Simple Linear Regression

Typical use

Modeling the connection between two continuous variables requires simple linear regression. Forecasting the value of an outcome (or response) from the value of an input variable (or predictor variable) is frequently the purpose. The degree of relationship between two variables can be established using simple linear regression. the value of the dependent variable at a particular independent variable value.

Observations on the output

There are a number of non-linear features included in the lstat and medv relationship from Boston Data. Lm.fit gives basic information about the model's output. To offer additional information, we were using a summary (lm.fit). This gives the model's coefficients' standard errors, p-values, R2 statistic, F-statistic, as well as other statistics.

Most significant finding of the exercise

The lm() method can be utilized to fit a straightforward linear regression model. Regression, vertical, or horizontal lines might well be added to a graph with the abline() function. When the lwd is equal to n command is employed, the regression line's width is multiplied by the number of n. A number of charting symbols can be constructed using the pch option.

Multiple Linear Regression

Typical use

A number of independent variable are combined in a statistical process called multiple linear regression (MLR), also referred to as multiple regression. Modeling the linear relationship between the explanatory (independent) factors and response (dependent) variables is the aim of multiple linear regression. Because multiple regression takes into account several explanatory variables, it can be thought of as an elaboration of ordinary least-squares (OLS) regression.

Observations on the output

Each coefficient is accompanied by its Estimate, Standard Error, t value, and Pr(>|t|) in the summary. The one most of them are significant, but age is not since a number of other predictors are related to age and age is no longer beneficial in their presence.

Most significant finding of the exercise

A multivariate linear regression model was fitted using least squares using the lm() method. To perform a regression using all the predictors, one would have to write all 12 variables in the Boston data set, which can be challenging. Instead, lm(medv ., data = Boston) may be employed as a shortcut. If we want to run a regression using all except one of the variables. All predictors are used in a regression except age when the syntax is lm(medv . - age, data = Boston).

Qualitative Predictors

Typical use

In several domains, qualitative predictors—also referred to as categorical variables, aka dummy variables, indicator variables or, in R's language, factors—are widely used. They must be handled with caution and a thorough comprehension of how these variables are expressed.

Observations on the output

Interactional factors Advertising seems to have a major impact on income, while age does not. The contrasts() function creates two fake variables. Fair and Moderate

Logistic Regression

Typical use

Logistic Regression is a widely used statistical algorithm in machine learning for classification tasks. It is a supervised learning algorithm that is used to predict the probability of a binary (two-class) or multi-class outcome based on one or more predictor variables.

Observations on the output

In this case, Lag1 has the lowest p-value. With a positive return yesterday, the market is less likely to rise today, according to the predictor's negative coefficient. There is no conclusive proof of a true association between lag1 and direction because the p-value, even with a value of 0.15, is still high.

Most significant finding of the exercise

The probability estimates are generated for the training data that were used to fit the logistic regression model if a data set is not given to the predict() method.

Linear Discriminant Analysis

Typical use

In statistics and other domains, linear discriminant analysis (LDA) is a technique used to identify a linear combination of attributes that distinguishes between two or more classes of objects or events. It is possible to utilize the resulting combination as a linear classifier or, more frequently, to reduce the dimensionality before further classification.

Observations on the output

The output of LDA, a practical and adaptable machine learning technique, can offer insightful information about the functionality and behavior of the model. In addition to dimensionality reduction and feature selection, it is frequently utilized for classification problems. To get the optimum performance for a particular application, it is crucial to carefully select the right number of features and fine-tune the algorithm's hyperparameters. The LDA result shows that 49.2% of the training observations, or 1=0.492 and 2=0.508, correlate to days when the market decreased. It also provides the group means, which LDA employs to calculate k. Each predictor within each class is averaged out to create the group means. They show a tendency for the previous two days' returns to be positive on days when the market declines and a tendency for them to be negative on days when the market rises.

Most significant finding of the exercise

Data visualization, dimension reduction, and classification are all done using linear discriminant analysis. It has been in circulation for a while. Even though LDA is straightforward, it frequently yields reliable, respectable, and understandable classification results. LDA is frequently used as a benchmarking technique before turning to other, more intricate and flexible techniques when solving real-world classification problems.

