

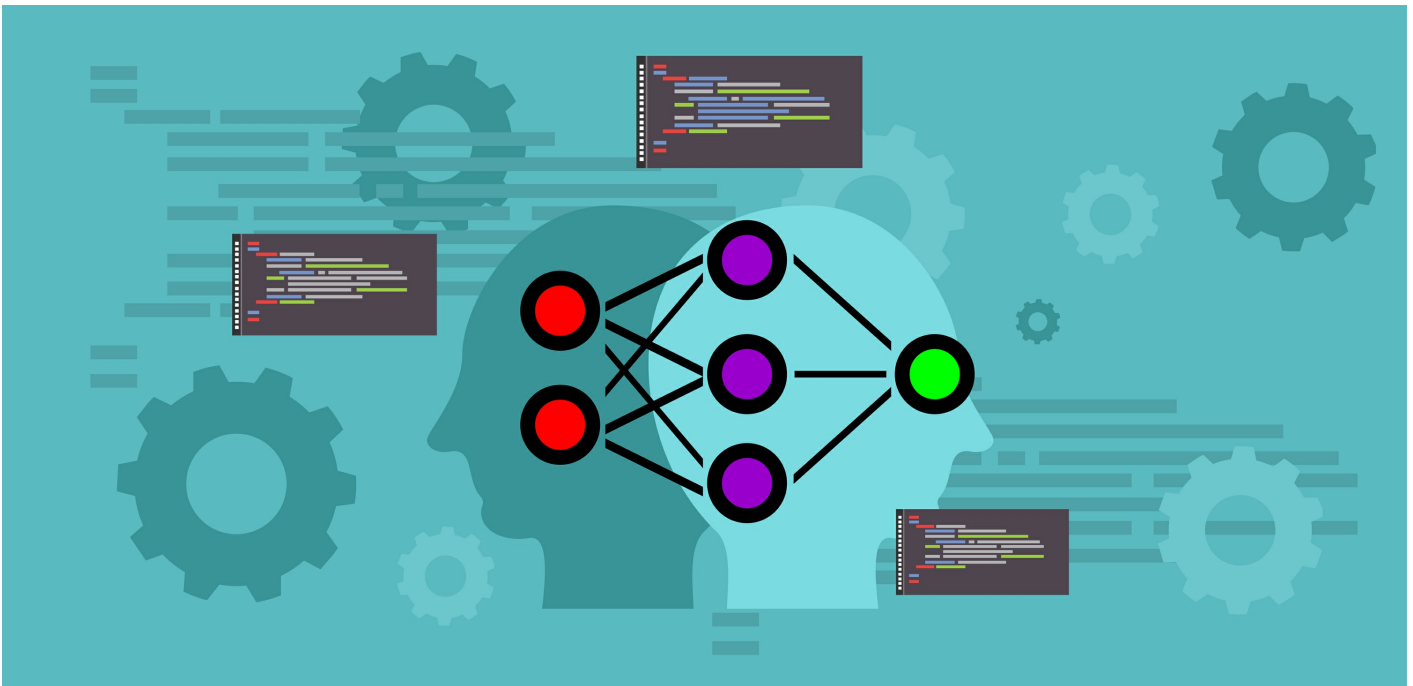
LEARNING CORNER



ROHIT
GARG – JAN 19,
2018



Advertisement



The purpose of this research is to put together the 7 most commonly used classification algorithms along with the python code: Logistic Regression, Naïve Bayes, Stochastic Gradient Descent, K-Nearest Neighbours, Decision Tree, Random Forest, and Support Vector Machine

1 Introduction

1.1 Structured Data Classification

Classification can be performed on structured or unstructured data. Classification is a technique where we categorize data into a given number of classes. The main goal of a classification problem is to identify the category/class to which a new data will fall under.

Few of the terminologies encountered in machine learning – classification:

Over 100,000 people subscribe to our newsletter.



See stories of Analytics and AI in your inbox.

- **Classifier:** An algorithm that maps the input data to a specific category.
- **Classification model:** A classification model tries to draw some conclusion from the input values given for training. It will predict the class labels/categories for the new data.
- **Feature:** A feature is an individual measurable property of a phenomenon being observed.
- **Binary Classification:** Classification task with two possible outcomes. Eg: Gender classification (Male / Female)
- **Multi class classification:** Classification with more than two classes. In multi class classification each sample is assigned to one and only one target label. Eg: An animal can be cat or dog but not both at the same time
- **Multi label classification:** Classification task where each sample is mapped to a set of target labels (more than one class). Eg: A news article can be about sports, a person, and location at the same time.

