APPLIED PREDICTIVE MODELING

AN OVERVIEW OF APPLIED PREDICTIVE MODELING

STEVEN TAYLOR

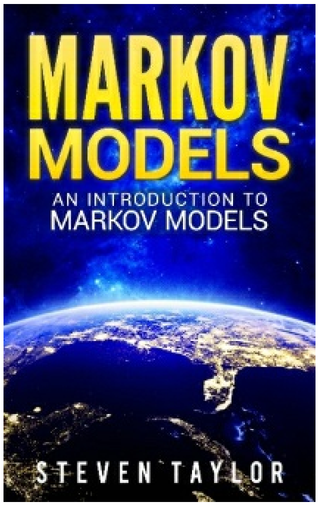
Applied Predictive Modeling

An Overview of Applied Predictive Modeling

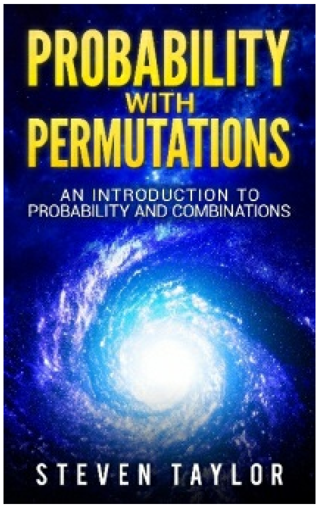
By Steven Taylor

Please Check Out My Other Books Before You Continue

Below you will find my other popular books that are popular on Amazon and Kindle as well. Simply click on the links below to check it out.



Markov Models



Probability with Permutations

If the links do not work, for whatever reason, you can simply search for the titles on the Amazon website to find the books.

The following publication is reproduced below with the goal of providing information that is as accurate and reliable as possible. Regardless, purchasing this publication can be seen as consent to the fact that both the publisher and the author of this book are in no way experts on the topics discussed within and that any recommendations or suggestions for example those that made herein are for informational purposes only. Professionals should be consulted as needed prior to undertaking any of the action endorsed herein.

This declaration is deemed fair and valid by both the American Bar Association and the Committee of Publishers Association and is legally binding throughout the United States.

Furthermore, the transmission, duplication or reproduction of any of the following work including specific information will be considered an illegal act irrespective of if it is done electronically or in print. This extends to creating a secondary or tertiary copy of the work or a recorded copy and is only allowed with express written consent from the Publisher. All additional right reserved.

The information in the following pages is broadly considered to be a truthful and accurate account of facts and as such any inattention, use or misuse of the information in question by the reader will render any resulting actions solely under their purview. There are no scenarios in which the publisher or the original author of this work can be in any fashion deemed liable for any hardship or damages that may befall them after undertaking information described herein.

Additionally, the information in the following pages is intended only for informational purposes and should thus be thought of as universal. As befitting its nature, it is presented without assurance regarding its prolonged validity or interim quality. Trademarks in particular are mentioned without written consent and can in no way be considered an endorsement from the trademark holder.

The trademarks particularly used are without any consent, and the publication of the trademark is without permission or backing by the trademark owner.

All trademarks and brands within this book are for clarifying purposes only and are the owned by the owners themselves, not affiliated with this document.

Table of Contents

Introduction

Predictive modeling is a process of creating models. Predictive modeling further requires testing and validating that model in order to predict the probability of any outcome. When it comes to the predictive analytics software, there are a number of different modeling methods including artificial intelligence, machine learning, and statistics. These methods are used as certain predictive analytics software solutions in order to predict an outcome. Every model is chosen on the certain basis of validation, testing, and evaluation using the detection theory in order to guess the probability of an outcome in terms of a given set of various input ideas.

Models can use multiple classifiers when it comes to the trying to determine the certain probability of a collection of data specifically when belonging to another collection. Different models prominently available on the particular modeling portfolio, in fact, allow you to obtain relevant and useful information. The model also gives you the information needed to develop different predictive models. Every derived model has its own weakness and strengths, so it is best suited for certain types of problems to be solved.

Every model is created by training a certain algorithm by using various historical data. Further, the model is saved in order to be used again, so every model is reusable. In order to analyze relevant results, but without the historical data, a reused model is applied using a previously trained algorithm. Common business rules can be easily applied to the various similar data. Majority of the predictive modeling solutions has the capability in order to export various models information into a certain local file in PMML or Predictive Modeling Markup Language. This standard format is used for sharing your model with some other PMML applications in order to perform analysis on different similar data. Predictive modeling solutions commonly use statistics in order to predict an outcome.

The event that you want to predict most commonly is in the future. However, predictive modeling also may be applied to various unknown events as well, regardless of when those events happened. For instance, predictive models are commonly used in order to detect criminal affairs and identify suspects, after a certain crime has occurred.

In a majority of cases, the model is chosen by using detection theory in order to guess the probability of any outcome when you have a set amount of various input data. For instance, you can use an email and try to determine how likely that email is a spam. In this case, models may use multiple classifiers when it comes to the trying to determine that data set. Data set in this case may be ham or spam. Predictive modeling is commonly synonymous with the field of machine learning depending on some definitional boundaries. Predictive modeling is also commonly referred to as a development or research context. When it comes to the deploying predictive models commercially, then they are referred to as the field of predictive analytics.

Predictive modeling processes also use data mining when it comes to the predicting outcomes. Every model is based on a different number of various predictors. Predictors are different variables which greatly influence every future result or outcome. Once data is gathered for different relevant predictors, statistical models are formulated. These models also may employ a complex neural network as well as a simple linear equation eminently mapped out by some sophisticated software. Further, additional data becomes visible so statistical analysis model is revised or validated. Predictive modeling techniques are commonly associated with weather forecasting and meteorology. However, predictive modeling has many other applications when it comes to the business management. For instance, Bayesian spam filters are commonly used in predictive modeling in order to identify the probability that some given message is a spam. When it comes to the fraud detection, predictive modeling is widely used in order to identify different outliers in a data collection which directly point toward some fraudulent activity.

When it comes to the customer relationship management, predictive modeling techniques are used in order to target messaging certain customers who will most likely make purchases. Other applications of predictive modeling include change management, capacity planning, physical and digital security, city planning, engineering and disaster recovery. It may be very tempting to think that with a wide range of big data sets, predictive models may be more accurate. It should be noted that various statistical theorems, in fact, show that after a particular point when it comes to the feeding more data into some predictive analytics model will not provide and obtain more accurate results. However, analyzing some representative portions of already available information known as the process of data sampling, can greatly help the overall speed development time on various models as well as allow them to be ready for deployment more quickly. Predictive modeling is simply the process of using already obtained results in order to create and validate a certain model peculiarly used in order to predict various future outcomes. Predictive modeling is a valuable tool used in the field of predictive analytics. Predictive modeling, in fact, aims to answer on various questions regarding what may occur in the future, and what are the possible outcomes. With predictive modeling, you simply try to determine what all possible outcomes.

It is more than apparent that this rapid migration of various digital products, in fact, has created large sets of data commonly very available as well as accessible for various businesses. Big data is commonly utilized by organizations and companies in order to improve businesses' overall dynamics when it comes to the customer-to-business connection. This wide range of real-time data set is gotten from various sources including internet browsing history, cloud computing platforms, smartphone data and social media.

