Support Vector Machines

Data Science in Context

September 18, 2019

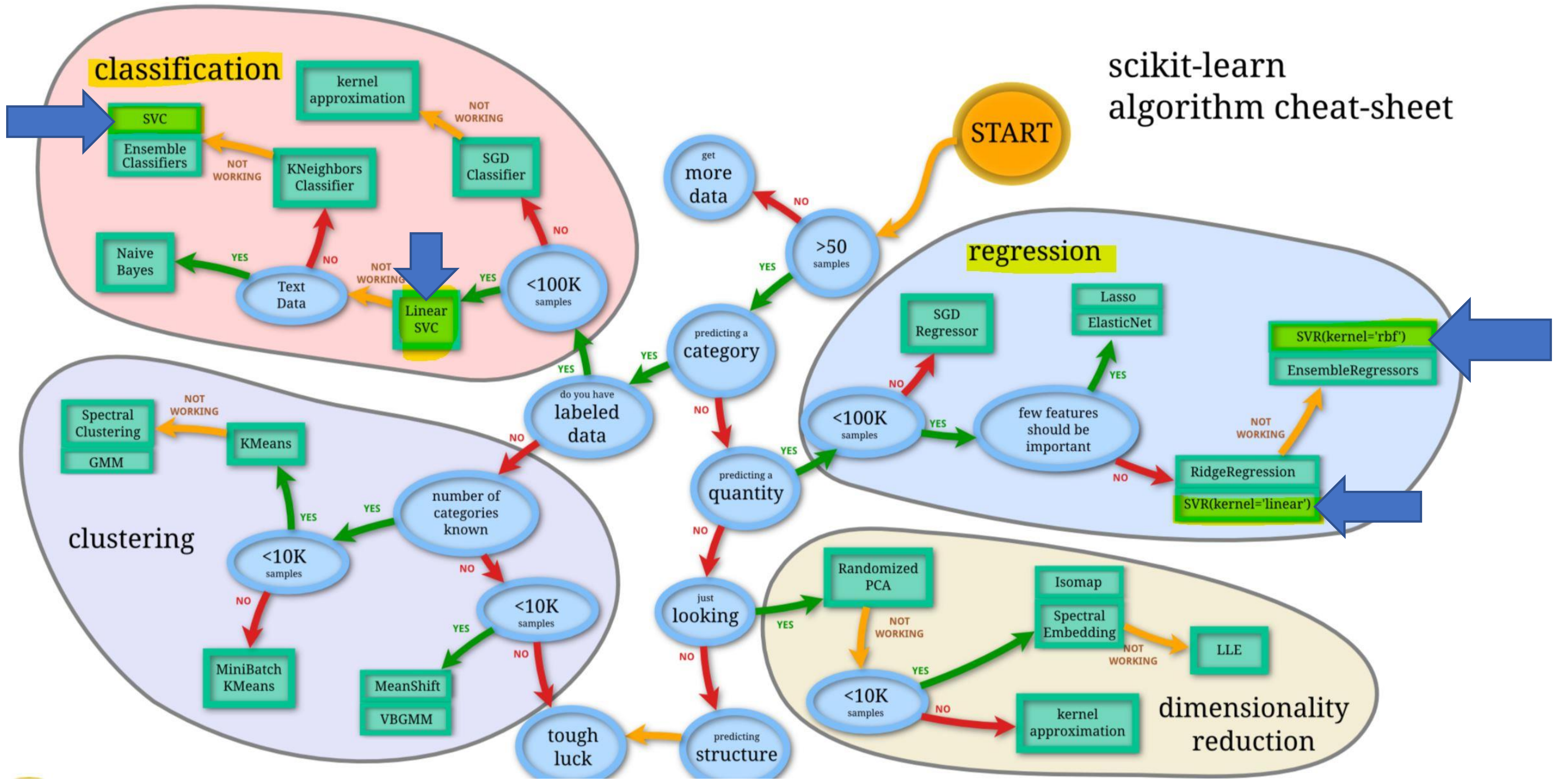
Michael Y.

Last week we looked into Unsupervised Learning.

Today I would like to look at a specific **Supervised Learning** technique which is discussed in the current reading (Chapter 4 of Data Science for Business):

Support Vector Machines (SVM)