The MASS library's lda() function, which fits an LDA model, is used. The coefficients of linear discriminants are used to produce the LDA decision rule from the equation lda(Direction Lag1 + Lag2, data = Smarket, subset = train). This equation yields the linear combination of lag1 and lag2 that is used.

Quadratic Discriminant Analysis

Typical use

Applications for machine learning and recognition of patterns routinely use the statistical classification technique known as quadratic discriminant analysis (QDA). Because it is a form of supervised learning, the model must be trained using labeled data.

Observations on the output
Using QDA, the group means are produced. Nevertheless, because the QDA classifier utilizes a quadratic function of the predictors rather than a linear one, it does not incorporate the coefficients of the linear discriminants. It also provides insight into the decision boundary between the classes and how the model is making its decisions.

Most significant finding of the exercise
The qda () function is used to fit an LDA model. Compared to the linear forms assumed by LDA and logistic regression, the quadratic form presented by QDA may more correctly capture the underlying relationship. Before concluding that this approach will continuously outperform the market, we suggest verifying its performance on a broader test set.

Naive Bayes
Typical use
Naive Bayes, which is rapid, can quickly and accurately predict the class of a test dataset. Because it functions well with them, it can be used to fix problems with multi-class prediction. The Naive Bayes classifier performs better than other models with less training data if the independence of features is real.

Observations on the output
Despite its "naive" assumption of feature independence, the output of Naive Bayes is observed to typically perform well in practice. This is because Naive Bayes may still identify significant patterns in the data and make precise predictions, even though the assumption of feature independence is frequently false in practice. Finally, it's important to remember that the output of Naive Bayes can depend on the prior probability selected, especially if the training data is unbalanced. To make accurate forecasts in such circumstances, it could be required to modify the prior probabilities.

Naive Bayes configured to run on the best available information, generating accurate predictions more than 59% of the time. This operates moderately worse than QDA, but much better than LDA.

Most significant finding of the exercise
It has been demonstrated that Naive Bayes is very interpretable since it offers a simple method for estimating the weight of each feature in the classification decision. In many applications, it is crucial to comprehend the variables influencing the classification choice, hence interpretability is key. Altogether, Naive Bayes has had a significant impact on machine learning. Its practical effectiveness and high-dimensional data compatibility have made it a preferred choice for numerous practical applications.

Naive Bayes is implemented using R's naiveBayes() function. Each quantitative feature is modelled by default with a Gaussian distribution in this naive Bayes classifier implementation. Estimates of the likelihood that each observation belongs to a particular class can be generated by the predict() function.

K-Nearest Neighbors
Typical use
K-nearest neighbors (KNN) is a non-parametric algorithm used for both classification and regression tasks in machine learning. The basic idea behind KNN is to classify or predict a new data point based on the class or value of its K-nearest neighbors in the feature space.

Observations on the output

K-Nearest Neighbors (KNN) is a popular non-parametric machine learning algorithm used for classification and regression tasks. The output of KNN can provide insights into the performance and behavior of the model. Some observations on the output of KNN include:

Decision boundaries: KNN is a distance-based algorithm that uses the proximity of instances to make predictions. The output of KNN can be used to visualize the decision boundaries of the model in the feature space. This can be helpful in understanding how the model makes predictions and identifying regions of the feature space where the model may be uncertain.

Computational complexity: The computational complexity of KNN can be relatively high, especially for large datasets with many features. The output of KNN can provide information on the training and prediction times of the model, which can be used to assess the computational cost of the algorithm.

With K=1, only 50% of the data are correctly predicted, hence the outcomes are not very good. With K=3, results have slightly improved. Increasing K further, it turns out, does not yield any additional benefits. The QDA method appears to yield the best outcomes for this data out of all the ones we've examined so far.

Most significant finding of the exercise

KNN is carried out using the knn() method, which is a component of the class library. The size of the variables is crucial when using the KNN classifier to identify the observations that are most similar to a specific test observation. The KNN classifier will be substantially more impacted by large-scale variables than by small-scale variables in terms of the separation between the observations.