By analyzing various historical data, there is a certain probability which a business may be able to predict so they can plan accordingly to those predictions. It should be noted that data obtained is commonly too complex as well as unstructured for humans to analyze it in terms of a short period of time. Due to data complexity, which large amounts of data present, organizations and companies increasingly are using various predictive analytics tools in order to forecast the relevant outcomes of any event that is likely to occur in the nearest future. Predictive analytics firstly collects than further processes various historical data in large amounts and uses some powerful computers in order to assess what occurred in the past. Predictive analytics then uses different predictors as well as some known features in order to create a predictive model principally used in obtaining outputs. Any predictive model is also used in order to learn how various points of data are connected with each other. Two of the most commonly used predictive modeling techniques are neural networks and regression.

Chapter 1 Predictive Modelling Techniques

When it comes to the field of statistics, regression mainly refers to some linear relationship that exists between output and input variables. It should be noted that every predictive model in terms of linear function requires a predictor or feature to predict the outcome or output. For instance, a bank which hopes to detect fraudulent activity like money laundering in its beginning stages commonly incorporate this linear predictive model. The bank wants to know which of its customers are very likely to engage in fraudulent activities like money laundering at some point. When the bank's customer data is obtained and presented, further a predictive model will be built around some dollar value transfer in order to recognize that difference existing between a normal transaction and a money laundering transaction.

The optimal outcome of any predictive model should represent a pattern greatly signaling which customer most likely laundered money and which customer didn't engage in fraudulent activities. When the model is obtained, it perceives when a pattern of certain fraud is emerging when it comes to the certain customers. Further, the model will create a signal for a specific action that will be attended to by fraud analysts.

Predictive models are commonly used in neural networks like deep learning and machine learning that are fields in Artificial Intelligence. In fact, the neural networks are mainly inspired by the human brain and they are created containing a web of various interconnected nodes represented in hierarchical levels singally representing the main foundation for Artificial Intelligence. The real power of neural networks commonly lies in the capability to handle various nonlinear data relationships. The neural networks are also able to create various patterns and relationships between different variables which prove to be too time consuming and too complex for human analysts. If we take the bank example again, it may know various variables like a value of transfers markedly initiated by customers into the model to obtain that desired outcome and to find out who of its customers is likely to engage in fraudulent activities like money laundering.

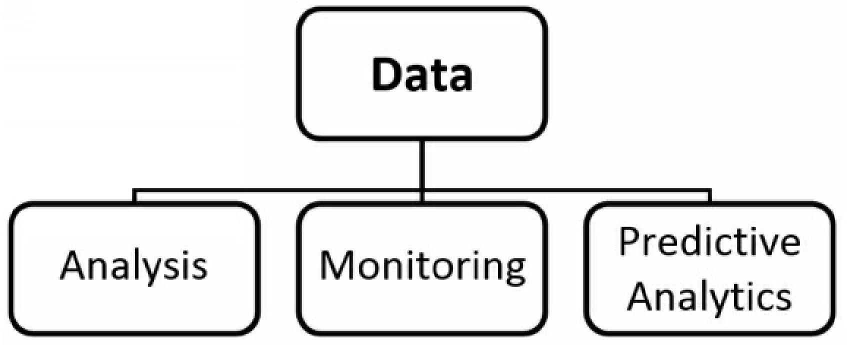
A neural network is capable of creating some more powerful patterns in the cases when it successfully create that relationship existing between input variables such as time logged in, IP address of the customer's device, geographic location, sender or recipient of the funds, and all other relevant features, which are likely to make up a fraudulent activity.

Other predictive modeling techniques commonly used by financial organizations and companies include time series data mining, decision trees, and Bayesian analysis. Organizations and companies take predictive modeling advantages in terms of big data through various predictive modeling measures highly capable of better understanding how customers engage with their various products. In other words, companies are able to identify potential risks as well as various opportunities for companies.

Regression Model

When it comes to the predictive models, it should be noted that any regression model may be used for various predictive purposes. Generally speaking, there are two most commonly used classes of models non-parametric and parametric. There is also a third class of predictive models known as semi-parametric that includes features of both parametric and non-parametric models. Parametric models make various assumptions regarding one or multiple population parameters noticeably characterizing various underlying distributions. Non-parametric regressions, on the other hand, make a fewer number of assumptions in comparison to their parametric counterparts.

We use logistic or linear regression techniques in order to develop more accurate models for guessing an outcome of our interest. Commonly, we create various separate segments in order to judge their overall effectiveness. However, it is much better way to create a single model and then enable it with incorporating segmentation variables which refer to the model as an input. This is the better way to go since creating various separate models including separate segments is very time-consuming and simply not worth the overall effort. On the other hand, creating separate models with separate segments can provide greater predictive power.



When it comes to the predictive modeling, logistic and linear regressions are commonly referred to as the first algorithms. Due to their enormous popularity, a majority of analysts end up incorporating regression techniques into their models. The truth about regression models is that there is a wide range of different forms of regressions that can be easily incorporated and used. Each regression form, in fact, has its own specific condition and importance which comes suitably in different situations.

Regression analytics is a predictive modeling technique that investigates an existing relationship between an independent variable and dependent target variable. The technique is commonly used for predicting outcomes, finding time series as well as finding a causal effect relationship existing between variables. For instance, a relationship between a number of road accidents and rash driving may be studied profoundly through regression techniques.

Regression analysis is a very important tool for analyzing and modeling various data. The technique enables you to find the differences between different distances in terms of data points from a certain curve or line represented on regression graph. Regression analysis is used in order to estimate the relationship existing between multiple variables. For instance, if you want to find out and estimate growth in sales of a business generally based on various current economic conditions you will use regression analysis techniques. You will have the most recent company data that indicates that business growth in comparison to the overall growth in the economy. If you use that insight, you are able to predict business’s future sales of any company primarily based on the past and current information.

Regression analysis brings multiple advantages including indicating the relevant relationship existing between independent and dependent variables and indicating the overall strength of obtained multiple dependent and independent variables. Regression analysis allows you to compare different effects of those variables greatly measured on various scales like the effect of different price changes as well as the number of different promotional activities. These regression benefits may greatly improve market management and data analysis in order to eliminate as well as to evaluate the best collection of variables highly used for building further predictive models.

There are different types of regression techniques used in order to make various predictions. These regression techniques are commonly driven by three specific metrics including independent variables, a shape of the regression line and kind of dependent variables. We already know that regression models involve three parameters including a dependent variable, independent variable and various unknown parameters commonly represented as vector or scalar. In different fields of regression analysis applications, there are different terminologies used. On the other hand, a regression model is always based on knowledge about the certain relationship between independent and dependent variables, which do not rely on the certain data. If there is no such knowledge available, certain flexible and more convenient form of a regression model will be used.

Regression analysis provides different tools for finding various solutions for different unknown parameters in order to minimize the distance that exists between predicted and measured values of certain dependent variables. The methods are known as least squares methods. Regression analysis also under particular statistical assumptions outstandingly using obtained information in order to provide statistical information about some unknown parameters as well as about some predicted values in terms of dependent variables.

If you consider certain regression model that includes three unknown parameters, an experimenter most certainly will perform ten measurements and all of them will be at an exact same value when it comes to the obtained independent variables. In this certain case, regression commonly fails to give a specific collection containing estimated values for those three unknown parameters. In this case, the best thing you can do is to estimate that average value as well as the standard deviation when it comes to the obtained dependent variables. In a similar manner, if you measure two different values of an independent variable, you will obtain enough data in order to integrate regression containing two unknown variables. It should be noted that you will not be able to find enough data if your regression contains three and more unknown variables.

In the case when the experimenter performs measurements at some various three values in term of independent variables, then regression will provide a unique collection of different estimates for those three unknown parameters. If you have a case, which involves general linear regression, the statement from the above is equivalent to the certain requirement that the obtained matrix of independent variables is, in fact, invertible.

