LISBON
SCHOOL OF
ECONOMICS &
MANAGEMENT
UNIVERSIDADE DE LISBOA

Carlos J. Costa

CLASSIFICATION

(2021)

Summary

- Classification
- K -Near Neighbour (KNN)
- Support Vector Machines (SVM)
- Naive Bayes
- Logistic Regression
- Decision Trees

Classification

Classification

age	address	income	ed	employ	equip	callcard	wireless	churn
33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No

age	address	income	ed	employ	equip	callcard	wireless	churn
35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	?

Classification

Age	Sex	BP	Cholesterol	Na	K	Drug
23	F	HIGH	HIGH	0.793	0.031	drugY
47	M	LOW	HIGH	0.739	0.056	drugC
47	M	LOW	HIGH	0.697	0.069	drugC
28	F	NORMAL	HIGH	0.564	0.072	drugX
61	F	LOW	HIGH	0.559	0.031	drugY
22	F	NORMAL	HIGH	0.677	0.079	drugX
49	F	NORMAL	HIGH	0.79	0.049	drugY
41	M	LOW	HIGH	0.767	0.069	drugC
60	M	NORMAL	HIGH	0.777	0.051	drugY
43	M	LOW	NORMAL	0.526	0.027	drugY

Categorical Variable

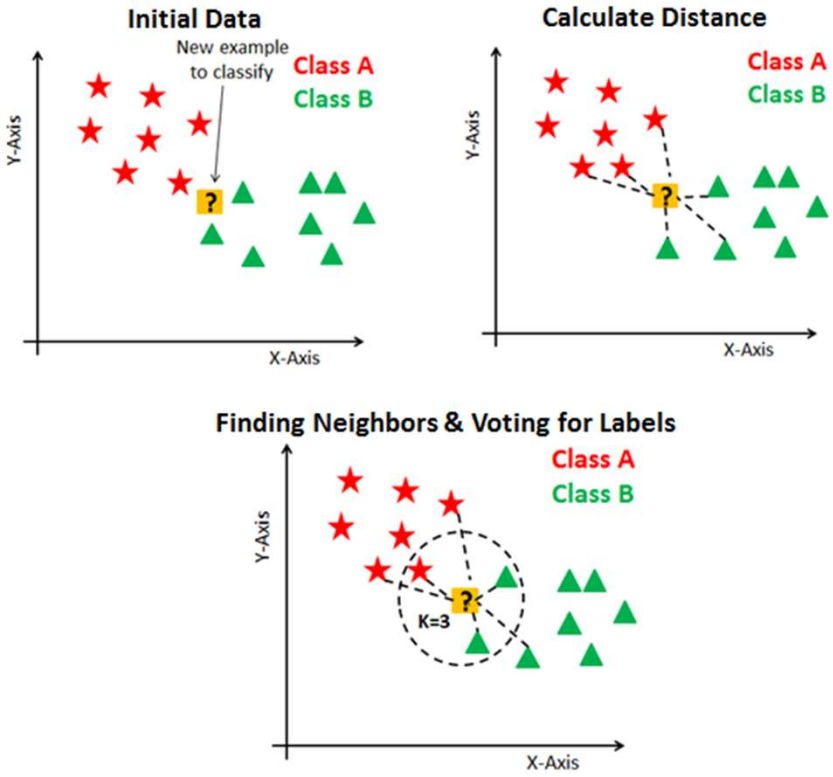
Age	Sex	BP	Cholesterol	Na	K	Drug
36	F	LOW	HIGH	0.697	0.069	