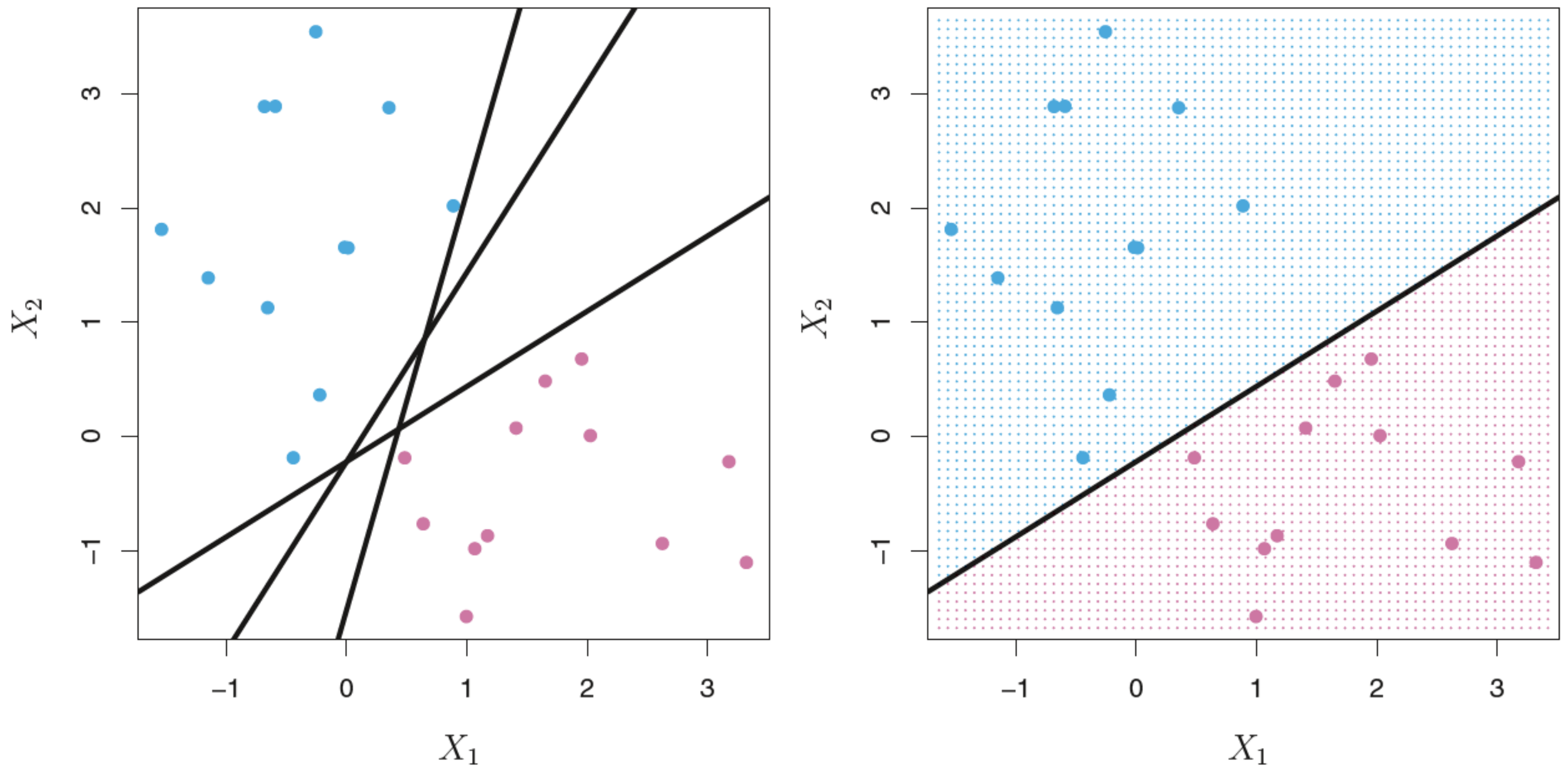
Support Vector Machines are used for both Classification (SVC) and for Regression (SVR):



Source: scikit-learn, a key python library implementing numerous machine-learning algorithms
https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

A Support Vector Machine performs classification by splitting the data set using "Hyperplanes".
Below are three possible separations on the dataset:

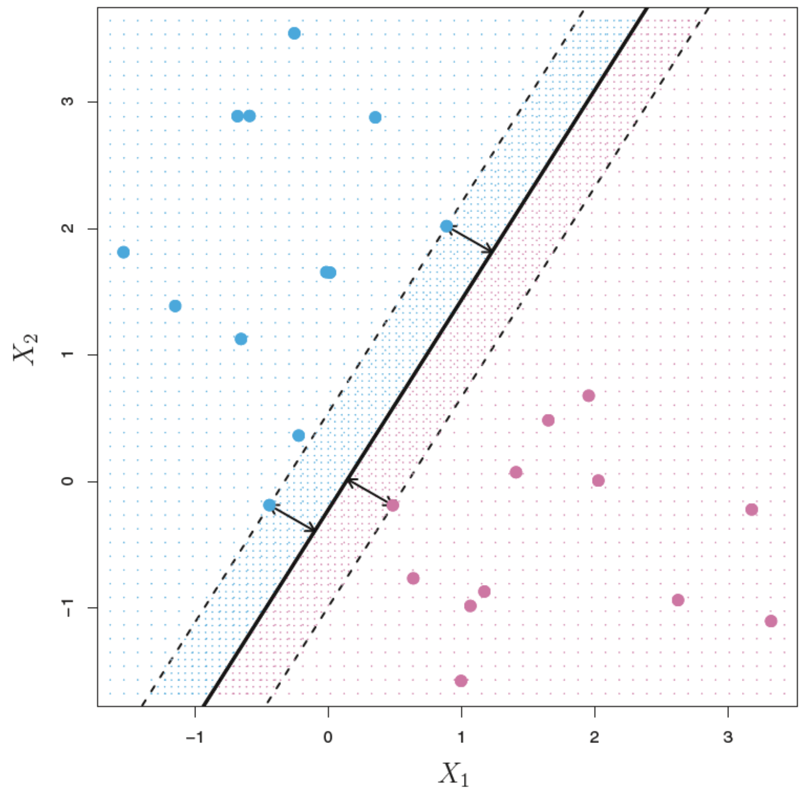
340 9. Support Vector Machines



Source: James, Witten, Hastie, Tibshirani: Introduction to Statistical Learning, p.340

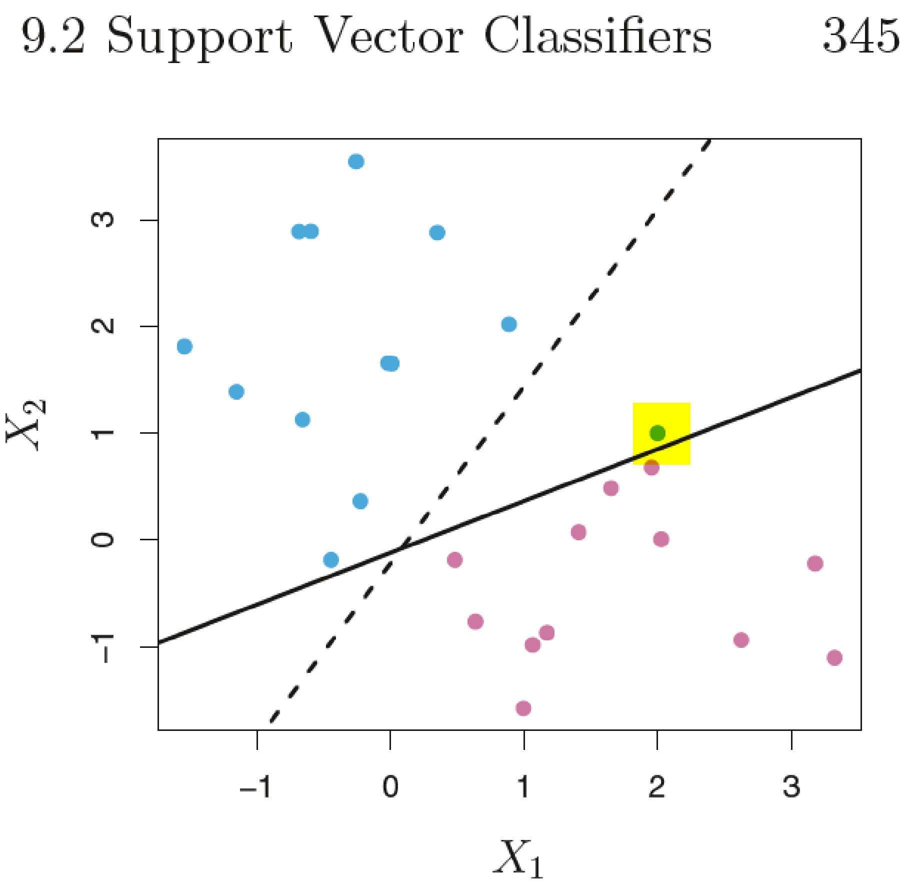
One method used by Support Vector Machines is to seek the "**Maximal Margin Classifier**", which finds the largest separation ("Margin") between the classes:

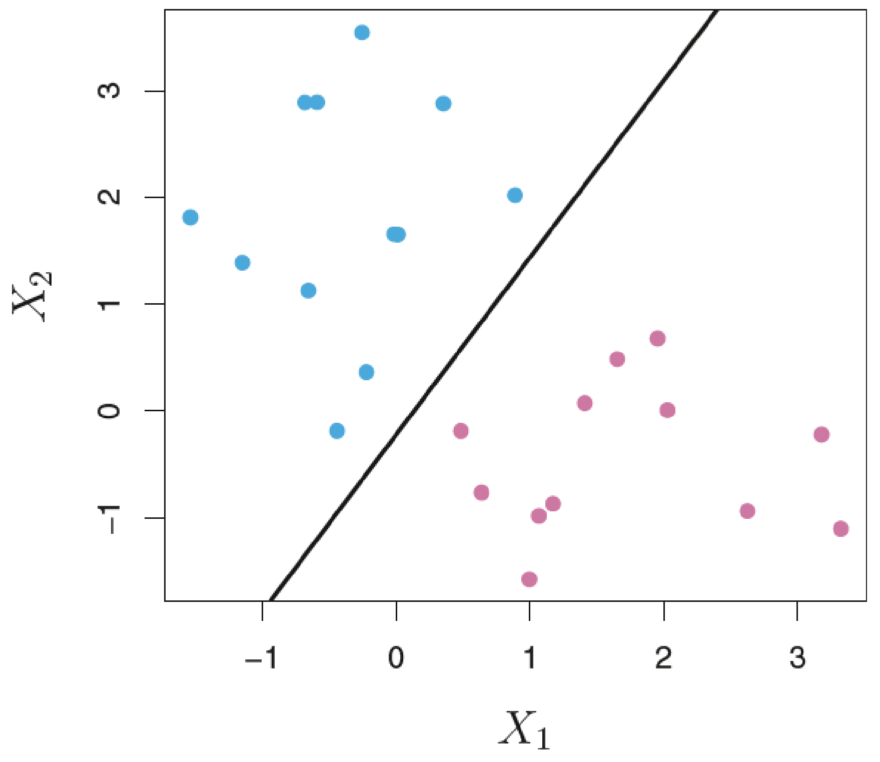
342 9. Support Vector Machines



Source: James, Witten, Hastie, Tibshirani: Introduction to Statistical Learning, p.342

However, this method can is highly subject to outliers, as shown below.
One additional point (in the Blue category), highlighted in yellow, is added to the picture, in a location close to the Red points.
The Maximal Margin Classifier undergoes a dramatic shift:

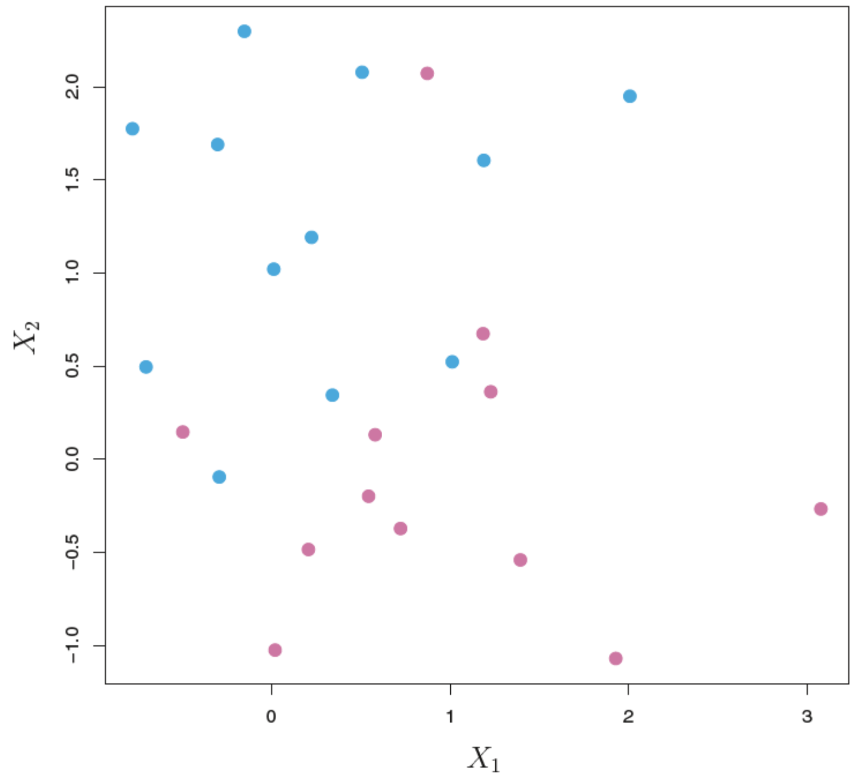

9.2 Support Vector Classifiers 345



Source: James, Witten, Hastie, Tibshirani: Introduction to Statistical Learning, p.345