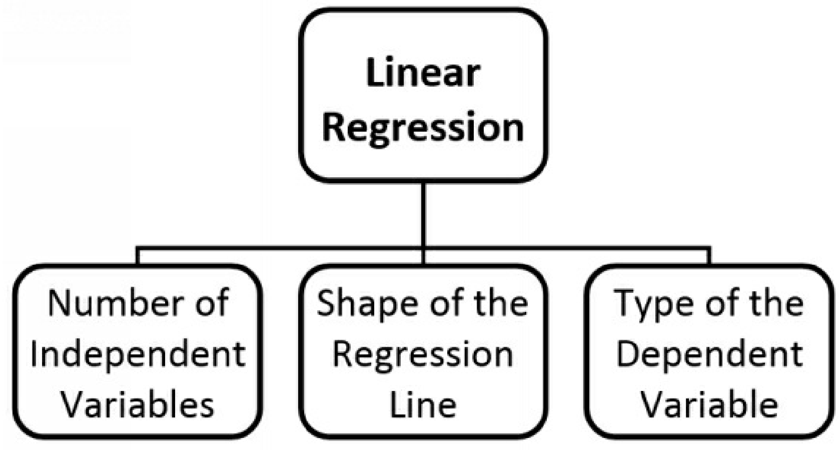
When the total number of measurements is larger than that number of obtained unknown parameters, the measurement errors will be normally distributed. These obtained measurements are also used in order to make different statistical predictions about obtained unknown parameters. You will have to access different information greatly referred to as the certain degrees of freedom.

Linear regression is the most common technique used when it comes to the predictive modeling. In linear regression technique, an independent variable is commonly discrete or continuous while the dependent variable is commonly only continuous. Linear regression models establish an existing relationship between one or multiple independent variables and dependent variables using a straight line known as the regression line.

Linear Regression Equation

$$Y = a + b * X + e$$

Linear regression is represented by the equation above. In the equation, a represents intercept while b is the slope of regression line. In the linear regression equation, e represents error term. This equation is commonly used in order to predict the certain value of some target variable signally based on already given predictor variables. When it comes to the difference between multiple linear regression and simple linear regression there is just one difference. Simple linear regression has a single independent value, while multiple linear regressions have more than one independent value.


Linear Regression
Number of Independent Variables
Shape of the Regression Line
Type of the Dependent Variable

In order to obtain a value of a and value of b, you can use method Least Square. This method is most commonly used when it comes to the linear regression models. The method easily can calculate the best possible fit line for obtained data by minimizing the overall sum of all squares, notably presented in the vertical deviations from every data point to the line. The deviations, in fact, are first squared when they are added so there is no any canceling out when it comes to the negative and positive values.

You can evaluate your model performance using certain R-square metric. Important point you should remember is that there has to be a certain linear relationship between dependent and independent variables. It should be noted as well that multiple regression commonly suffers from autocorrelation, multicollinearity, and heteroscedasticity. On another hand, linear regression is also very sensitive to various Outliers. That can greatly affect the overall regression line and it can eventually affect the predicted values as well.

Multicollinearity may also greatly increase the certain variance, which exists between coefficient estimates. It can make these estimates very sensitive even to some minor changes in the regression model. This commonly results in unstable coefficient estimates. When it comes to the multiple independent variables, you can use backward elimination, forward selection and stepwise regression approach in order to select independent variables of the greatest significance.

Logistic Regression

Logistic regression is widely used in order to find the probability of event success and event failure. We use logistic regression when there is a dependent variable above all binary variable. In other words, the dependent binary is true or false and yes or no depending on number preeminently used to denote the dependent variable. When you are working with various binomial distributions or dependent variables, first you have to choose a connection function that will suit the best for relevant distribution. That will be a logit function. Parameters will be used in order to maximize the probability of observing some sample values rather than just minimizing the overall sum of various squared error prominently used in the ordinary regression.

Important points of logistic regression include the fact that logistic regression commonly doesn't require any linear relationship between independent and dependent variables. Logistic regression is widely used in order to solve various classification problems since it can easily handle different types of connections between variables since it can predict various odd ratios by applying a nonlinear transformation. In order to avoid under- fitting and over-fitting, you should include all relevant variables. The best approach is to first ensure that logistic regression practices are used as a step wise method in order to estimate your logistic regression model.

The model most likely will require large sample sizes due to fact that maximum probability estimates are commonly less powerful at some low sample sizes in comparison to some ordinary least square approach. You should keep in mind that independent variables shouldn't correlate with other variables which is an example of no multicollinearity. On the other hand, you have an option to include various interaction effects in terms of categorical variables in the model. In the case when the values of certain dependent variables are ordinal then we call it Ordinal logistic regression. On the other hand, when dependent variables are multi class, we call it as Multi-nominal logistic regression.

Stepwise Regression

Stepwise regression is a form of regression modeling commonly used when dealing with some multiple independent variables. In this particular technique, the certain selection containing independent variables is done with the great of automated processes that involve no human intervention. This feature, in fact, is achieved by observing various statistical values such as R-square, AIC, t-stats metrics in order to discern different relevant and significant variables.

Stepwise regression technique commonly fits every regression model by dropping or adding some covariates one at a time, notably based on a particular criterion. Most commonly used Stepwise regression methods are standard stepwise regression model, backward elimination model, and forward selection model.

Standard stepwise regression models commonly do two important things, they remove and add predictors above all needed for every step. Forward selection regression starts with most relevant predictor in the models while adding a certain variable for every step.

Backward elimination regression starts when all predictors within the model are obtained. The further process removes the predictor with the least significant features for every step. The goal of this modeling technique is to improve and maximize the overall prediction power using a minimum number of available predictor variables. This method is commonly used when handling data sets with higher dimensionality.

Polynomial Regression

A regression equation, in fact, is a polynomial equation when the power of independent variables is higher than one.

Polynomial Regression Equation

$$Y = a + b * x ^ 2$$

In this particular regression technique, the best way to go is towards the best fit line when it is not a straight line. In this case, it is better than the best fit line is represented as a curve which fits into various data points.

Important things to remember when it comes to the polynomial regression is to always look out for a certain curve toward the ends and to see whether obtained shapes, as well as trend, make sense. Higher polynomials may end up producing some weird results when it comes to the extrapolation.

There might be a great temptation in order to fit those higher degree polynomials in order to get lower error rates, but this commonly my result in model over- fitting. You should always plot the connection in order to see the fit as well as to focus on making sure that obtained curves fit the represented curves naturally.

Ridge Regression

Ridge regression is a widely used technique in the situations when the data suffers from common multicollinearity meaning that independent variables are greatly correlated. When it comes to the multicollinearity, even when the least squares estimates are unbiased, their variances are increasingly high which leads to deviations of the observed value that commonly is far away from that true value. When you add a degree of bias to your regression estimates, the technique of ridge regression greatly reduced these standard errors.

Equation for Linear Regression

$$Y = a + b * x + e$$

Letter e represents the error above all the value particularly needed in order to correct that prediction error existing between a predicted value and the observed value.

Equation for Multiple Independent Values

$$Y = a + y = a + b1\ x1 + b2\ x2 + \dots + e$$

When it comes to the linear equation, these prediction errors may be decomposed into sub components. It is mainly due to the bias and due to the certain variance. Prediction errors may occur due to these components. Ridge regression techniques are used in order to solve this multicollinearity issues through lambda or shrinkage parameter. If you use lambda equation, you will have two main components. First one is lambda of certain summation or beta-square where beta represents coefficient. The second component is that least square term. Beta-square is further added to obtained least square to shrink the parameter to some other parameter that has very low variance.

