CHAPTER 6

SOFTWARE PERFORMANCE PREDICTION PROCESS USING
REGRESSION ANALYSIS

6.1 INTRODUCTION

In software engineering, the systems are constructed by integrating many reusable components which are known as features. Adding a feature in to a software system will increase the functionality of the system. This methodology will require less time to build a system and decreased time for marketing, improved quality of the product, and the products profile can be diversified. But still the success of the compositional approach mainly relies on the modularity of the reusable components. This chapter explains the process of regression analysis for software performance prediction and the need for modularity is made essential right from the beginning of the software engineering era, and also now in the new technological fields like synthetic biology.

6.2 REGRESSION ANALYSIS

In statistical modeling, regression analysis is a measurable procedure for evaluating the connections among factors. It incorporates numerous systems for demonstrating and breaking down a few factors, when the attention is on the relationship between a reliant variable and at least one independent factor (or 'indicators'). All the more particularly, relapse investigation helps one see how the commonplace estimation of the reliant variable (or 'paradigm variable') changes when any of the free factors is differed, while the other independent factors are held settled. Most regularly, regression analysis evaluates the contingent desire of the dependent variable given the
independent factors – that is, the normal estimation of the needy variable when the independent factors are settled. Less normally, the attention is on a quantile, or other area parameter of the contingent dissemination of the reliant variable given the independent factors. In all cases, the estimation target is a component of the independent factors called the regression work. In regression analysis, it is additionally important to portray the variety of the needy variable around the regression work which can be depicted by a likelihood dissemination.

In a software system made from re-usable components called features, the interaction between the features occurs when the behavior of one feature influences or disturbs the behavior of another feature. In order to develop a highly-secured system the developer must analyze all the possible outcomes, both the positive and negative outcomes, of the feature interaction (Clande Knaus 2008). In the case feature interaction problem, the fact is that it is not possible to isolate a feature. It communicates and co-operates with other features. But it is possible to detect or predict desired and undesired feature interactions in a scalable fashion.

The high dynamic quality of the features in self-adaptive systems, cloud computing, and dynamic product lines exhibits a new challenge to find a solution for feature interaction problem (Muffy Calder et al. 2003). It is noticed that even in internet applications, requirements engineering, transport systems, computational sciences and many other fields which does not come under the computer science field it is hard to solve feature interaction problem. There cannot be a common solution for feature interaction problem prevailing in many of the systems; the solution for a system will differ entirely from the solution for another system. At present many systems are
providing options to the user to select the needed features based on the application scenario. Rather than focusing on the conventional requirements the users pay attention to the requirements like minimal energy consumption and max response time. To meet the non-functional requirements a prior knowledge on which combination of features will decrease the performance of the system is essential. But it is infeasible to acquire knowledge about all combination of the features. So, it will be better to identify the set of features whose combination doesn’t degrade the performance of the system.

In this part of research, the focus is about predicting the performance of the system in which the set of features is selected by the user. The prediction is made based on the previous recorded performance of the system for different combination of the features (Simonetta Balsamo et al. 2004). This prediction will help users to identify the set of features whose combination gives maximum performance.

6.3 PROPOSED REGRESSION ANALYSIS MODEL FOR SOFTWARE SYSTEM EVALUATION

The following is the proposed Random Forest based regression analysis model and the algorithm to implement this model is also discussed along with the performance analysis.

6.3.1 Random Forest-based Regression Analysis

A related yet particular approach is Necessary Condition Analysis (NCA), which assesses the most extreme (as opposed to normal) estimation of the dependent variable for a given estimation of the independent factor (roof line instead of focal line) keeping...
in mind the end goal to distinguish what estimation of the autonomous variable is important yet not adequate for a given estimation of the needy variable.

Regression Analysis is generally utilized for expectation and anticipating, where its utilization has significant cover with the field of machine learning. Regression analysis is additionally used to comprehend which among the independent factors are identified with the needy variable, and to investigate the types of these connections. In limited conditions, regression investigation can be utilized to deduce causal connections between the autonomous and ward factors. However, this can prompt to hallucinations or false connections, so alert is prudent; for instance, relationship does not infer causation.