However, it is **not always possible** to find such a pure separation, because the data may not be separable, such as in the example below:

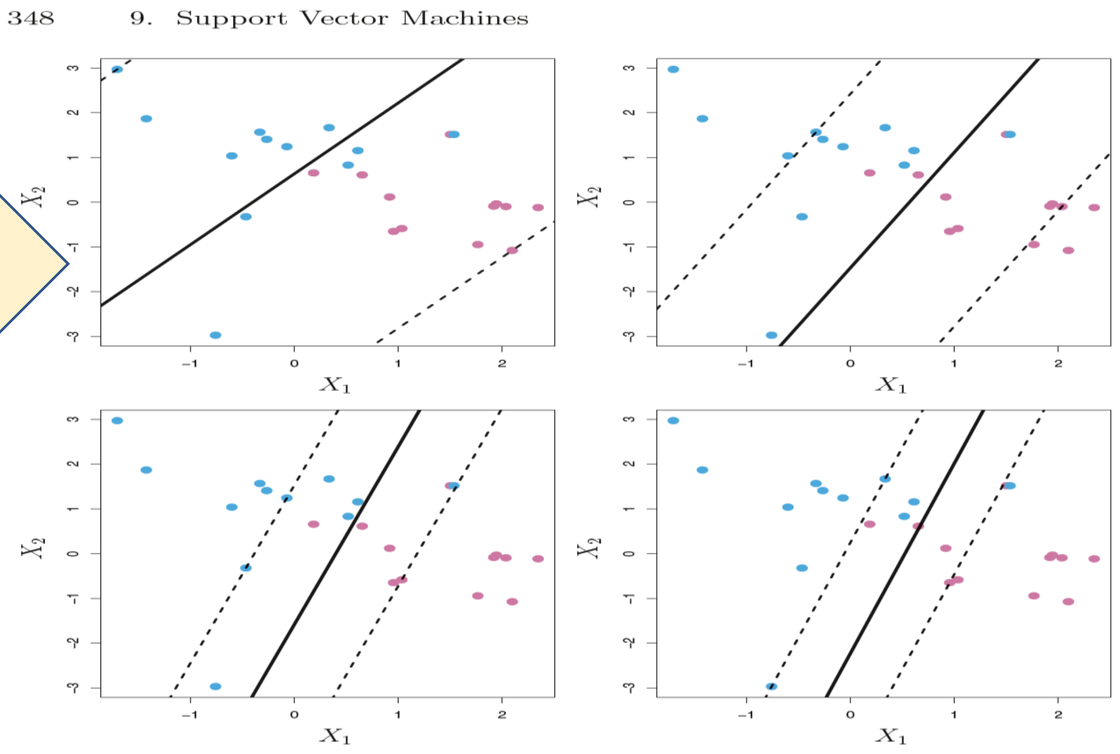
344 9. Support Vector Machines



Source: James, Witten, Hastie, Tibshirani: Introduction to Statistical Learning, p.344

The distance by which observations fall on the incorrect side is measured as an “**error**.” An observation which falls on the **correct side of the hyperplane**, but within the margin, is assigned an error between 0 (on the margin) and 1 (on the hyperplane.) An observation which falls on the **wrong side of the hyperplane** is assigned an error > 1. An **constrained optimization** is performed which seeks to **find the widest possible Margin** while **limiting the sum of “errors”** to be less than a specified “tuning parameter.”

Large tuning parameter ➔ wide margin, many breaches, high bias, low variance



Small tuning parameter ➔ narrow margin, fewer breaches, low bias, high variance

Source: James, Witten, Hastie, Tibshirani: Introduction to Statistical Learning, p.348

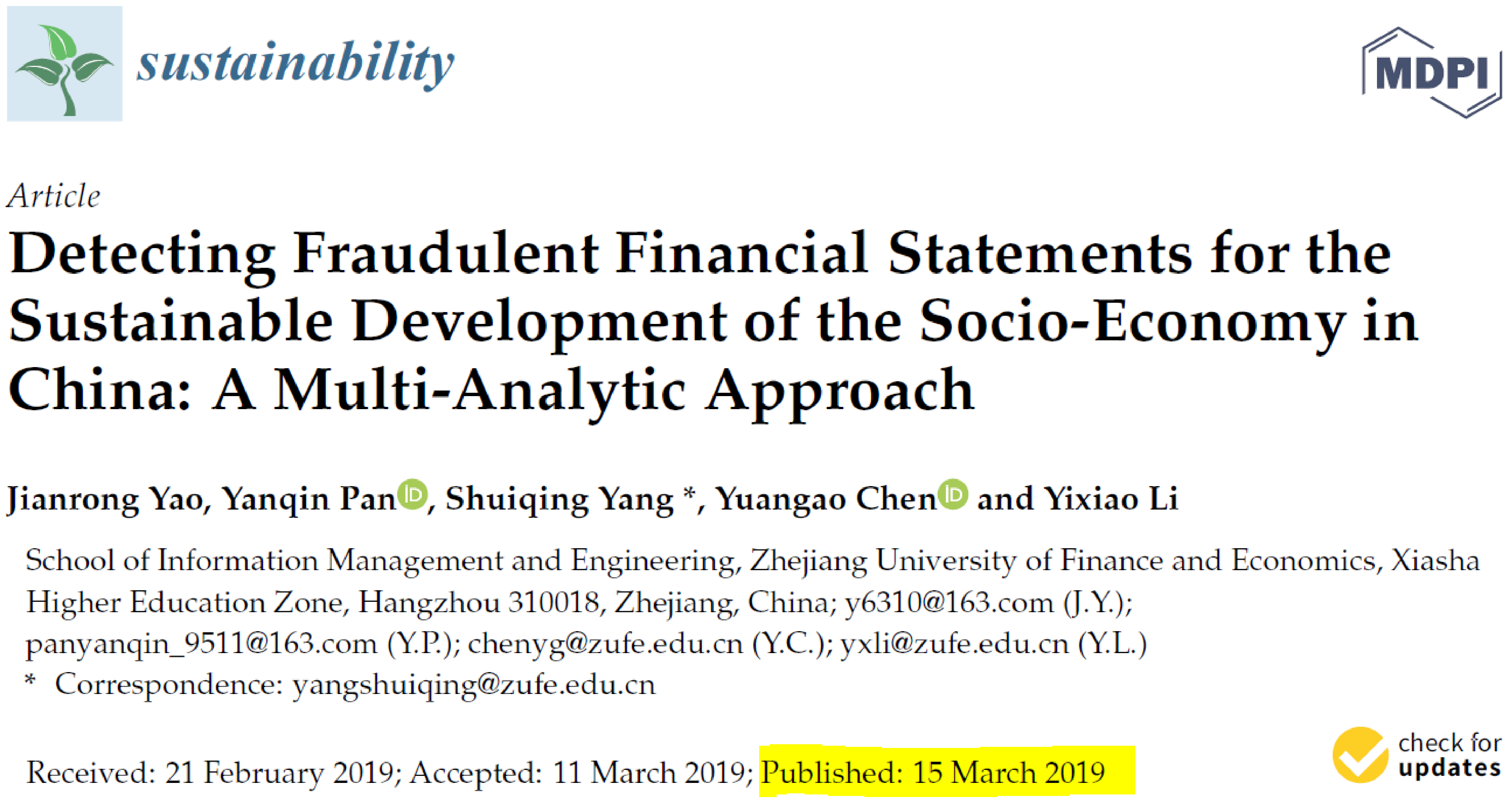
Results of a case study:

Comparison of various ML techniques for Detection of Fraudulent Financial Statements among Chinese companies listed in Shanghai and Shenzhen

Yao, Pan, Yang, Chen and Li, “Detecting Fraudulent Financial Statements for the Sustainable Development of the Socio-Economy in China: A Multi-Analytic Approach”, in Sustainability **2019**, 11, 1579;

https://www.mdpi.com/journal/sustainability ;

https://doi.org/10.3390/su11061579



sustainability

MDPI

Article

Detecting Fraudulent Financial Statements for the Sustainable Development of the Socio-Economy in China: A Multi-Analytic Approach

Jianrong Yao, Yanqin Pan, Shuiqing Yang *, Yuangao Chen and Yixiao Li

School of Information Management and Engineering, Zhejiang University of Finance and Economics, Xiasha Higher Education Zone, Hangzhou 310018, Zhejiang, China; y6310@163.com (J.Y.); panyanqin_9511@163.com (Y.P.); chenyg@zufe.edu.cn (Y.C.); yxli@zufe.edu.cn (Y.L.)

* Correspondence: yangshuiqing@zufe.edu.cn

Received: 21 February 2019; Accepted: 11 March 2019; Published: 15 March 2019

check for updates

Earlier this year, a team of Chinese researchers published a paper examining **detectability of financial statement fraud** using various machine-learning classification techniques, performed on:

- a full set of 24 features

and on **dimensionally-reduced** variables, selected via:

- stepwise regression (13 features) and
- principal components (16 components.)

Number of firms analyzed – matched samples

- A total of n=**134** instances of **fraudulent financial reporting** between 2008 and 2017 were identified among companies listed on the Shanghai and Shenzhen Stock Exchanges
- To obtain "matched samples", for each of the above companies the researchers identified 3 similar firms which were not accused of fraudulent reporting, where each firm was similar in asset size and industry classification during the same year as the fraudulent reporting (m=**402**)
- Thus, the total sample size was m+n = 134+402 = **536**
- The data is randomly split between training and testing, with the results of the classification or misclassification of the test data tabulated and analyzed as shown below.

Yao, Pan, Yang, Chen and Li, "Detecting Fraudulent Financial Statements for the Sustainable Development of the Socio-Economy in China: A Multi-Analytic Approach", in Sustainability **2019**, 11, 1579; https://www.mdpi.com/journal/sustainability ; https://doi.org/10.3390/su11061579

The diagram below illustrates the experimental setting, where the data was analyzed on all 24 variables, and again following two separate dimensionality reduction techniques, using 6 different classifiers:

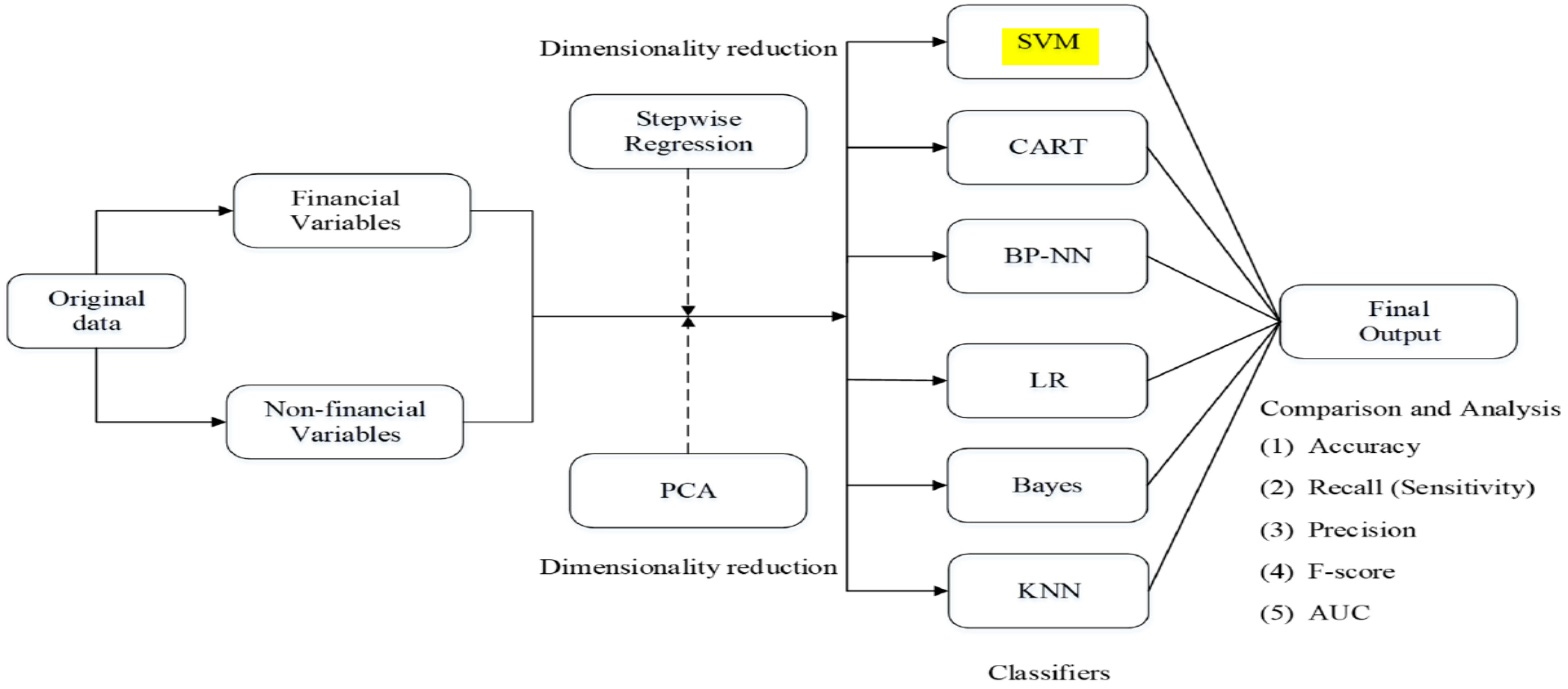


Figure 3. Experimental setting.

Results in the confusion matrix:

(**TP**) true positive (a **non-fraud** firm **correctly** classified as a **non-fraud** firm);
(**FN**) false negative (a **non-fraud** firm **incorrectly** classified as a **fraud** firm);
(**TN**) true negative (a **fraud** firm **correctly** classified as a **fraud** firm);
(**FP**) false positive (a **fraud** firm **incorrectly** classified as a **non-fraud** firm).

The quality of the results were evaluated using these standard metrics:

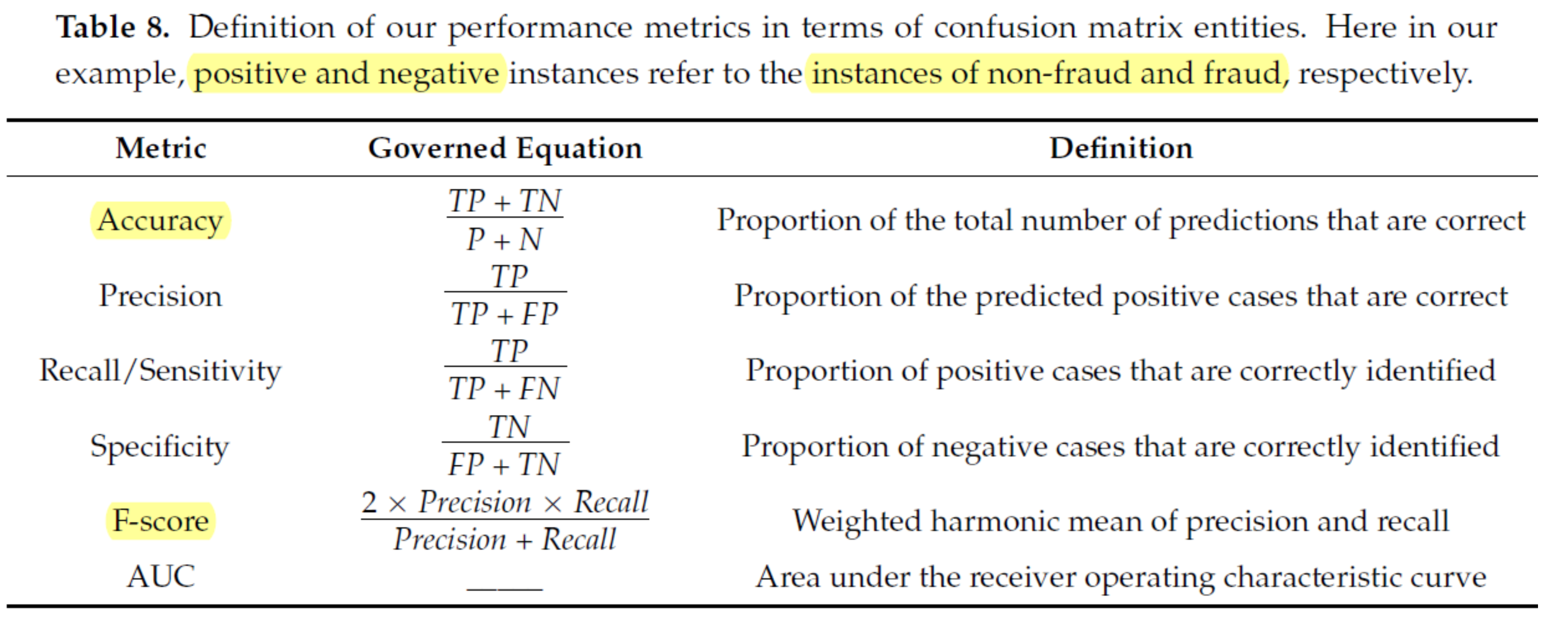


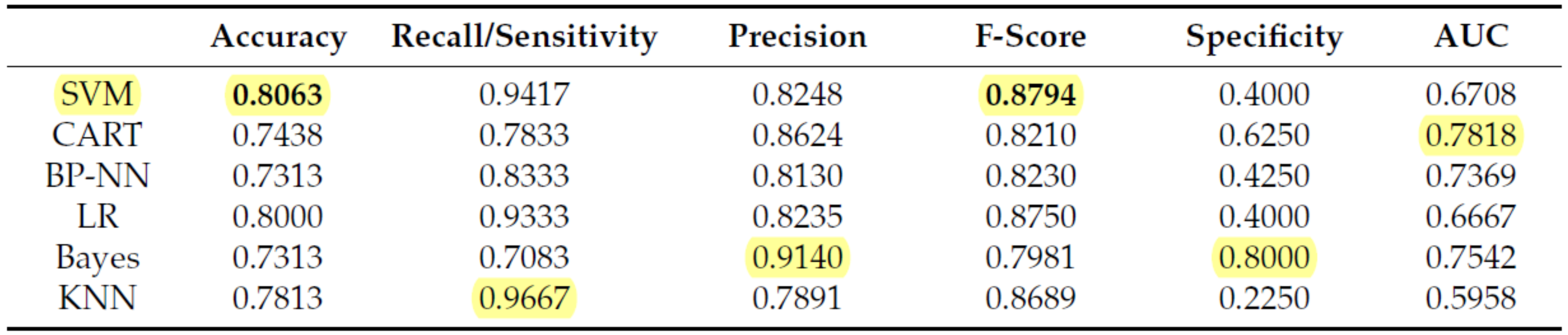
Table 8. Definition of our performance metrics in terms of confusion matrix entities. Here in our example, positive and negative instances refer to the instances of non-fraud and fraud, respectively.

Metric	Governed Equation	Definition
Accuracy	$\frac{TP + TN}{P + N}$	Proportion of the total number of predictions that are correct
Precision	$\frac{TP}{TP + FP}$	Proportion of the predicted positive cases that are correct
Recall/Sensitivity	$\frac{TP}{TP + FN}$	Proportion of positive cases that are correctly identified
Specificity	$\frac{TN}{FP + TN}$	Proportion of negative cases that are correctly identified
F-score	$\frac{2 \times Precision \times Recall}{Precision + Recall}$	Weighted harmonic mean of precision and recall
AUC	____	Area under the receiver operating characteristic curve

Yao, Pan, Yang, Chen and Li, "Detecting Fraudulent Financial Statements for the Sustainable Development of the Socio-Economy in China: A Multi-Analytic Approach", in Sustainability **2019**, 11, 1579; https://www.mdpi.com/journal/sustainability ; https://doi.org/10.3390/su11061579

Comparing the various classifiers, **SVM performed best** on Accuracy and on F-score:

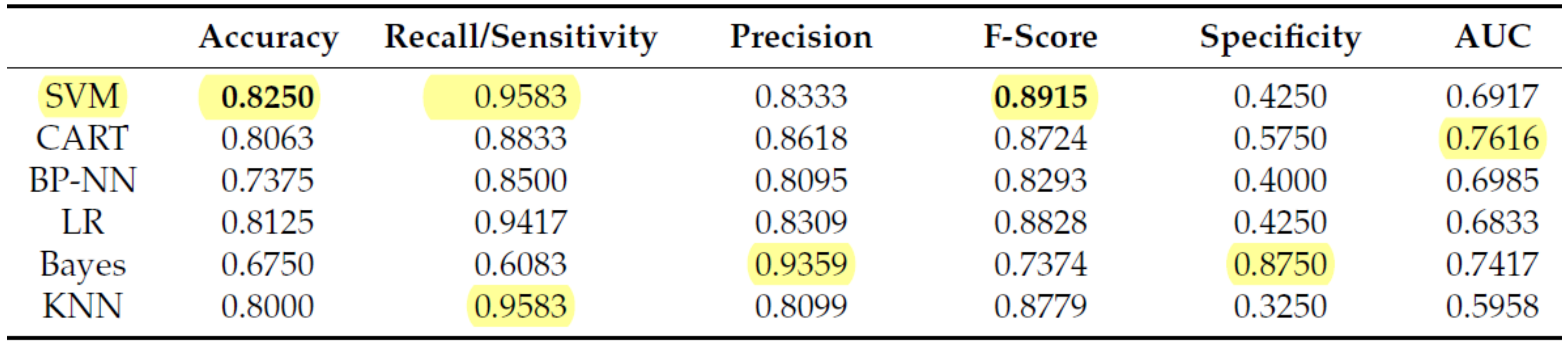
Table 9. Classification performance of the six classifiers using all variables.



	Accuracy	Recall/Sensitivity	Precision	F-Score	Specificity	AUC
SVM	**0.8063**	0.9417	0.8248	**0.8794**	0.4000	0.6708
CART	0.7438	0.7833	0.8624	0.8210	0.6250	0.7818
BP-NN	0.7313	0.8333	0.8130	0.8230	0.4250	0.7369
LR	0.8000	0.9333	0.8235	0.8750	0.4000	0.6667
Bayes	0.7313	0.7083	0.9140	0.7981	0.8000	0.7542
KNN	0.7813	0.9667	0.7891	0.8689	0.2250	0.5958

Reducing the number of features using **stepwise regression**, SVM still performed best on Accuracy and on F-score, and also improved its result on Recall/Sensitivity to match KNN (K-Nearest Neighbors):

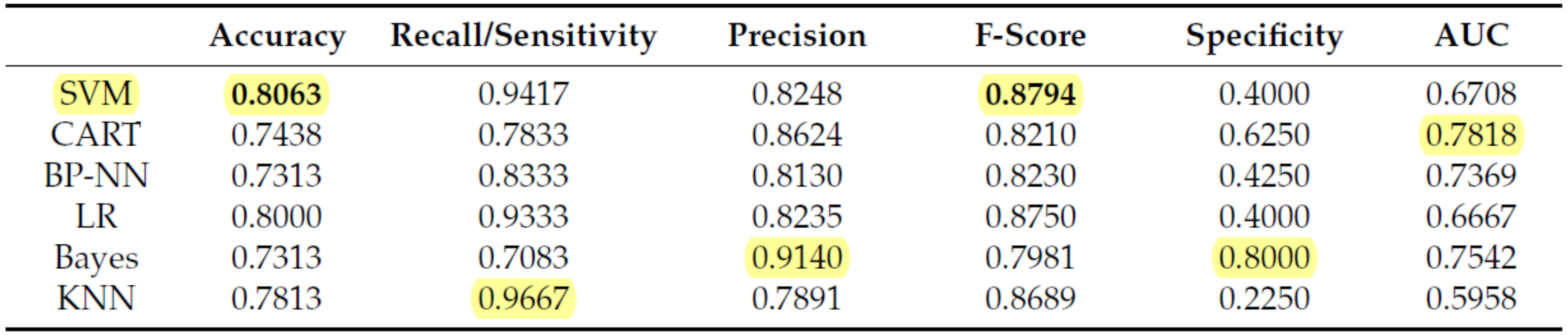
Table 10. Classification performance of the six classifiers using 13 significant original variables.