Important things you should remember when it comes to the ridge regression are that the assumptions of this technique are the same as those in the least square regression with exception of normality. Ridge regression also shrinks the outcome value of all coefficients, but it does not reach value zero peculiarly suggesting that there is no selection feature involved. Ridge regression is an entire regularization method using L2 regularizations.

Elastic Net Regression

Elastic Net regression is a hybrid type of Ridge regression techniques. Using this technique, models are trained with L1 as well as L2 before as regularizers. Elastic Net regression techniques are very usefully when it comes to the models with multiple correlated features. A practical advantage of Elastic Net regression techniques include trading-off between Ridge and Lasso techniques remarkably allowing Elastic Net to obtain or inherit some features of that great Ridge’s stability mainly in terms of rotation.

The most important point of Elastic Net regression techniques you should remember are the facts that it greatly encourages various group effects especially in cases where there are highly correlated different variables. You should also keep in mind that there are no any limitations when it comes to the total number of previously selected variables. On the other hand, it may suffer from occasional double shrinkage.

Lasso Regression

Lasso regression is very similar to Ridge regression since it also penalizes the total size of all regression coefficients. In addition to obtaining the size of all regression coefficients, this technique is also capable of reducing the overall variability as well as improving the accuracy of all linear regression models. On the other hand, lasso regression differs from other regression uniquely ridge regressions mainly in a way that that the technique uses total values.

It uses penalty function rather than squares supremely used in ridge regressions. This, in fact, commonly leads to equivalently constraining or penalizing the sum of all absolute values of all estimate values that commonly causes some of the estimates value to turn to value of a zero. Important points you should remember are that the assumptions when it comes to the Lasso regression techniques are commonly same as those in least squares regression with an exception of normality. It also shrinks all coefficients to exactly zero that helps in overall feature selection.

Lasso regression method uses L1 regularization just like Ridge regression methods. In the case when you have a certain group of predictors that are highly correlated Lasso regression will pick one of them and further shrink all remaining predictors to the value of zero.

Ecological Regression

Ecological regression is kind of a statistical regression technique principally used when it comes to the history and political science in order to estimate a selection of voting behavior from obtained aggregate data.

For instance, if a country has a known vote of Democratic expressed in percentage and if the country knows a percentage of the population who are Catholics, data analytics run the ecological regression of certain dependent variable against the obtained independent variable. The result is the estimated or predicted Democratic vote. Ecological regression technique has been commonly used in litigation preeminently brought under certain Voting Rights Act from 1965 in order to see how whites and blacks vote.

Bayesian Linear Regression

Bayesian linear regression is widely used an approach in statistics. The approach is used to linear regression where statistical analysis is commonly undertaken within the wide context of Bayesian inference. In the case when a regression model has some errors which have a certain form of normal distributions and if a certain form of prior distribution is obtained, there are explicit results available for further posterior probability distributions of all models’ obtained parameters.

Bayesian linear regression is very common approach since it assumes that the number of total measurements is not enough in order to say anything meaningful about a model. In this approach, the data is supplemented with some additional information eminently as prior to a probability distribution. These prior beliefs about certain parameters are further combined with previously obtained data’s probability function. According to Bayes theorem, it yields various posterior beliefs about certain parameters. The prior, in fact, can take various functional forms depending on the information, which is available, and the domain.

Chapter 2 Predictive Modelling Process

Predictive analytics models are created by using various data, machine-learning techniques, and statistical algorithms in order to identify the probability of some future outcomes highly based on obtained historical data. The ultimate goal is to pass beyond some traditional descriptive statistical models and to report on what has occurred in order to provide the best possible assessment on what may occur in the future. The outcome is to streamline decisions strikingly making and producing some new insights, which lead to some better actions.

Predictive models commonly use obtained results in order to develop and train models which will be used in order to predict certain values for various new data. These modeling results represent a likelihood of some target variable like revenue acutely based on previous estimates from a collection of various input variables. This is greatly different from the various descriptive models which help us understand what occurred. These models also can help us with diagnostic models, so we better understand some key relations significantly helpful when it comes to the understanding why something occurred.

More and more companies and organization decide to turn to predictive modeling in order to increase their competitive advantage as well as to improve their bottom line. Growing volumes, as well as increasing types of data, raise this great interest in using data in order to produce various valuable information. There are also cheaper, faster as well as easier to use software greatly as a fertile ground for rapid growth in predictive modeling terminology. On the other hand, recent tougher economic conditions, as well as an increased need for greater competitive differentiation, also served as a fertile ground for various predictive modeling techniques.

Easy-to use, as well as interactive software, is becoming more and more prevalent, so predictive modeling has come out of its traditional domains of statistics and mathematics. A great number or business analytics, as well as other business experts, start to realize that using predictive modeling techniques can greatly improve their business management.

When it comes to the business process various software solutions lets you create different models and run one or multiple algorithms. The first step is to create a model. The following step is testing the model. Your model will be tested in terms of certain data collection.

In some scenarios, the model testing is done based on previously obtained data in order to see model predictions. The third step when it comes to the business process is validating the models. Validate the model will run certain results by using different visualization tools as well as profound business data understanding. Evaluating the model is the fourth step. You will be evaluating the best fit model based on previous models. You will also choose the model utterly right fitted for certain data.

The predictive modeling process involves running single or multiple algorithms within the data collection where that certain prediction will be carried out. This process, in fact, is an iterative processing so it commonly involves the step of training the model by using multiple different models on the same data collection. The process further goes when you create the best fit model principally based on the obtained business data understanding.

Predictive Modelling Steps:

1. *Pre-processing*
2. *Data Mining*
3. *Results Validation*
4. *Understanding Business and Data*
5. *Preparing Data*
6. *Modeling Data*
7. *Evaluation*
8. *Deployment*

When it comes to the types of models considered as predictive models, decision models, and descriptive models. Predictive models are used in order to analyze the past performance for some future predictions. On the other hand, descriptive models are able to quantify all elements of any decision to predict the outcomes of decisions that involve multiple variables. Decision models commonly describe the relationship existing between elements contained in a decision to predict the outcomes of various decisions that involve multiple variables.

Various algorithms are used in order to perform statistical analysis as well as data mining to determine various patterns and trends in data. The predictive modeling offers various solutions like time series, neural networks, regressions, outliners, k-means and other. Software solutions commonly available provide integration in order to open source R library.

When it comes to the Time series algorithms, they perform different time-based predictions. Examples include Double exponential smoothing and Exponential smoothing. When it comes to the regression algorithm, they predict different continuous variables intensely based on some other variables within a data collection. Examples include Exponential regression, Multiple linear regression, Geometric regression and Logarithmic regression. Association algorithms, on the other hand, are capable of finding the most frequent pattern in some large transactional data collections in order to generate various association rules. The example of an Association algorithm is Apriori.

Life Cycle of Predictive Model:

- *Process the Data*
- *Exploratory Data Analysis*
- *Build Model in Modelling Environment*
- *Deploy Model in Operational Systems with Scoring Application*
- *Refine Model*
- *Retire Model and Deploy Improved Model*

Clustering algorithms are able to cluster various observations into certain collections containing similar components, Example of clustering algorithms are Kohonen, K-means, and TwoStep. When it comes to the Decision tree algorithms, they firstly classify then predict single or multiple discrete variables eminently based on some other variables in the data collection. Examples of Decision trees algorithms are CNR Tree and C 4.5. Outliner detection algorithms have a capability of detecting various outlying values within any data collection. Examples of these algorithms include Nearest Neighbour Outlier and Inter Quartile Range.