Numerous systems for completing regression analysis have been created. Well known techniques, for example, linear regression and ordinary least squares regression are parametric, in that the relapse capacity is characterized as far as a limited number of obscure parameters that are assessed from the information. Nonparametric relapse alludes to methods that permit the relapse capacity to lie in a predefined set of capacities, which might be unending dimensional.

The execution of regression analysis techniques practically speaking relies on upon the type of the information producing procedure, and how it identifies with the relapse approach being utilized. Since the genuine type of the information producing procedure is for the most part not known, relapse examination regularly depends to some degree on making suspicions about this procedure. These presumptions are at times testable if an adequate amount of information is accessible. Relapse models for
forecast are regularly helpful notwithstanding when the suspicions are reasonably
damaged, despite the fact that they may not perform ideally. In any case, in numerous
applications, particularly with little impacts or inquiries of causality in light of
observational information, regression techniques can give deluding comes about.

In a smaller sense, regression may allude particularly to the estimation of consistent
reaction factors, instead of the discrete reaction factors utilized as a part of grouping.
The instance of a ceaseless yield variable might be all the more particularly alluded to
as metric relapse to recognize it from related issues.

Regression models involve the following variables:

The unknown parameters, denoted as $\beta$, which may represent a scalar or a vector.

The independent variable $X$.

The dependent variable $Y$.

In various fields of application, different terminologies are used in place of dependent
and independent variables. A regression model relates $Y$ to a function of $X$ and $\beta$ is
given in Equation (6.1),

$$Y \approx f(X, \beta) \quad (6.1)$$

The approximation is usually formalized as $E(Y|X) = f(X, \beta)$. To carry out regression
analysis, the form of the function $f$ must be specified. Sometimes the form of this
function is based on knowledge about the relationship between $Y$ and $X$ that does not
rely on the data. If no such knowledge is available, a flexible or convenient form for $f$
is chosen.
Assume now that the vector of unknown parameters $\boldsymbol{\beta}$ is of length $k$. In order to perform a regression analysis, the user must provide information about the dependent variable $Y$.

If $N$ data points of the form $(Y, X)$ are observed, where $N < k$, most classical approaches to regression analysis cannot be performed: since the system of equations defining the regression model is under determined, there are not enough data to recover $\boldsymbol{\beta}$.

If exactly $N = k$ data points are observed, and the function $f$ is linear, the equations $Y = f(X, \boldsymbol{\beta})$ can be solved exactly rather than approximately. This reduces to solving a set of $N$ equations with $N$ unknowns (the elements of $\boldsymbol{\beta}$), which has a unique solution as long as the $X$ are linearly independent. If $f$ is nonlinear, a solution may not exist, or many solutions may exist.

The most common situation is where $N > k$ data points are observed. In this case, there is enough information in the data to estimate a unique value for $\boldsymbol{\beta}$ that best fits the data in some sense, and the regression model when applied to the data can be viewed as an over determined system in $\boldsymbol{\beta}$.

In the last case, the regression analysis provides the tools for:

Finding a solution for unknown parameters $\boldsymbol{\beta}$ that will minimize the distance between the measured and predicted values of the dependent variable $Y$ (also known as method of least squares). Under certain statistical assumptions, the regression analysis uses the
surplus of information to provide statistical information about the unknown parameters $\beta$ and predicted values of the dependent variable $Y$.

### 6.3.2 Algorithm for Random Forest Model

Decision trees are considered to be one of the most important machine learning algorithms, because its performance is not degraded even when the feature values are scaled or transformed. The accuracy of the trained model is not affected when irrelevant features are added. Especially the trees that are expanded deep have the ability to learn highly irregular patterns (Antonio Criminisi et al. 2012). Because of low bias and very high variance generally they over fit their training sets. Random forest is a way of averaging multiple decision trees. The individual trees are trained with different parts of the same training set, aimed at reducing the variance. This is achieved normally by increasing the bias and losing interpretability to certain limit. Random forest is an ensemble based machine learning algorithm can also be used for regression analysis. It operates by forming a multitude of decision trees at the time of training. Then the model output the mean prediction of the individual trees (Anne-Laure Boulesteix et al. 2012). The schematic view of Random Forest Analysis is shown in Fig. 6.1.