The following are the steps involved in building a classification model:

- **Initialize** the classifier to be used.
- **Train the classifier:** All classifiers in scikit-learn uses a fit(X, y) method to fit the model(training) for the given train data X and train label y.
- **Predict the target:** Given an unlabeled observation X, the predict(X) returns the predicted label y.
- **Evaluate** the classifier model

Also Read [Document Classification using Apache Spark in Scala](#)

1.2 Dataset Source and Contents

The dataset contains salaries. The following is a description of our dataset:

- **of Classes:** 2 ('>50K' and '<=50K')
- **of attributes (Columns):** 7
- **of instances (Rows):** 48,842

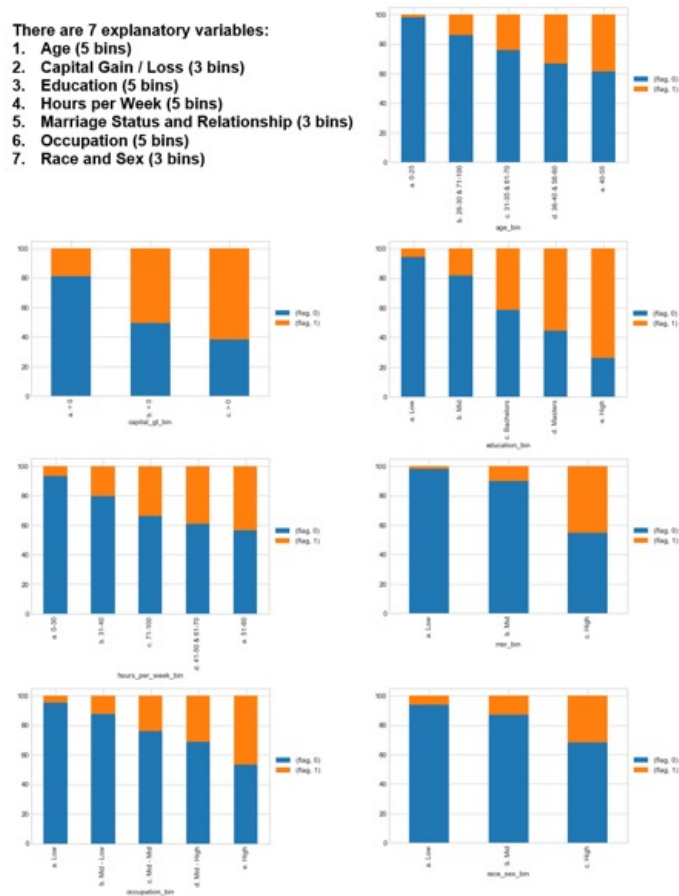
This data was extracted from the census bureau database found at:

<http://www.census.gov/ftp/pub/DES/www/welcome.html>

1.3 Exploratory Data Analysis

There are 7 explanatory variables:

1. Age (5 bins)
2. Capital Gain / Loss (3 bins)
3. Education (5 bins)
4. Hours per Week (5 bins)
5. Marriage Status and Relationship (3 bins)
6. Occupation (5 bins)
7. Race and Sex (3 bins)



2 Classification Algorithms (Python)

2.1 Logistic Regression

Definition: Logistic regression is a machine learning algorithm for classification. In this algorithm, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function.

Advantages: Logistic regression is designed for this purpose (classification), and is most useful for understanding the influence of several independent variables on a single outcome variable.

Disadvantages: Works only when the predicted variable is binary, assumes all predictors are independent of each other, and assumes data is free of missing values.

```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(x_train, y_train)
y_pred=lr.predict(x_test)
```

2.2 Naïve Bayes

Definition: Naive Bayes algorithm based on Bayes' theorem with the assumption of independence between every pair of features. Naive Bayes classifiers work well in many real-world situations such as document classification and spam filtering.

Advantages: This algorithm requires a small amount of training data to estimate the necessary parameters. Naive Bayes classifiers are extremely fast compared to more sophisticated methods.

Disadvantages: Naive Bayes is known to be a bad estimator.

```
from sklearn.naive_bayes import GaussianNB
nb = GaussianNB()
nb.fit(x_train, y_train)
y_pred=nb.predict(x_test)
```

2.3 Stochastic Gradient Descent

Definition: Stochastic gradient descent is a simple and very efficient approach to fit linear models. It is particularly useful when the number of samples is very large. It supports different loss functions and penalties for classification.

Advantages: Efficiency and ease of implementation.

Disadvantages: Requires a number of hyper-parameters and it is sensitive to feature scaling.

