Support Vector Machines

# Loading libraries and in-built datasets

import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
from sklearn.datasets import load\_breast\_cancer  
  
cancer = load\_breast\_cancer()  
print(cancer.keys())  
print(cancer["DESCR"])

# Putting data into a data frame

df\_feat = pd.DataFrame(cancer["data"],columns=cancer["feature\_names"])  
print(df\_feat.info())

# Do visualisation on your own

# on you own

# Train test split and training classifier

X = df\_feat  
y = cancer["target"]  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=101)

# Fitting test data to model, making predictions and checking metrics  
model = SVC()  
model.fit(X\_train,y\_train)  
predictions = model.predict(X\_test)  
print(classification\_report(y\_test,predictions))  
print(confusion\_matrix(y\_test,predictions))

# Picking best parameters using grid search

# Checking best parameters using grid search  
param\_grid = {"C":[0.1,1,10,100,1000],"gamma":[1,0.1,0.01,0.001,0.0001]}  
grid = GridSearchCV(SVC(),param\_grid,verbose=5)  
grid.fit(X\_train,y\_train)  
print(grid.best\_params\_)  
print(grid.best\_estimator\_)

# Checking metrics again to ascertain accuracy

# Checking metrics again after grid search  
grid\_predictions = grid.predict(X\_test)  
print(classification\_report(y\_test,grid\_predictions))  
print(confusion\_matrix(y\_test,grid\_predictions))