The training algorithm for random forest utilizes the bootstrap aggregating technique or bagging for learning. Let us considered the training set $X = x_1, ..., x_n$ which has a response $Y = y_1, ..., y_n$. The bagging is done repeatedly on the data set for $B$ times to select a random sample with replacement of the training set and to fit trees to these samples.

For $b = 1, ..., B$: 
1. Sample, with replacement, \( n \) training examples from \( X, Y \); call these \( X_b, Y_b \).

2. Train a decision or regression tree \( f_b \) on \( X_b, Y_b \).

Once training is completed the model can be used to predict the response for test samples, \( x' \) by averaging the predictions of the individual regression trees on \( x' \) or by taking the majority vote in the case of decision trees is given in Equation (6.2),

\[
\hat{f} = \frac{1}{B} \sum_{b=1}^{B} \hat{f}_b(x')
\]  

(6.2)

The bootstrapping method of learning increases the performance of the model by reducing the variance of the model. But at the same time the bias is not increased. Training many trees with a single training set will generate strongly correlated trees. Sometimes the same tree might be produced when the training algorithm is deterministic in nature.

Fig. 6.1 Schematic View of Random Forest Analysis
When the samples are bootstrapped to produce different training sets, the trees will be de-correlated. The value of B, the number of samples/trees is decided based on the size and nature of the training set. To find the optimal value of B, cross-validation method is used. In some of the application the value of B is fixed by verifying the out-of-bag error.

Traditional assumptions for regression analysis include:

- The training samples are illustrative of the populace for the surmising prediction.
- The error is an irregular variable with a mean of zero contingent on the illustrative factors.
- The independent variables are measured with no error. (Note: If this is not really, displaying might be done rather utilizing mistakes as a part of factors model procedures).
- The independent variables (indicators) are directly free, i.e. it is unrealistic to express any indicator as a direct blend of the others.
- The errors are uncorrelated, that is, the variance–covariance network of the errors is corner to corner and each non-zero component is the difference of the mistake.
- The change of the error is consistent crosswise over perceptions (homoscedasticity). If not, weighted slightest squares or different techniques may rather be utilized.

These are adequate conditions for the minimum squares estimator to have alluring properties; specifically, these suppositions infer that the parameter evaluations will be impartial, predictable, and productive in the class of direct unprejudiced estimators. It is essential to note that real information once in a while fulfills the suspicions. That is,
the technique is utilized despite the fact that the presumptions are not valid. Variety from the suppositions can now and then be utilized as a measure of how far the model is from being valuable. A large portion of these suspicions might be casual in more propelled medications. Reports of statistical analysis often include examinations of tests on the sample data and system for the fit and convenience of the model.

Assumptions incorporate the geometrical support of the factors. Independent and subordinate factors frequently allude to values measured at point areas. There might be spatial patterns and spatial autocorrelation in the factors that damage factual suppositions of regression. Geographic weighted regression is one method to manage such information. Likewise, factors may incorporate qualities amassed by ranges. With amassed information, the modifiable areal unit issue can bring about extraordinary variety in relapse parameters. While breaking down information accumulated by political limits, postal codes or enumeration territories results might be exceptionally unmistakable with an alternate selection of units.

Regression models predict an estimation of the \( Y \) variable, given known estimations of the \( X \) factors. Prediction inside the scope of qualities in the dataset utilized for model-fitting is referred to casually as interpolation. Prediction outside this scope of the information is known as extrapolation. Performing extrapolation depends emphatically on the regression suppositions. The further the extrapolation goes outside the information, the more space there is for the model to flop because of contrasts between the presumptions and the specimen information or the true values.
It is by and large advised that when performing extrapolation, one ought to go with the assessed estimation of the needy variable with an expectation interim that speaks to the vulnerability. Such interims have a tendency to extend quickly as the estimations of the free variable(s) moved outside the range secured by the watched information.

