWarning: Setting a random_state has no effect since shuffle is False. This will raise an error in 0.24. You should
  FutureWarning
/usr/local/lib/python3.7/dist-packages/sklearn/model_selection/_split.py:296: FutureWarning: Setting a random_state has no effect since shuffle is False. This will raise an error in 0.24. You should
  FutureWarning
/usr/local/lib/python3.7/dist-packages/sklearn/model_selection/_split.py:296: FutureWarning: Setting a random_state has no effect since shuffle is False. This will raise an error in 0.24. You should
  FutureWarning
  Name     Score
0  KNN  0.719787
1  SVC  0.758581
2   LR  0.765278
3   DT  0.691971
4  GNB  0.757021
5   RF  0.775133
6   GB  0.773649
```

```
axis = sns.barplot(x = 'Name', y = 'Score', data = kf_cross_val)
axis.set(xlabel='Classifier', ylabel='Accuracy')
for p in axis.patches:
    height = p.get_height()
    axis.text(p.get_x() + p.get_width()/2, height + 0.005, '{:1.4f}'.format(height), ha="center")

plt.show()
```



2. Comparing Classifiers with lazypredict

```
!pip install lazypredict
```

```
Collecting lazypredict
  Downloading https://files.pythonhosted.org/packages/97/38/cadb2b79268c7f82f6b027bf0b2f68750aafc5c70b6e1bc46b357386e07b/lazypredict-0.2.9-py2.py3-none-any.whl
Collecting PyYAML==5.3.1
  Downloading https://files.pythonhosted.org/packages/64/c2/b80047c7ac2478f9501676c988a5411ed5572f35d1beff9cae07d321512c/PyYAML-5.3.1.tar.gz (269kB)
     |████████████████████████████████| 276kB 10.1MB/s
Collecting xgboost==1.1.1
  Downloading https://files.pythonhosted.org/packages/7c/32/a11befbb003e0e6b7e062a77f010dfcec0ec3589be537b02d2eb2ff93b9a/xgboost-1.1.1-py3-none-manylinux2010_x86_64.whl (127.6MB)
     |████████████████████████████████| 127.6MB 79kB/s
Collecting joblib==1.0.0
  Downloading https://files.pythonhosted.org/packages/34/5b/bd0f0fb5564183884d8e35b81d06d7ec06a20d1a0c8b4c407f1554691dce/joblib-1.0.0-py3-none-any.whl (302kB)
     |████████████████████████████████| 307kB 44.3MB/s
Collecting pandas==1.0.5
  Downloading https://files.pythonhosted.org/packages/af/f3/683bf2547a3eaeec15b39cef86f61e921b3b187f250fcd2b5c5fb4386369/pandas-1.0.5-cp37-cp37m-manylinux1_x86_64.whl (10.1MB)
     |████████████████████████████████| 10.1MB 22.6MB/s
Collecting scikit-learn==0.23.1
  Downloading https://files.pythonhosted.org/packages/b8/7e/74e707b66490d4eb05f702966ad0990881127acecf9d5cdcef3c95ec6c16/scikit_learn-0.23.1-cp37-cp37m-manylinux1_x86_64.whl (6.8MB)
     |████████████████████████████████| 6.8MB 28.2MB/s
Requirement already satisfied: six==1.15.0 in /usr/local/lib/python3.7/dist-packages (from lazypredict) (1.15.0)
Collecting tqdm==4.56.0
  Downloading https://files.pythonhosted.org/packages/80/02/8f8880a4fd6625461833abcf679d4c12a44c76f9925f92bf212bb6cefaad/tqdm-4.56.0-py2.py3-none-any.whl (72kB)
     |████████████████████████████████| 81kB 10.5MB/s
Requirement already satisfied: click==7.1.2 in /usr/local/lib/python3.7/dist-packages (from lazypredict) (7.1.2)
Collecting scipy==1.5.4
  Downloading https://files.pythonhosted.org/packages/dc/7e/8f6a79b102ca1ea928bae8998b05bf5dc24a90571db13cd119f275ba6252/scipy-1.5.4-cp37-cp37m-manylinux1_x86_64.whl (25.9MB)
     |████████████████████████████████| 25.9MB 146kB/s
Collecting lightgbm==2.3.1
  Downloading https://files.pythonhosted.org/packages/0b/9d/ddcb2f43aca194987f1a99e27edf41cf9bc39ea750c3371c2a62698c509a/lightgbm-2.3.1-py2.py3-none-manylinux1_x86_64.whl (1.2MB)
     |████████████████████████████████| 1.2MB 36.7MB/s
Collecting pytest==5.4.3
  Downloading https://files.pythonhosted.org/packages/9f/f3/0a83558da436a081344aa6c8b85ea5b5f05071214106036ce341b7769b0b/pytest-5.4.3-py3-none-any.whl (248kB)
     |████████████████████████████████| 256kB 51.9MB/s
Collecting numpy==1.19.1
  Downloading https://files.pythonhosted.org/packages/50/8f/29d5688614f9bba59931683d5d353d738d4a3007833219ee19c455732753/numpy-1.19.1-cp37-cp37m-manylinux2010_x86_64.whl (14.5MB)
     |████████████████████████████████| 14.5MB 19.9MB/s
Requirement already satisfied: python-dateutil>=2.6.1 in /usr/local/lib/python3.7/dist-packages (from pandas==1.0.5->lazypredict) (2.8.1)
Requirement already satisfied: pytz>=2017.2 in /usr/local/lib/python3.7/dist-packages (from pandas==1.0.5->lazypredict) (2018.9)
Collecting threadpoolctl>=2.0.0
  Downloading https://files.pythonhosted.org/packages/c6/e8/c216b9b60cbba4642d3ca1bae7a53daa0c24426f662e0e3ce3dc7f6caeaa/threadpoolctl-2.2.0-py3-none-any.whl
Requirement already satisfied: py>=1.5.0 in /usr/local/lib/python3.7/dist-packages (from pytest==5.4.3->lazypredict) (1.10.0)
Collecting pluggy<1.0,>=0.12
  Downloading https://files.pythonhosted.org/packages/a0/28/85c7aa31b80d150b772fbe4a229487bc6644da9ccb7e427dd8cc60cb8a62/pluggy-0.13.1-py2.py3-none-any.whl
Requirement already satisfied: more-itertools>=4.0.0 in /usr/local/lib/python3.7/dist-packages (from pytest==5.4.3->lazypredict) (8.8.0)
Requirement already satisfied: packaging in /usr/local/lib/python3.7/dist-packages (from pytest==5.4.3->lazypredict) (20.9)
Requirement already satisfied: attrs>=17.4.0 in /usr/local/lib/python3.7/dist-packages (from pytest==5.4.3->lazypredict) (21.2.0)
Requirement already satisfied: importlib-metadata>=0.12; python_version < "3.8" in /usr/local/lib/python3.7/dist-packages (from pytest==5.4.3->lazypredict) (4.6.0)
Requirement already satisfied: wcwidth in /usr/local/lib/python3.7/dist-packages (from pytest==5.4.3->lazypredict) (0.2.5)
Requirement already satisfied: pyparsing>=2.0.2 in /usr/local/lib/python3.7/dist-packages (from packaging->pytest==5.4.3->lazypredict) (2.4.7)
```