Neural network algorithms commonly perform classification, forecasting as well as various statistical pattern recognition. Most common examples include MONMLP Neural Network and Net Neural Network. When it comes to Ensemble models, they are part of Monte Carlo analytics in which various numerical predictors are commonly conducted by using some slightly different conditions. Factor analysis, on the other hand, deals with the great variability that exists among observed and correlated variables when it comes to the potentially some lower number of those unobserved variables that we call factors. An example is Maximum likelihood algorithm.

When it comes to the Naive Bayes, they are common probabilistic classifiers especially based on Bayes' theorem with naive or strong independence assumptions. Support vector machines, on the other hand, are common supervised learning models immensely commonly associated with various learning algorithms, which analyze data as well as recognize different patterns. Support vector machine is commonly used for regression analysis and classification. Uplift modeling technique uses models profoundly having the incremental impact of treatment when it comes to the individual's behavior while survival analysis is used in order to determine and analyze time to different events.

When it comes to features in predictive modeling, those include data analysis and further data manipulation, data visualization, statistics and hypothesis testing. Tools used for data analysis create a completely new data collection. Data analysis tools further modify, categorize, club, merge and filter obtained data collections. Visualization feature includes various interactive graphics and relevant reports. On the other hand, statistics tools are used in order to create as well as to confirm an existing relationship between all variables within a data collection. Hypothesis testing is a creation of various models, evaluation and lastly choosing the most suitable model.

Steps To Effective Predictive Modeling

Every successful predictive modeling project is executed following certain steps. Prediction modeling is commonly used process for building certain models in order to predict some future outcomes by using various statistical techniques. On the other hand, historical data will be needed in order to generate any model. Some prior occurrences also need to be determined, classified as well as validated. The stage of predictive modeling process includes data gathering and data cleansing, data transformation or data analysis, building initial predictive model and inferences.

Before doing anything, you have to define your business objectives. Every project starts when a business objective is determined. The model which you will create is supposed to address to a certain business question. Your business objective will clearly allow you to define a certain scope of the project. It will also provide you with some exact testing in order to measure your project's success.

In order to train your model, you will use obtained historical data. This data is commonly scattered across some multiple sources, so it requires data preparation and data cleansing. Data also can contain some duplicate records and duplicate outliers. You will decide to remove or keep them in accordance with your business objective. Data also may have some missing values, so it may require transformation as well. Data may be used in order to generate various derived attributes, which have greater predictive power for your particular objective. Generally speaking, the overall quality of the data, in fact, initially indicates the quality of your model which will be created.

The next step involves sampling your data. You will have to split that obtained data into two data collections. The first data set is training and the second is testing dataset. You will build your model by using the first training dataset. You will use the test data in order to verify the overall accuracy of your model's outcome. Doing exactly this is crucial. If you perform this step differently, you will increase over-fitting risk. On the other hand, the test data highly ensures a certain valid way in order to accurately measure the overall performance of your model.

The next step is to build your model. It should be noted that sometimes the business objectives and the data lend themselves to a certain model or algorithm. On some other occasions, the best possible approach is not a clear-cut. While you are exploring the data, you should run as many different algorithms as possible and further compare the outcomes. You should base your choice on that final model when it comes to the overall results. Sometimes, it is better to run various ensemble models on the data simultaneously and then choose that final model while comparing models' outcomes.

After you have built your model, the next step is deploying your models. You have to deploy it to in order to reap all of its benefits. This process sometimes requires coordination with some other departments as well. You should make sure that you know how to suitably present your outcomes. After you have deployed your model, you should monitor its overall performance as well as continue to improve it. It should be noted that majority of models decay over some period of time. In other words, you should refresh your model and keep it up to date. Regularly upgrade it with some new data.

To summarize, you will read data from different sources and then perform data cleansing operation like identification of various noisy data as well as removal of different outliners in order to make your predictions more accurate. You will apply R packages in order to handle possible missing data as well as to impure certain values. Data transformation step requires a data that will be transformed for certain processing by normalizing that data with paying attention to the overall data significance. Normalization may be done by scaling certain values to a certain range. In addition to this, various not so relevant attributes may be removed if you perform a correlation analysis that will play not so significant role when it comes to the determining the results.

By building a predictive model, you will generate a decision tree or apply logistic or linear regression technique in order to achieve higher accuracy. This process involves choosing suitable classification algorithm as well as identifying various test data and then generating different classification rules. You will identify the confidence of your classification model as you apply it against different test data. Further, you will perform a cluster analysis in order to segregate different data collections. You will use meaningful subsets of obtained populations in order to make inferences.

After you have built classification predictive model, you can create more models with one difference. The differences are different parameters, so when creating your second model you will have to tune obtained parameters from one algorithm to another.

Predictive Modelling Steps :

1. *Formulation of Objectives*
2. *Review and Interpretation of Available Data*
3. *Model Conceptualization*
4. *Code Selection*
5. *Field Data Collection*
6. *Input Data Preparation*
7. *Calibration and Sensitivity Analysis*
8. *Predictive Runs*
9. *Uncertainty Analysis*

Creating Predictive Model with Logistic Regression

First, you have to load your data by following the code listing. Further, you have to create an instance of the certain classifier. When it comes to this step, two lines of the code listing import obtained logistic regression library. The second line of code listing creates a certain instance of your logistic regression algorithm. You will notice the regularization parameters that are used in order to prevent model over-fitting. On the other hand, these parameters are not strictly mandatory, since the constructors will work properly even without the parameters.

Loading the Data

```
from sklearn . datasets import load _ iris
```

```
iris = load _ iris ()
```

Creating an Instance of the Classifier

```
from sklearn import linear _ model
logClassifier = linear _ model . LogisticRegression (C=1, random _
state=111)
```

When creating a logistic regression classifier by using C=150, you will, in fact, create a better plot of the overall decision surface. Further, you will need to split the data collection into test sets and training sets and then you are able to create an instance of a certain logistic regression classifier.

Running the Training data

```
from sklearn import cross  _  validation
X _ train, X _ test, y _ train, y _ test = cross _ validation.train_test_split
(iris.data, iris.target, test _ size=0.10, random _ state=111)
logClassifier . fit (X _ train, y _ train)
```

The first line in the code listing imports the certain library exceedingly allowing you to split data collection into two sets. The second line calls the certain function form obtained library, which splits data collection into two parts as well as assigns the newly divided data collections two pairs of variables. The third line takes the instance of that logistic regression classifier that you have created. It also calls the fit method in order to train models within the training data collection.

The next step is to visualize the classifier. When you look at your decision surface area, you will think that looks like there is some tuning needed. Specifically, if you look near the middle parts of the plot, you will see that majority of the data point, which belongs to this area, in fact, are lying within the area located more to the right side. It will visually look much better when you use a suitable setting for your model. After you have visualized the classifiers, you have to run the test data. The first code listing displays test data collection. On the other hand, the third line represents the output.

Running the Test Data

```
predicted = log Classifier . predict (X _ test)
predictedarray ( [0, 0, 2, 2, 1, 0, 0, 2, 2, 1, 2, 0, 2, 2, 2] )
```

The following step is to evaluate your model. You may cross-reference the outcome from all predictions against certain y-test array. You can see that the model predicted all points within the test data correctly.

Evaluating the Model

```
from sklearn import metrics
predictedarray ( [0, 0, 2, 2, 1, 0, 0, 2, 2, 1, 2, 0, 2, 2, 2] )
y _ testarray ( [0, 0, 2, 2, 1, 0, 0, 2, 2, 1, 2, 0, 2, 2, 2] )
metrics.accuracy_score(y_test, predicted)1.0 # 1.0 is 100 percent accuracy
predicted == y _ testarray ( [ True, True, True, True, True, True, True, True,
True, True, True, True, True, True, True], dtype = bool )
```

Building Predictive Model Using Python

Given the rise of Python, especially in the last few years, we will build a predictive model with Python due to its simplicity. You should invest some quality time during initial phases like brain storming sessions, hypothesis negation and discussions to better understand and have a better insight into the domain. These activities will help you to better relate to issues, so eventually, you will be able to create more powerful and more efficient business solutions.