For such reasons and others, exactly tend to state that it may be impulsive to attempt extrapolation. Notwithstanding, this does not cover the full arrangement of demonstrating blunders that might be made: specifically, the presumption of a specific frame for the connection amongst \( Y \) and \( X \). A legitimately directed regression analysis will incorporate an evaluation of how well the expected shape is coordinated by the watched information, however it can just do as such inside the scope of estimations of the free factors really accessible. This implies any extrapolation is especially dependent on the suppositions being made about the auxiliary type of the relapse relationship. Best-hone exhortation here is that a direct in-factors and straight in-parameters relationship ought not be picked basically for computational comfort, but rather that all accessible learning ought to be sent in developing a relapse demonstrate.

In the event that this learning incorporates the way that the needy variable can't go outside a specific scope of qualities, this can be made utilization of in selecting the model – regardless of the possibility that the watched dataset has no qualities especially close such limits. The ramifications of this progression of picking a fitting useful shape for the relapse can be awesome when extrapolation is considered. At least, it can guarantee that any extrapolation emerging from a fitted model is "reasonable" (or as per what is known).
6.4 EXPERIMENTS AND RESULTS

Initially the cross-validation algorithm is applied to split the dataset into training and testing. The cross-validation is used for the estimation of the level of fit of a model to a dataset that is independent of the data that were used to train the model.

6.4.1 Dataset

The focus in this research work is essentially on the configurable systems, which gives the users an option to select features based on their requirement. But because of the feature interaction problem, for a selected list of features the performance of the corresponding software may be reduced. To prevent these difficulties this work concentrates on developing a system which will predict the performance of the system based on a regression analysis using the Random Forest algorithm. Based on the predicted performance the user may change his preferred list of features. For our experiments and analysis out of the six real-world configurable systems namely Berkley DataBase (C), Berkley DataBase (Java), Apache, SQLite, LLVM, and x264, the SQLite is considered. The dataset is benchmark dataset generated by (Jianmei Guo et al. 2013) and it contains performance measurement for different configuration of the system. In each of the system the performance has been measured by utilizing a conventional benchmark factor provided by the vendor of the software or used widely in the application domain. In this research work we have focused only on industrial programs rather than self-developed programs.

6.4.2 Repeated Random Sub Sampling Validation

The validation is implemented by following the steps below:
1. Assign each observation to any one of the groups namely training and validation randomly.

2. Apply the training set to the model and generate a model which classifies or predicts the class or response variable for a given test sample.

3. Using the samples from the test or validation dataset check the performance of the model and store this information.

4. Repeat steps 1 to 3 many times.

This method is also known as Monte Carlo cross-validation, which splits the dataset into training and validation randomly. For each split, the model is fit to the training data and the predictive accuracy is measured using the test set.

The results are then averaged over the splits. The advantage of this method over the k-fold cross validation is that the percentage of the split between the training and validation does not depend on the number of iteration. When this method of validation is adopted the results may vary when the analysis is repeated with random splits. This kind of variation in the results is also termed as Monte Carlo variation.

6.4.3 Building Random Forest Model

After dividing the dataset in to training and testing set using the random sub-sampling validation method, the random forest regression model is built with 500 decision trees. Random Forest does not require split sampling method to estimate the accuracy of the built model. Self-testing possible even if all data is used for training as 2-3rd of available training data is used to grow any one tree and the remaining one-third portion of training data always used to calculate out of bag error to assess model performance.
The Out of bag error (Jan Kodovsky et al. 2012) is calculated for random forests during training, so there is no need for a separate test set to validate result. The error is calculated internally during the training time. While the forest is built using the training data, each decision tree is tested with the 1/3rd of the samples (36.8%) which are not utilized for the training the tree. The out of bag error estimate can be considered as an internal error estimate of a random forest while it is being constructed. The Fig. 6.2 shows the plot of the out of bag error estimated during training phase of the regression model. The plot indicates that there is no significant reduction in the error rate after 100 decision trees.