```
from lazypredict.Supervised import LazyClassifier
```

```
/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:143: FutureWarning: The sklearn.utils.testing module is  deprecated in version 0.22 and will be removed in version 0.24. The correspondin
  warnings.warn(message, FutureWarning)
```

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
diabetes = pd.read_csv('/content/diabetes2.csv')
feature_names = ['Pregnancies', 'Glucose', 'BloodPressure', 'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age']
diabetes_mod = diabetes[(diabetes.BloodPressure != 0) & (diabetes.BMI != 0) & (diabetes.Glucose != 0)]
X = diabetes_mod[feature_names]
y = diabetes_mod.Outcome
X_train, X_test, y_train, y_test = train_test_split(X, y, stratify = diabetes_mod.Outcome, random_state=0)
# fit all models
clf = LazyClassifier(predictions=True)
models, predictions = clf.fit(X_train, X_test, y_train, y_test)
print(models, predictions)
```

```
100%|██████████| 29/29 [00:01<00:00, 19.71it/s]                              Accuracy  ...  Time Taken
Model                                     ...
LinearDiscriminantAnalysis        0.79  ...        0.03
LGBMClassifier                    0.80  ...        0.09
NearestCentroid                   0.75  ...        0.02
LogisticRegression                0.78  ...        0.03
RidgeClassifier                   0.78  ...        0.02
LinearSVC                         0.78  ...        0.06
AdaBoostClassifier                0.78  ...        0.12
ExtraTreesClassifier              0.79  ...        0.20
KNeighborsClassifier              0.78  ...        0.02
RidgeClassifierCV                 0.78  ...        0.02
XGBClassifier                     0.77  ...        0.11
SVC                               0.77  ...        0.03
Perceptron                        0.76  ...        0.02
NuSVC                             0.77  ...        0.03
CalibratedClassifierCV            0.77  ...        0.13
RandomForestClassifier            0.77  ...        0.23
SGDClassifier                     0.76  ...        0.02
BernoulliNB                       0.75  ...        0.01
LabelPropagation                  0.75  ...        0.04
GaussianNB                        0.73  ...        0.02
LabelSpreading                    0.75  ...        0.05
DecisionTreeClassifier            0.73  ...        0.02
QuadraticDiscriminantAnalysis     0.72  ...        0.01
BaggingClassifier                 0.73  ...        0.05
PassiveAggressiveClassifier       0.70  ...        0.02
ExtraTreeClassifier               0.66  ...        0.02
DummyClassifier                   0.51  ...        0.02

[27 rows x 5 columns]      AdaBoostClassifier  BaggingClassifier  ...  XGBClassifier  LGBMClassifier
0                    0                  0  ...              0               0
1                    0                  0  ...              0               0
2                    0                  0  ...              0               0
3                    1                  1  ...              1               1
4                    0                  1  ...              1               0
..                 ...                ...  ...            ...             ...
176                  1                  1  ...              1               1
177                  0                  0  ...              0               0
178                  0                  1  ...              0               0
179                  0                  0  ...              0               0
180                  0                  0  ...              0               0

[181 rows x 27 columns]
```

✓ 1s completed at 11:17 AM