Also Read [Understanding AVA– Image Discovery Tool Used By Netflix To Power Its Content Posters](#)

```
from sklearn.linear_model import SGDClassifier
sgd = SGDClassifier(loss='modified_huber', shuffle=True, random_state=101)
sgd.fit(x_train, y_train)
y_pred=sgd.predict(x_test)
```

2.4 K-Nearest Neighbours

Definition: Neighbours based classification is a type of lazy learning as it does not attempt to construct a general internal model, but simply stores instances of the training data. Classification is computed from a simple majority vote of the k nearest neighbours of each point.

Advantages: This algorithm is simple to implement, robust to noisy training data, and effective if training data is large.

Disadvantages: Need to determine the value of K and the computation cost is high as it needs to compute the distance of each instance to all the training samples.

```
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors=15)
knn.fit(x_train, y_train)
y_pred=knn.predict(x_test)
```

2.5 Decision Tree

Definition: Given a data of attributes together with its classes, a decision tree produces a sequence of rules that can be used to classify the data.

Advantages: Decision Tree is simple to understand and visualise, requires little data preparation, and can handle both numerical and categorical data.

Disadvantages: Decision tree can create complex trees that do not generalise well, and decision trees can be unstable because small variations in the data might result in a completely different tree being generated.

```
from sklearn.tree import DecisionTreeClassifier
dtree = DecisionTreeClassifier(max_depth=10, random_state=101,
                              max_features = None, min_samples_leaf = 15)
dtree.fit(x_train, y_train)
y_pred=dtree.predict(x_test)
```

2.6 Random Forest

Definition: Random forest classifier is a meta-estimator that fits a number of decision trees on various sub-samples of datasets and uses average to improve the predictive accuracy of the model and controls over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement.

Advantages: Reduction in over-fitting and random forest classifier is more accurate than decision trees in most cases.

Disadvantages: Slow real time prediction, difficult to implement, and complex algorithm.

Also Read [Spreadsheet Veracity & Lineage – Managing Spreadsheet Series: 3 of 5](#)

```
from sklearn.ensemble import RandomForestClassifier
rfm = RandomForestClassifier(n_estimators=70, oob_score=True, n_jobs=-1,
                           random_state=101, max_features = None, min_samples_leaf = 30)
rfm.fit(x_train, y_train)
y_pred=rfm.predict(x_test)
```

2.7 Support Vector Machine

Definition: Support vector machine is a representation of the training data as points in space separated into categories by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

Advantages: Effective in high dimensional spaces and uses a subset of training points in the decision function so it is also memory efficient.

Disadvantages: The algorithm does not directly provide probability estimates, these are calculated using an expensive five-fold cross-validation.

```
from sklearn.svm import SVC
svm = SVC(kernel="linear", C=0.025, random_state=101)
svm.fit(x_train, y_train)
y_pred=svm.predict(x_test)
```

3 Conclusion

3.1 Comparison Matrix

- **Accuracy: (True Positive + True Negative) / Total Population**

- Accuracy is a ratio of correctly predicted observation to the total observations. Accuracy is the most intuitive performance measure.
- True Positive: The number of correct predictions that the occurrence is positive
- True Negative: The number of correct predictions that the occurrence is negative

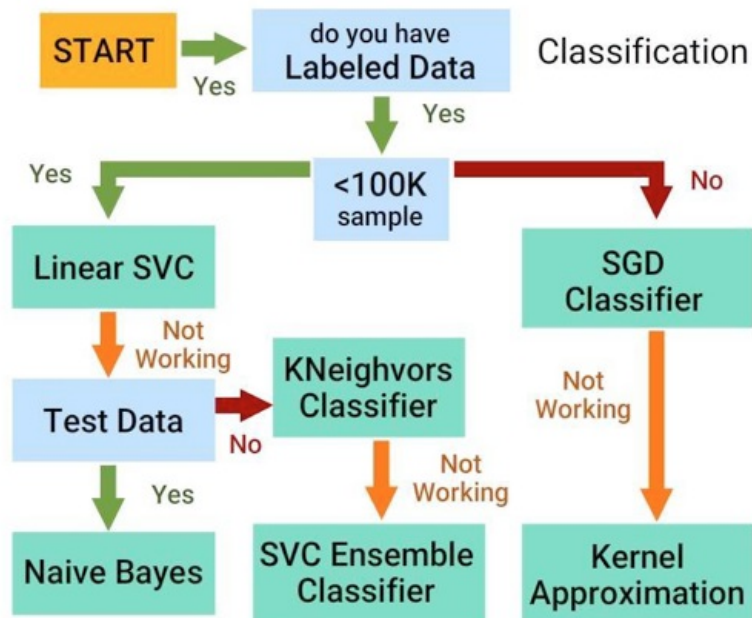
- **F1-Score: (2 x Precision x Recall) / (Precision + Recall)**

- F1-Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. F1-Score is usually more useful than accuracy, especially if you have an uneven class distribution.
- Precision: When a positive value is predicted, how often is the prediction correct?
- Recall: When the actual value is positive, how often is the prediction correct?