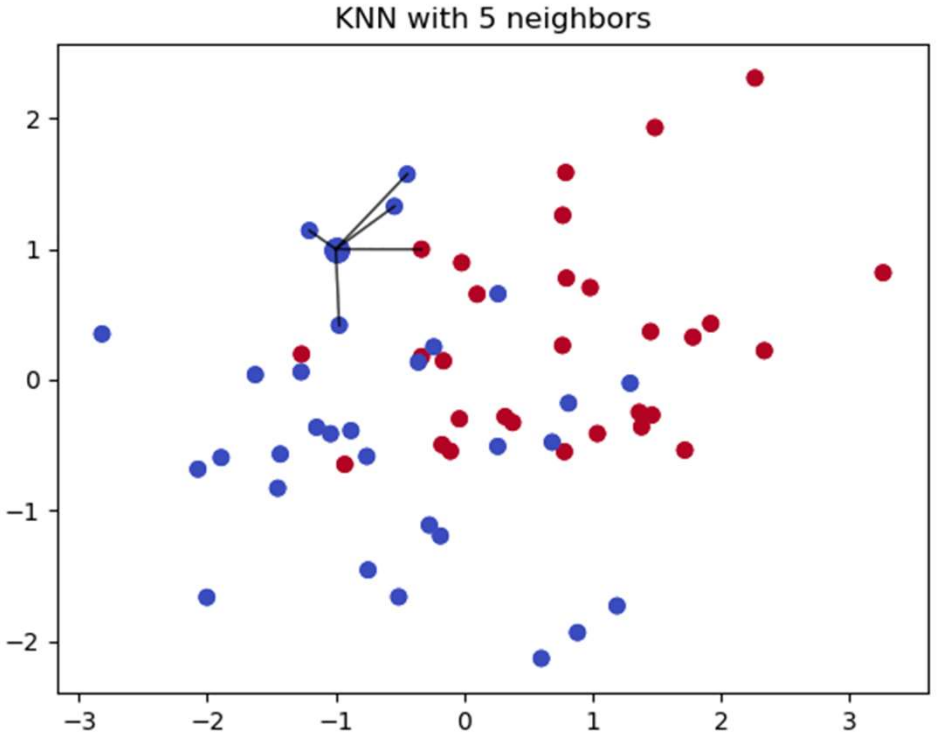
Classification

Classification algorithms in machine learning:

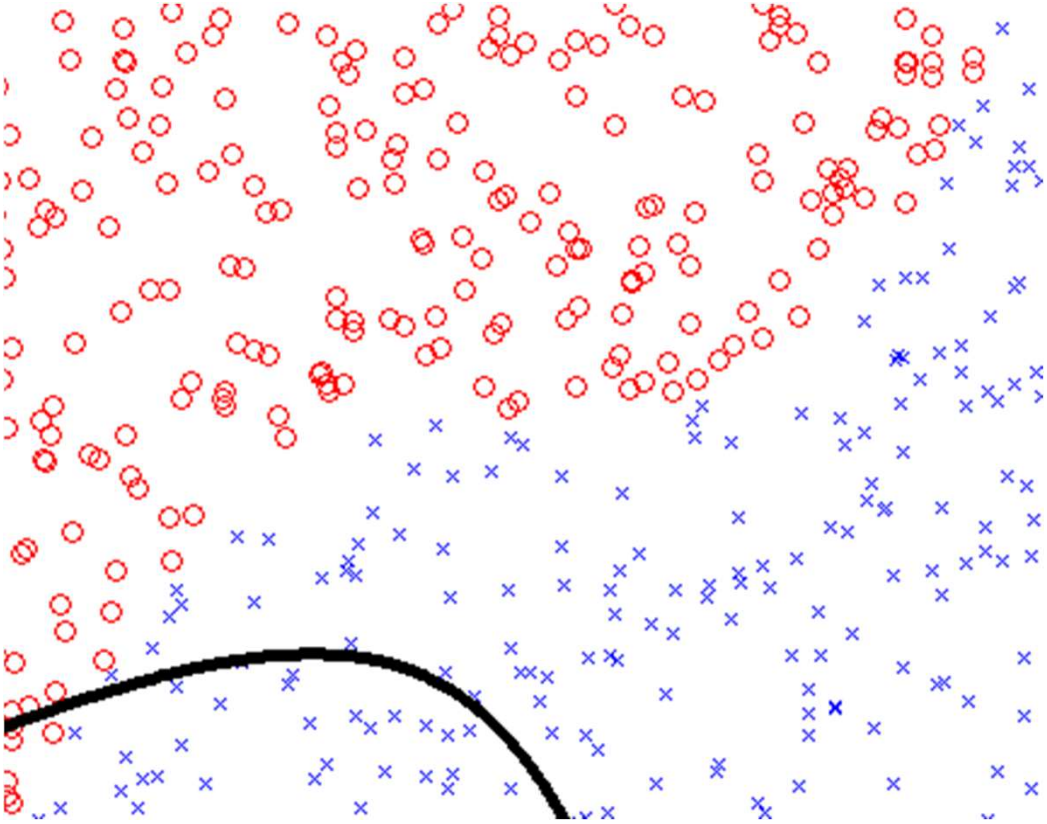
- K -Near Neighbour (KNN)
- Support Vector Machines (SVM)
- Naive Bayes
- Logistic Regression
- Decision Trees
- Linear Discriminate Analysis
- Neural Networks

