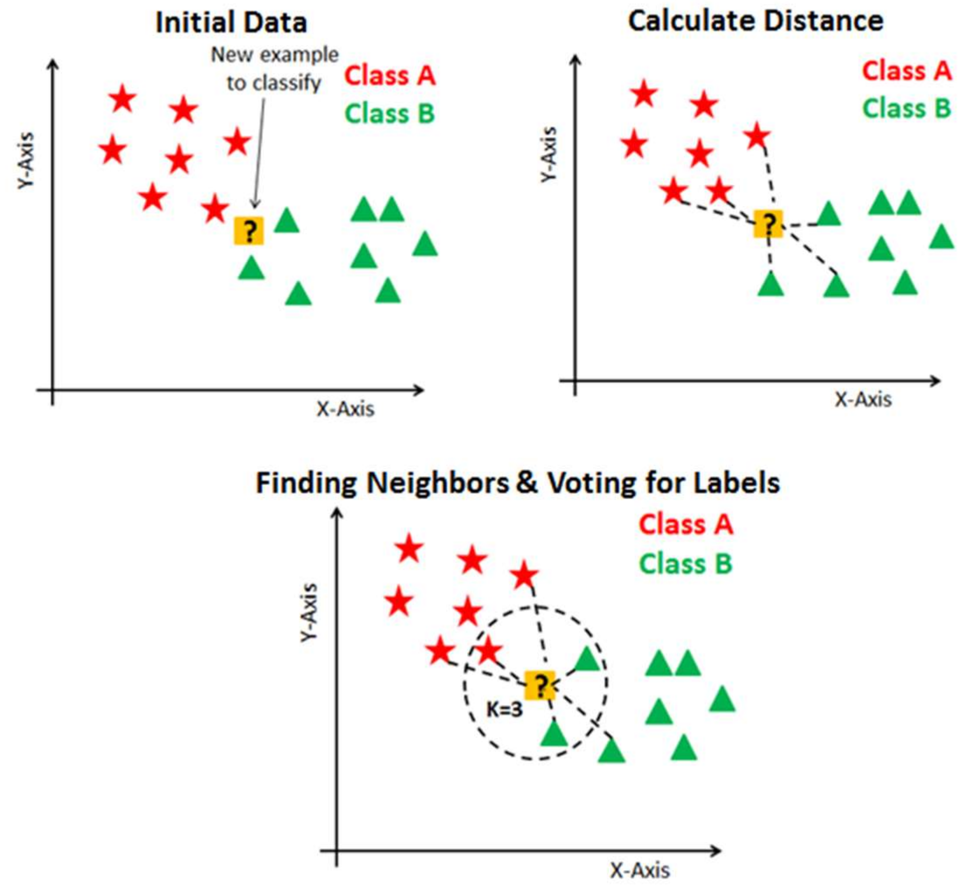
LISBON SCHOOL OF ECONOMICS & MANAGEMENT
UNIVERSIDADE DE LISBOA

Ano Letivo 2018/2019

CLASSIFICATION ALGORITHMS

Classification

1. KNN (K-Nearest Neighbour):



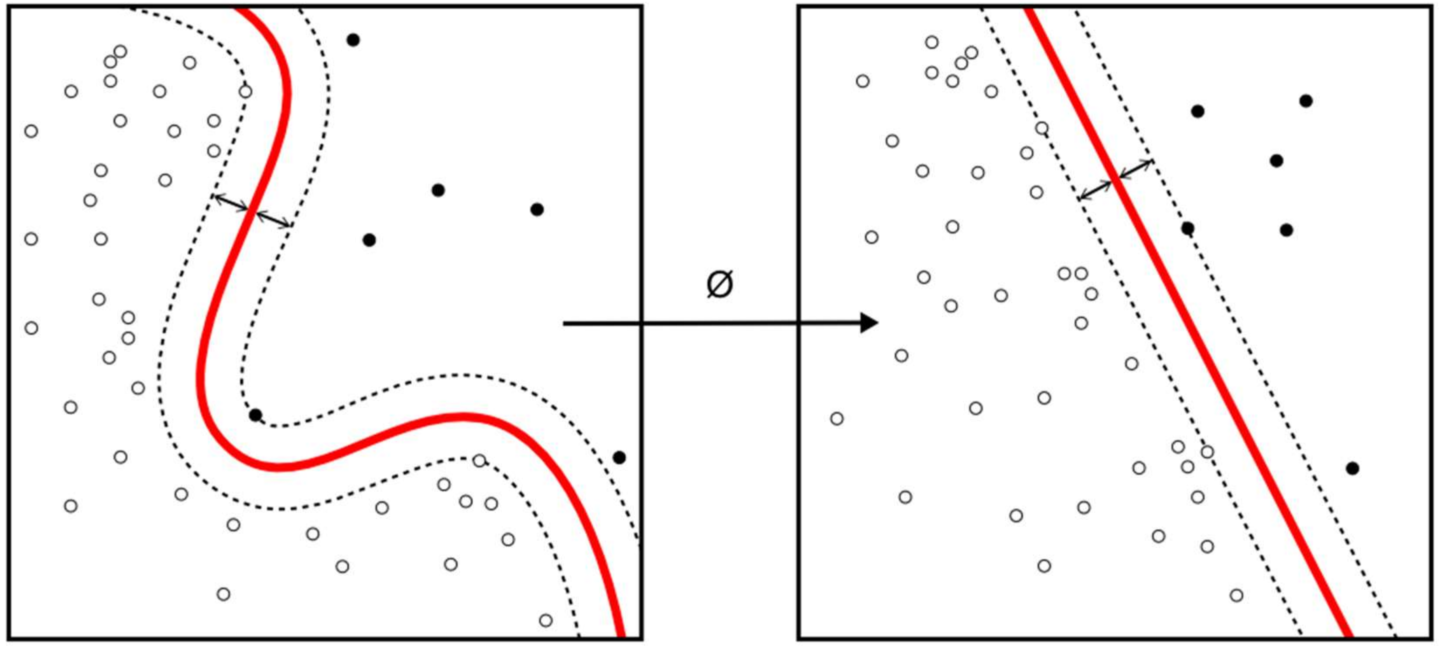
Classification

1. KNN (K-Nearest Neighbour):

```
from sklearn.preprocessing import StandardScaler
standardizer=StandardScaler()
X=standardizer.fit_transform(Xfeatures)

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y,
test_size=0.3)

from sklearn.neighbors import KNeighborsClassifier
model = KNeighborsClassifier()
result=model.fit(X_train,y_train)
y_pred = model.predict(X_test)

from sklearn.metrics import confusion_matrix
a=confusion_matrix(y_test, y_pred)
import seaborn as sn
sn.heatmap(a, annot=True)
```

Classification

2. SVM (support vector machine)



Classification

2. SVM (support vector machine):

```
from sklearn.preprocessing import StandardScaler
standardizer=StandardScaler()
X=standardizer.fit_transform(Xfeatures)

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y, 
test_size=0.3)

from sklearn.svm import SVC
model = SVC()
result=model.fit(X_train,y_train)
y_pred = model.predict(X_test)

from sklearn.metrics import confusion_matrix
a=confusion_matrix(y_test, y_pred)
import seaborn as sn
sn.heatmap(a, annot=True)
```

Classification

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}$$

$$X = (x_1, x_2, x_3, \ldots.., x_n)$$

3. **Navie Bayes**

Classification

3. Navie Bayes

```
from sklearn.preprocessing import StandardScaler
standardizer=StandardScaler()
X=standardizer.fit_transform(Xfeatures)

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, Y,
test_size=0.3)

from sklearn.naive_bayes import GaussianNB
model = GaussianNB()
result=model.fit(X_train,y_train)
y_pred = model.predict(X_test)

from sklearn.metrics import confusion_matrix
a=confusion_matrix(y_test, y_pred)
import seaborn as sn
sn.heatmap(a, annot=True)
```