	Accuracy	Recall/Sensitivity	Precision	F-Score	Specificity	AUC
SVM	**0.8250**	0.9583	0.8333	**0.8915**	0.4250	0.6917
CART	0.8063	0.8833	0.8618	0.8724	0.5750	0.7616
BP-NN	0.7375	0.8500	0.8095	0.8293	0.4000	0.6985
LR	0.8125	0.9417	0.8309	0.8828	0.4250	0.6833
Bayes	0.6750	0.6083	0.9359	0.7374	0.8750	0.7417
KNN	0.8000	0.9583	0.8099	0.8779	0.3250	0.5958

Comparing the various classifiers, **SVM performed best** on **Accuracy** and on **F-score**:

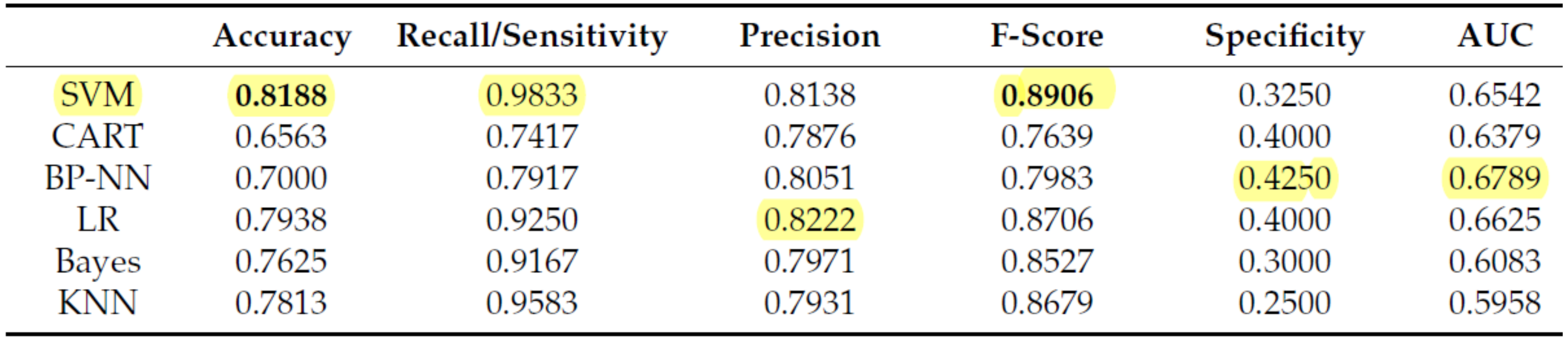
Table 9. Classification performance of the six classifiers using all variables.



	Accuracy	Recall/Sensitivity	Precision	F-Score	Specificity	AUC
SVM	**0.8063**	0.9417	0.8248	**0.8794**	0.4000	0.6708
CART	0.7438	0.7833	0.8624	0.8210	0.6250	0.7818
BP-NN	0.7313	0.8333	0.8130	0.8230	0.4250	0.7369
LR	0.8000	0.9333	0.8235	0.8750	0.4000	0.6667
Bayes	0.7313	0.7083	0.9140	0.7981	0.8000	0.7542
KNN	0.7813	0.9667	0.7891	0.8689	0.2250	0.5958

Reducing the number of features using **Principal Components (PCA)**, SVM still performed best on **Accuracy** and on **F-score**, and also improved its result on **Recall/Sensitivity** to take the lead:

Table 11. Classification performance of the six classifiers using 16 principal components.



	Accuracy	Recall/Sensitivity	Precision	F-Score	Specificity	AUC
SVM	**0.8188**	0.9833	0.8138	**0.8906**	0.3250	0.6542
CART	0.6563	0.7417	0.7876	0.7639	0.4000	0.6379
BP-NN	0.7000	0.7917	0.8051	0.7983	0.4250	0.6789
LR	0.7938	0.9250	0.8222	0.8706	0.4000	0.6625
Bayes	0.7625	0.9167	0.7971	0.8527	0.3000	0.6083
KNN	0.7813	0.9583	0.7931	0.8679	0.2500	0.5958

Conclusions from the authors of the study:

- **Part 1:** We found that the **SVM** classifier performs most outstandingly when there are 24 input variables and the accuracy is 0.8063.
- **Part 2:** The accuracy of the classification of financial statements can be further improved by using the 13 input variables selected by the **stepwise regression** method compared to the 24 input variables used. Moreover, the performance of the **SVM** classifier is the most outstanding.
- **Part 3:** After using the **PCA** method to transform the original 24 input variables, we selected the first 16 principal components based on previous studies. The accuracy of almost all classifiers decreased compared to classifiers that use 24 input variables.
- Overall, the experimental results based on accuracy and F-score indicated that **SVM is the top performer** and the classification performance is improved more by stepwise regression rather than PCA.

SVM can be accessed via several R packages:

- library **e1071** : use function **svm()**
- library **LiblineaR** : function is also named **LiblineaR()**
- library **kernlab** : use function **ksvm()** or **lssvm()**
- library **caret** : use function **train()** with a **method parameter**:
  - method = 'svmLinearWeights2’ (uses LiblineaR)
  - method = 'svmLinear3’ (uses LiblineaR)
  - method = 'lssvmLinear’ (uses kernlab)
  - method = 'lssvmPoly’ (uses kernlab)
  - method = 'lssvmRadial’ (uses kernlab)
  - method = 'svmLinearWeights’ (uses e1071)