KNN

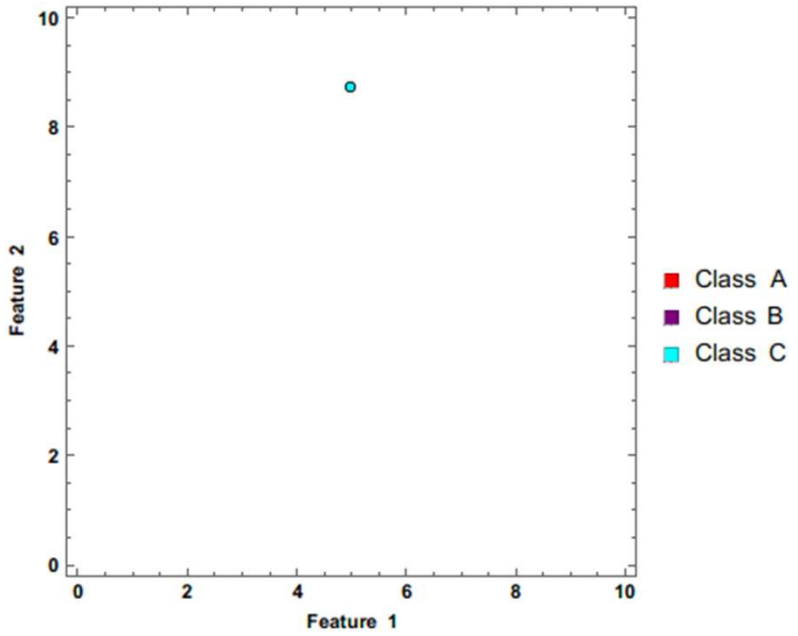

Initial Data
New example
to classify
Class A
Class B
Y-Axis
X-Axis
?
Calculate Distance
Class A
Class B
Y-Axis
X-Axis
?
Finding Neighbors & Voting for Labels
Class A
Class B
Y-Axis
X-Axis
K=3
?


KNN with 5 neighbors
2
1
0
−1
−2
−3
−2
−1
0
1
2
3

SVM (support vector machine)