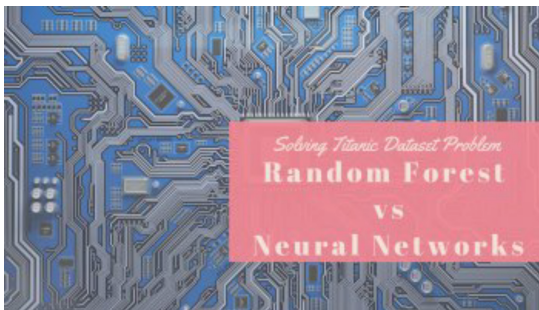
Classification Algorithms	Accuracy	F1-Score
Logistic Regression	84.60%	0.6337
Naïve Bayes	80.11%	0.6005
Stochastic Gradient Descent	82.20%	0.5780
K-Nearest Neighbours	83.56%	0.5924
Decision Tree	84.23%	0.6308
Random Forest	84.33%	0.6275
Support Vector Machine	84.09%	0.6145

Code location: <https://github.com/f2005636/Classification>

3.2 Algorithm Selection



Related



Solving The Titanic ML Survival Problem Using Random Forest vs Neural Networks on Tensorflow. Which One is Better?

Jan 8, 2018

In "Learning Corner"



Start Building Your First Machine Learning Project With This Famous Dataset

Feb 6, 2018

In "Learning Corner"



Document Classification using Apache Spark in Scala

Sep 26, 2016

In "Learning Corner"

Provide your comments below

2 comments

2 Comments

Sort by Top

Add a comment...



Rahul Tandon ·
Indian Institute of Technology, Bombay
this article is another collectors item. thanks !

Like · Reply · 2 · 11w



Sanjay Krishnamurthy ·
Annamalai University
Nice article..

Like · Reply · 1 · 11w

Facebook Comments Plugin

Rohit Garg

Rohit Garg has close to 7 years of work experience in field of data analytics and machine learning. He has worked extensively in the areas of predictive modeling, time series analysis and segmentation techniques. Rohit holds BE from BITS Pilani and PGDM from IIM Raipur.

SHARE THIS

9

0

0

0

0

0

PREVIOUS ARTICLE

Top 5 Challenges That Are Holding Back AI Startups To Scale Fast

NEXT ARTICLE

Indian Economy To Reach \$5 Trillion By 2025, AI And IoT Will Be Major Contributors, Says NITI Aayog Chief

SEARCH

Search



BITS Pilani
Pilani | Dubai | Goa | Hyderabad

PG Certification in **Big Data** Engineering

200+ hrs of learning



APPLY NOW



Become a Data Scientist
with India's Best


POST GRADUATE
DIPLOMA IN DATA SCIENCE

APPLY NOW



MANIPAL
ACADEMY of HIGHER EDUCATION

(Deemed to be University under Section 3 of the UGC Act, 1956)



ANALYTIX LABS

Your Gateway to great career in Analytics!

REGISTER

Sign up for a free live online class



GREAT LAKES
INSTITUTE OF MANAGEMENT

greatlearning

LEADERS IN ANALYTICS

proschool
An **ims** Initiative



Govt. Approved
9 Months
PG Diploma in
Data Science



10 ivy
YEARS Professional School

BECOME A DATA SCIENTIST

- Expert faculty from IIT, ISI, IIM.
- Placement Assistance

LATEST VIDEO

Behind The Scenes: Data Visualisation In Business Intelligence ft BDB



THE MACHINE CONFERENCE 2018

11th MAY 2018
HOTEL HYATT REGENCY, MUMBAI
WWW.THEMACHINECON.COM

Over 100,000 people subscribe to our newsletter.

See stories of Analytics and AI in your inbox.





CAREER WITH

[ABOUT US](#)

[US](#)

[ADVERTISE](#)

[COPYRIGHT](#)

[PRIVACY](#)

[TERMS OF USE](#)

[CONTACT US](#)

Copyright 2018 Analytics India Magazine Pvt Ltd. ALL RIGHTS RESERVED.