In other words, spend enough time working on understanding the issue. You will not be biased with data points as well as thoughts, so do hypothesis initially before you embark on your adventure. At some later stages, it is possible that you will be in a hurry in order to complete your project, so most likely you will not have enough time to invest it into successfully finishing your project,

You should follow your timeline, and make it your regular practice if you can. It will be of great help when it comes to the building efficient and accurate predictive models. Most certainly, these practices will lead to less iteration of work surpassingly needed at some later stages. Stages of building a model are descriptive analysis on obtained data, data treatment including outlier fixing and missing values, data modeling and estimation of model's performance.

Commonly, data preparation takes about fifty percent of the work when it comes to the building an effective predictive model. There are also enormous benefits of automation. Using machine learning tools, overall time needed to perform tasks has been greatly reduced. The time you might need to do overall descriptive analysis most likely is restricted to understanding big features and common missing values. Data exploration stage includes steps of identifying ID Input as well as Target features. It also includes identifying numerical and categorical features. The last step while doing data exploration is to identify certain columns with missing values.

Stage two is data treatment or missing values treatment. There are numerous ways to deal with issues at this stage. You should focus on quick as well as smart techniques in order to build your model. One method includes creating dummy flag for model's missing values. It will work since missing values commonly carry a great amount of information. Further, you will impute missing values with median or mean. Median and mean imputation commonly perform very well, so a majority of people prefer to work by imputing mean value. On the other hand, in a case when you work with a skewed distribution, it is a better way to go with a median. Further, you will impute some missing values of certain categorical value. You will create a new level in order to impute categorical variables and all missing values will be coded. You are able to look at frequency, impute and mix the missing values with other value prominently having a higher frequency. This simple method greatly reduces the time required to treat data.

The following step is data modeling. You can use Random Forest or GBM, depending on the problem you are going to solve. Both of these techniques are very effective when it comes to the creating useful benchmark solution. Majority of data scientists prefer to use these techniques commonly as their first model. The last stage is an estimation of performance. In order to validate the performance of your model, you can use various methods.

Among the best ways to go is to divide obtained train data collection into validating and train. You can build your model vitally based on seventy percent of train data collection. Further, you will cross-validate the collection using validate data set and then evaluate the overall performance by using evaluation metric. Now it is time to put all of this into action.

Import Libraries and Read Test

```
import pandas
import numpy
from sklearn . preprocessing import LabelEncoder
import random from sklearn . ensemble import RandomForestClassifier
from sklearn.ensemble import GradientBoostingClassifier
```

Train Data Set

```
Train = pd.read_csv ('C:/Users / Desktop /challenge /Train.csv' )
test = pd.read_csv ('C:/Users /Desktop /challenge /Test.csv' )
train ['Type'] = 'Train'
test ['Type'] = 'Test'
fullData = pd . concat ( [ train,test ], axis=0)
```

Summary of the Dataset

```
fullData . columns
fullData . head (10)
fullData . describe ()
```

Identify Target Variables, Numerical Variables, ID Variables and Categorical Variables

```
ID_col = [ 'REF_NO' ]

Target _ col = [ "Account.Status" ] cat _ cols =

['children','age_band','status','occupation','occupation_partner','home_status'
'self_employed_partner','year_last_moved','TVarea','post_code','post_area','g

num_cols = list (set (list (fullData.columns))-s

et (cat _ cols)-set (ID _ col)-set(target _ col)-set (data _ col)) other _ col=
['Type']
```

Identify the Variables with Certain Missing Values

```
fullData . isnull () . any

num _ cat _ cols = num _ cols+ca t_ cols num _ cat _ cols

fullData [var]. isnull ().any()= =True:

fullData [var+'_NA']= fullData [var] . isnull()*1
```

Impute Missing Variables

```
fullData  [num _ cols] =
fullData[num_cols].fillna(fullData[num_cols].mean(),inplace=True)
```

Imput Categorical Values

```
fullData [cat _ cols] = fullData [cat _ cols] . fillna (value = -9999)
```

Create a Label Encoder

```
number = LabelEncoder () fullData [var] = number . fit
_transform(fullData[var] . astype('str'))

fullData [ "Account.Status" ] =

number . fit  _ transform ( fullData["Account . Status" ] . astype( 'str'))

train =  fullData [ fullData ['Type']= ='Train']

test =  fullData [ fullData ['Type']=='Test'] train ['is _ train'] =
np.random . uniform (0, 1, len (train)) <= .75 Train, Validate =
train [train ['is _ train']==True], train [train ['is _ train']==False]
```

The following step is to pass dummy and imputed variables into the overall model process. You can use random forest in order to predict the class.

```
Features = list (set (list ( fullData.columns ) ) – set (ID _ col)-set (target _ col)-set (other _ col ))

X _ train = Train [ list(features)].values
Y _ train = Train [ "Account.Status"].values
X _ validate = Validate[ list(features)].values
Y _ validate = Validate[ "Account.Status"].values
X _ test = test [ list(features)].values
random . seed (100) rf = RandomForestClassifier ( n _ estimators=1000)
rf . fit (x _ train, y _ train)
```

Check Model’s Performance

```
status = rf . predict _ proba  (x _ validate)
fpr, tpr, _ = roc _ curve (y _ validate, status[:,1])
roc _ auc = auc ( fpr, tpr )
print roc _ auc
final _ status = rf . predict _ proba (x _ test)
test  ["Account.Status"] = final _ status [:,1]
test . to _ csv('C:/Users/Desktop/model _ output . csv', columns = [ 'REF_NO','Account.Status' ])
```

Chapter 3 Enhancing a Model Performance

When it comes to the imprecision in any predictive model, variance and bias are two main components which need to be improved. Generally speaking, there is a type of trade-off between these two components. In other words, by reducing one of them, you are at the same time increasing the other component. Bias in any predictive model, in fact, is a certain measure of model's inflexibility and model's rigidity. It means that the model is not able to capture all of the data's single. Bias is widely known as under-fritting. On the other hand, a variance is a certain measure of every model's inconsistency. Any high variance model tends to perform well when it comes to some data point while performing badly on some other points. This is known as over-fitting. It means that the model is too flexible to cope with the amount of data and ends up picking various noise in addition to useful signals.

There are various ways when it comes to the determining if a model suffering from more variance or more bias. If a model performs well in terms of training collection, but poorer when it comes to the hold-out-set, it is very likely suffering from the great variance. If a model performs poorly on both test data collection and training set, then it is suffering from very high bias. Depending on your model and whether if it is coping with high variance or high bias, you are able to resort to a technique in order to bring your predictive model where you want it to be. You can add more data, add more features, do feature selection, use regularization, boost, and bootstrap aggregation and use various classes of models.

If you add more data, it will reduce variance. Adding more data will also allow you to use greater flexibility models as well. Adding more features is always a great idea if affordable. If you add more features, you will increase overall model's flexibility as well as decrease bias, but on the expense of variance. There are times when adding more features is not so great idea like when your data collection is small regarding its data points. In this case, you can invest your time in adding more data, which is more appropriate when you have poor data points.