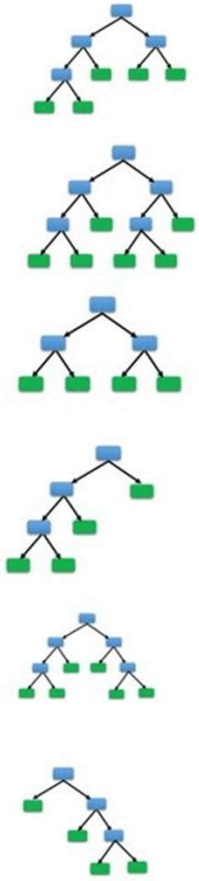
Naive Bayes



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

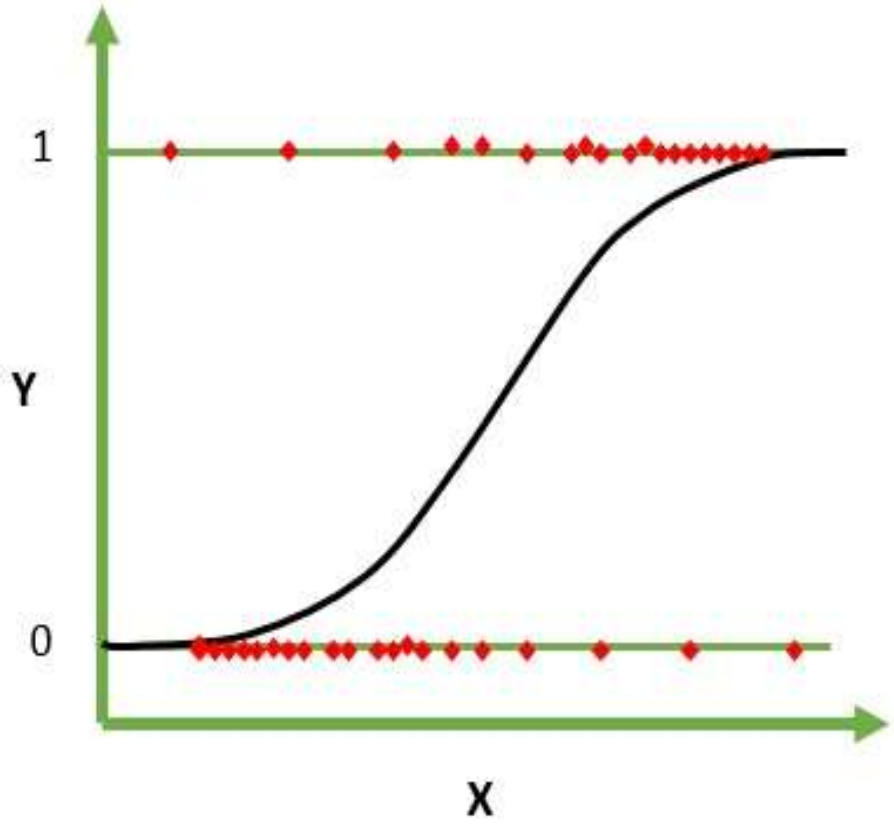
$$X = (x_1, x_2, x_3, ..., x_n)$$

Random Forest



Random Forest in Action!!!

Logistics Regression

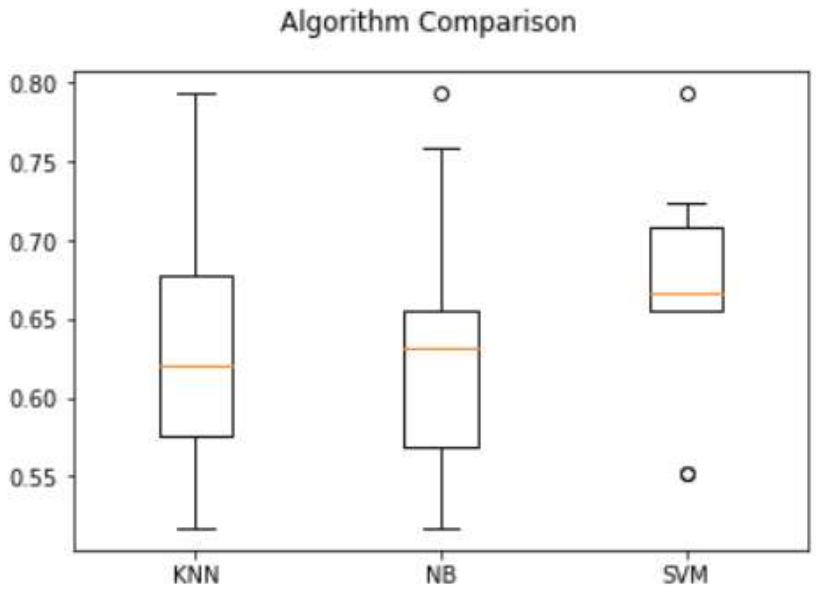


```
from sklearn.preprocessing import StandardScaler
standardizer=StandardScaler()
X=standardizer.fit_transform(Xfeatures)
```

```
from sklearn import model_selection
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC

models = []
models.append(('KNN', KNeighborsClassifier()))
models.append(('NB', GaussianNB()))
models.append(('SVM', SVC()))

results = []
names = []
scoring = 'accuracy'

seed = 7

for name, model in models:
    #, random_state=seed
    kfold = model_selection.KFold(n_splits=10)
    cv_results = model_selection.cross_val_score(model, X, Y, cv=kfold, scoring=scoring)
    results.append(cv_results)
    names.append(name)
    msg = "%s: %f (%f)" % (name, cv_results.mean(), cv_results.std())
    print(msg)
```

```
KNN: 0.635222 (0.084238)
NB: 0.635099 (0.084984)
SVM: 0.666872 (0.070033)
```

```
import matplotlib.pyplot as plt

fig = plt.figure()
fig.suptitle('Algorithm Comparison')
ax = fig.add_subplot(111)
plt.boxplot(results)
ax.set_xticklabels(names)
plt.show()
```



Conclusions

- Classification
- K -Near Neighbour (KNN)
- Support Vector Machines (SVM)
- Naive Bayes
- Logistic Regression
- Decision Trees

References

- Albon, Ch. (2018) *Machine Learning with Python Cookbook*. O'Reilly
- Domingos, P. (2015) *The Master Algorithm*, Penguin Books
- Hinton, J.; Sejnowski, T.(1999). *Unsupervised Learning: Foundations of Neural Computation*. MIT Press
- Morgan; P. (2019) *Data Science from Scratch with Python*, AI Science
- Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective* (1 edition). Cambridge, MA: The MIT Press.
- Otte, E.; Rousseau, R. (2002). "Social network analysis: a powerful strategy, also for the information sciences". *Journal of Information Science*. 28 (6): 441–453. doi:10.1177/016555150202800601.
- Stuart J. R., Norvig, P. (2010) *Artificial Intelligence: A Modern Approach*, Third Edition, Prentice Hall ISBN 9780136042594.