Another great idea is to perform feature selection. It should be noted that you do feature selection when you have already added a lot of features and when you have not that great number of data points. Feature selection process is almost inverse to adding more features. It will pull your models in the completely opposite direction by decreasing variance, but at the expense of bias. However, a trade-off may do well, in the case when you do feature selection carefully and methodologically and when you only remove uninformative and noisy features. In the case when you have enough data, predictive models are able to automatically cope with uninformative and noisy features, which mean that you don't have to explicit and perform feature selection. It should be noted that in recent years in this age of Big Data, this need for performing feature selection is rarely needed. However, feature selection is computationally intensive and non-trivial, so when you need to improve features of your model, do feature selection.

In order to improve your predictive model, you should use regularization. Using regularization is, in fact, neater version of performing feature selection. Regularization is capable of telling your model to try and use new features whenever possible, and not to trust and depend on any single feature. Regularization, in fact, relies on some smart implementation of various training algorithms. These are a common as well as a widely preferred version of previous feature selection.

Short for Bootstrap Aggregation is bagging. This process uses multiple versions of the exact same model significantly trained on different samples in terms of training data in order to reduce variance, but at the same time not to affect bias. Bootstrap aggregation may be computationally very intensive when it comes to the memory. On the other hand, you can use boosting as well, but it is more complicated concept highly relying on training multiple models and trying to learn from various models' errors. The process of boosting decreases bias and slightly affect variance. In this case, the price is memory size as well as computation time.

Another great option when it comes to the improving your predictive model is to use more different classes of models. You don't have to use all techniques from the above since there must a technique that is just right for your certain model. However, you are able to change the model class such as changing linear models to some neural network model. This will move your model to a completely different point in terms of the space. It should be noted that certain algorithms are simply better to particular data collections that some other algorithms. It is important to identify the model on which you can rely. Model accuracy, in fact, is not the only objective. It should be noted that some of the highly accurate models may be very hard to deploy. At the same time, there might be black boxes which are very hard to integrate or debug. Therefore, a majority of production systems search for less accurate and simpler models, notably those less resource-intensive models hugely easier to debug and deploy.

Improving the Accuracy of Your Predictive Model:

- *Add More Data*
- *Treat Outlier and Missing Values*
- *Feature Engineering*
- *Feature Selection*
- *Multiple Algorithms*
- *Algorithm Tuning*
- *Ensemble Methods*
- *Cross Validation*

Enhancing the overall performance of your model may be challenging, but we all have been in that situation when a model needs to be improved. It may happen that after you have tried different strategies and different algorithms, but your model still is in not as accurate as it is supposed to be. At this point, you may feel stuck and helpless, but there is no reason for giving up. These ways from the above will most certainly help you to improve the performance of your model. However there is no rule that you must follow, but you may follow these simple rules from the above, so you will certainly achieve that much desired high accuracy of your predictive model. You will combine the rules depending on your model. The model development process goes through multiple stages, beginning from dataset to model building. Before you start exploring the data, you have to understand existing variables relationships and it is always highly recommended to do hypothesis generation in order to learn more about model development. Hypothesis generation may be, in fact, the most under-rated step when it comes to the predictive modeling. It is very important that you invest your time into thinking about modeling problem as well as gaining the necessary domain knowledge. It can help you significantly when it comes to the building better features, so it is a crucial step just like every other step since it greatly can help you to improve your model's accuracy.

Tuning the Parameters

Parameters in any predictive model are used in order to increase the overall predictive power of predictive models as well as well to improve the model for further training data process. When it comes to the features extremely improving model's accuracy most commonly used features are n_estimators, and min_sample_leaf. There is a certain number of features which model is allowed to use when it comes to the individual predictive model. If you are using Python, you will see that there are various features in order to assign a maximum number of features your model will be allowed to use. You can use some of the following:

- *Auto/None*: This option is very simple since it will take all the features that make sense within a predictive model. If you use this feature, you will not put any obstacles and restriction on your predictive model.
- *sqrt*: This option is used when you want to take square root of the overall number of features in a single run.
- *0.2*: This option will allow your predictive model to take twenty percent of variables in a single run.

You may wonder how does these max feature impact the overall speed and performance of your predictive model. These features improve the performance of every predictive model since at every node you will have a greater number of available options, which you may consider to further use.

On the other hand, this is not always the case since these features also decrease the great diversity within a predictive model. On the other hand, you also decrease the overall speed of algorithm since you increase maximum features. You will have to find the right balance as well as to choose best suitable maximum feature for you predictive model. Better performance of your model also can be achieved if you use other parameters like n_estimators and min_sample_leaf. There are also features which will make model training much easier like n_jobs, random_state, and oob_score. These parameters are able to tell the model engine how many processors it can use. If there is a value -1 it means that there are no any restrictions. On the other hand, value 1 means that the model is able to use only a single processor. You can check this metric in Python.

```
%timeit

model = RandomForestRegressor (n _ estimator = 100, oob _ score =
TRUE,n_jobs = 1,random _ state =1 ) model . fit (X , y)

%timeit

model = RandomForestRegressor (n _ estimator = 100,oob _ score =
TRUE,n_jobs = -1,random _ state =1 ) model . fit (X , y)
```

This function “%timeit” is very useful since it runs a single function multiple times as well as it gives the fastest loop running time. This is very handy when it comes to the scaling up a certain function from some prototype to final data collection. Parameter random_state and oob_score makes solutions very easy to replicate. It should be noted that definite value of these parameters always produced the exact same results when you have given same training data and same parameters.Another parameter oob_score is random cross validation technique. In fact, this technique is very similar to other validation techniques, but the process is much quicker. Using this method you can simply tag each observation distinctly used in different models. Further, you will be able to find out that maximum vote score for each observation very much based on some models that didn't use this certain observation in order to train itself. You can use all of these parameters in one function.

```
model = RandomForestRegressor (n _ estimator = 100, oob _ score =
TRUE, n_jobs = -1, random _ state = 50
max _ features = "auto", min _ samples_leaf = 50 )
model . fit ( X,y )
```

Chapter 4 The Performance of the Prediction Models

When it comes to the predictive modeling, one of the most crucial steps is data preprocessing. It is critical that you clean the data before you continue problem-solving. Even in the case when you have good data, you will have to make sure that the data is in a certain useful scale and format.

You also have to make sure that even the least meaningful features are included within the data. If you are more disciplined when it comes to the handling your data, you will have better as well as more consistent results and better accuracy.

Select Data

Preprocess data

Transform Data

The first step it so selects data. This step is mainly concerned with selecting a relevant subset in terms of all available data on that you will be working with. You have to keep in mind what data is suitable for addressing your certain problem and what data will be proper for solving your problem. You should make more assumptions and be careful when it comes to the recording these assumptions so you can test them properly in later stages.

The following step is data preprocessing. After you have selected your data, you will need to consider in which manner you will use obtained data. The data preprocess is all about getting that selected data into a certain form on which you will work further. The most common steps when it comes to the data preprocessing are formatting, cleaning, and sampling.

- Data Formatting: The data which you have selected sometimes may be in that desired format and in the format prominently proper for further works. The data also may be in some relational database and you will like the data to be in a flat file. Data formatting is a process where you change data format into a suitable one depending on which problem you are working on.

- Data Cleaning: Cleaning data is a process of removing as well as fixing some missing data. There might be some instances incomplete. You may work with data which might not be suitable for your problem. When it comes to the various instances, those can be removed by data cleaning process. There also may be some sensitive information in certain attributes. Sometimes these attributes should be removed or anonymized from the data completely.

- Data Sampling: In some cases, there is an enormous amount of data, which you have to work with. In this case, more data may result in much longer overall running times. It also leads to greater memory requirements as well as larger computations requirements. By data sampling, you are able to take some smaller data representatives of your selected data exceptionally much faster way for prototyping and exploring solutions before you consider exploring the entire dataset.

The following step is data transformation. As soon as you preprocess data, you are able to transform it. This step is mainly influenced by certain algorithm you are working on and by the knowledge of the domain. In a great majority of cases, you will have to revisit over and over transformations of preprocessed data while you are working on your problem. The common data transformations steps are scaling, decomposition and aggregation.

- Data Scaling: The preprocessed data occasionally contain various attributes with some mixtures of scales for different quantities like sales volume, kilograms, and dollars. Many predictive modeling techniques like when data attributes have the exact same scale between values 0 and 1 for both the largest and the smallest given feature.

- Data Decomposition: Occasionally there might be some features reputably representing some complex concept which may be way more useful to predictive modeling when it is split up into the relevant constituent parts. An example of data splitting or data decompositions is a date which may be split into day and time elements which also may be split further.

- Data Aggregation: Commonly there are features which may be aggregated into some single feature which can be more useful and meaningful to your problem. In order to use the most useful features, you should perform data aggregation and combine useful features into a single feature.

Handling Class Imbalance

Imbalanced classes are able to put accuracy simply out of business. This problem is very common incomparably when it comes to the data classifications since it commonly occurs in data collection outstandingly containing disproportionate ratios of certain observations in every class. It is the fact that standard accuracy can no longer reliably measure performance, so predictive model's training tends to be much trickier.

Some of the most common imbalanced classes appear in many domains like advertising click-throughs, spam filtering, fraud detection, disease screening and many other. It is more than apparent that knowledge on how to cope with imbalanced classes is almost mandatory when it comes to the predictive modeling.

It is a great idea, to begin with common synthetic dataset Balance Scale Data that you can download. The dataset was originally generated by those models specially used in psychological experiments. However, it is very useful for predictive modeling as well since it is an entirely manageable size and contains a great number of imbalanced classes.

Import Libraries

```
import pandas

import numpy
```

```
pd.read_csv ( 'balance-scale.data', names = ['balance', 'var1', 'var2', 'var3',
'var4' ] )
```

```
df . head ()
```

The dataset contains some information when it comes to the scale and whether it is balanced or not. This information is based on distance and weight.

Count Every Class

```
Df [ 'balance '] . value  _  counts ()
```

Transform into Binary Classification

```
Df [ 'balance' ] = [1 if b= ='B' else 0 for b in df . balance ] df
['balance'].value_counts ()
```

You can see what is the percentage of observations which are balanced. When you have a dataset, you are able to show the real dangers when it comes to the imbalanced data. You will import the accuracy metric and Logistic Regression algorithm. You may use Scikit-Learn and download needed resources from there.

```
om sklearn . linear _ model import LogisticRegression from sklearn . metrics
import accuracy _ score
```

Train Model

```
y = df . balance
X = df . drop ( 'balance', axis=1 )
Clf _ 0 = LogisticRegression () . fit ( X, y )
Pred _  y _ 0 = clf _ 0 . predict ( X )
```

Confirm

```
Print ( accuracy _ score ( pred _ y _ 0 , y ) )
```

Causes of Poor Model Performance

There are some genuine good predictive models, but they still may be performing poorly. In order to evaluate these issues, you should assess metrics of recall, precision, bias, and variance since these metrics are main reasons why a predictive model is performing poorly. Bias and variance also commonly causes for model over-fitting. There are several models you can choose. For instance, you can use Logistic Regression in order to predict value as well as to classify some distinct outcomes. You can also use Neural Networks in order to model some nonlinear behaviors. When you build models, you firstly obtain relevant historical data, which helps your models to learn what it the certain relationship between input features and some predicted output. However, even in the case when the models are able to accurately predict certain values from obtained historical data, we still can't be sure that it will work accurately on some new data. We have to make sure that our model is actually performing as anticipated.

Seemingly good predictive models still may be poorly performing. In this section of the book, you will see how to evaluate these common issues when it comes to the model performance by assessing common metrics of recall, precision, bias, and variance, which cause an issue in model performance. When it comes to the evaluating a predictive model, the first thing you should do is to assess whether your model has a high variance or high bias. High bias commonly refers to a model in which overfitting is common. Over-fitting is extremely bad for model performance since your model is not able to accurately present relationship between predicted output and your inputs. These models also often tend to output high errors like that great difference between model's predicted value and the actual value.

On the other hand, high variance represents the entirely different situation when your model is so accurate so it is fitted to your data collection perfectly. It may seem like it is a good thing since the model perfectly fit the dataset, but it is a cause for great concern since these models commonly tend to fail when it comes to the generalizing your future datasets. In this case, while your model works great for your already existing data, you can't be sure that it will remain to work great when it comes to some other examples.

In order to determine whether your model has a high variance or high bias, you will use a straightforward method data splitting. By data splitting your data will be divided into two counterparts, training, and test datasets. In the case when your model has errors in both of these datasets, you will know that the problem is under.fitting. In the case when your model has a lower error in the training data collection, but at the same time it has a higher error in the test data collection, this is the biggest indicator of a model with high variance since your model has failed when it comes to the generalizing to that second collection of data. You are able to generate a model that will have an overall low error when it comes to the both test and train datasets. This model, in fact, will have balanced levels of both variance and bias.

Low precision, as well as low recall, also cause poor model performance. In the case when you have high accuracy, it is still possible that your model may be greatly susceptible to several types of errors. In the cases like this, it is helpful to look at a certain percentage of all positive classes that you are predicting. These are given by two common metrics Recall and Precision.

Precision is a metric that measures how often predictor in terms of a positive class is true. The metrics calculate the overall number of all true positives over the total number of false positives and true positives. On the other hand, recall is a metric primarly measuring how often certain actual positive classes are predicted in such manner. The metrics are calculated as the overall number of true positives divided by the total number of all false negatives and true positives. You can use precision metrics if you want to measure what fraction of prediction for the positive classes are in fact valid. On the other hand, a recall will tell you how often predictions capture the positive classes. Therefore, low precision emerges in the cases when there is a low number of positive predictions while low recall happens when positive values are not predicted.

Conclusion

Predictive modeling techniques will help when it comes to the leveraging the true power of the predictive analytics. This technique also can help you to save your time, simplify analysis as well as boost productivity. Predictive modeling uses mostly statistics in order to predict outcomes. Techniques may be applied in order to predict the future of various unknown events not depending on the exact time when those occur. Predictive modeling is very useful in various domains like customer relationship management, algorithmic trading, archaeology, uplift modeling auto insurance, health care and many other, so there is no surprise in the fact that predictive modeling technique has brought some major changes to various fields.

Predictive modeling is also synonymous with the field of machine learning wince both fields encompass a variety of different statistical techniques in addition to data mining which analyzes both current and historical facts in order to make predictions about future and various unknown events. In business, a majority of predictive models exploit certain pattern mainly found in transactional as well as in historical data in order to identify common risks and great opportunities. Models are also able to capture different relationships among many different factors in order to allow assessment of potential or risk specifically associated with a certain set of guiding decision making and conditions for various candidate transactions.

It is more than apparent that there is an enormous significance of predictive modeling in various fields, and simply it is almost impossible to imagine that we don't know and don't understand the power of predictive modeling. This book will be your best companion on this journey regardless of which predictive modeling technique you will use. It will most certainly help you to simplify the analysis, boost your productivity as well as save you some time.

Congratulations!

You finished reading this book! If you enjoyed reading this book, I would like to ask you to leave a review to support me.

Simply click on the link and it will take you directly to the book: Applied

Predictive Modeling

Thank you very much and have a great day!