![Fig. 6.2 Error Plot for Training Regression Model](image)

### 6.4.4 Performance Analysis

After the error estimation then the variable importance is calculated using the two important measures namely %IncMSE and IncNodePurity.
1. %IncMSE – Using the permutation this measure is estimated from the test data. For each tree present in the random forest, the prediction error for the test samples is estimated. This procedure is repeated after permutation of the predictor variable. Over all trees the difference between the error for the normal variable and the error for the permutation of the predictor variable is averaged. Then it is normalized by the standard deviation of the differences. If the difference is higher, then the importance of the variable is also high. The MSE is given in Equation (6.3),

\[ MSE = \text{mean}((y_{\text{actual}} - y_{\text{predicted}})^2) \]  

(6.3)

2. IncNodePurity - By averaging the overall decrease in the node impurities from splitting on the variable the node purity is calculated. It is measure by residual sum of squares. Impurity is calculated only at node at which that variable is used for that split. The importance of the top 10 variables is presented in the Fig. 6.3 using a variable importance plot. Depending on the variable importance plot, the variables can be selected for other predictive modeling or machine learning technique. This will give the user a clear idea about on list of features which are of high importance for the performance of the software.

Then the performance of the constructed prediction model is analyzed using the RMSE and MAE parameters. RMSE is the difference between values response predicted by a model or an estimator and the values actually observed. The RMSE represents the sample standard deviation of the differences between predicted values and observed values. These individual differences are called residuals when the calculations are performed over the data sample that was used for estimation, and are called prediction errors when computed out-of-sample.
Fig. 6.3 Variable Importance Plot (Top 10 Variables)

The RMSE serves to aggregate the magnitudes of the errors in predictions for various times into a single measure of predictive power. RMSE is a good measure of accuracy, but only to compare forecasting errors of different models for a particular variable and not between variables, as it is scale-dependent. In an optimal model the RMSE for the training and the test sets should be very similar. If the RMSE for the test set is much higher than that of the training set, it is likely that the training data may be over fit. The RMSE is given in Equation (6.4),

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} |f_i - y_i|^2}$$  \hspace{1cm} (6.4)

The MAE is a quantity used to measure how close forecasts or predictions are to the eventual outcomes. The MAE is given in Equation (6.5),
where $f_i$ is the predicted response and $y_i$ is the actual response. The Mean Absolute Error is an average of the absolute errors where $i$ is the prediction and the true value.

\[
MAE = \frac{1}{n} \sum_{i=1}^{n} |f_i - y_i| = \frac{1}{n} \sum_{i=1}^{n} |e_i| \tag{6.5}
\]

In attempting to find out whether the error measures in the estimation time frame are dependable, it ought to be checked whether the model under thought is probably going to have overfitted the data and whether its suspicions instinctively sensible. It ought to likewise be checked whether the figure plots resemble a sensible extrapolation of the past information. On the off chance that the presumptions appear to be sensible, then it is more probable that the error measurements can be trusted than if the suspicions were flawed.

On the off chance that the model has just a single or two parameters, (for example, an arbitrary walk, exponential smoothing, or basic relapse display) and was fitted to a direct or expansive sample of time series data (say, 30 perceptions or all the more), then it is most likely improbable to have overfitted the data. In any case, in the event that it has numerous parameters with respect to the quantity of perceptions in the estimation time frame, then overfitting is a particular plausibility. Regression models which are picked by applying programmed show determination strategies (e.g., stepwise or every single conceivable relapse) to vast quantities of uncritically picked applicant factors are inclined to overfit the data, regardless of the possibility that the quantity of regressors in the last model is little.
As an unpleasant guide against overfitting, figure the quantity of information focuses in the estimation time frame per coefficient assessed (counting regular files on the off chance that they have been independently evaluated from similar information). In the event that there are under 10 data focuses per coefficient assessed, there might be high likelihood of overfitting. Entirely, the assurance of a satisfactory example estimate should rely on upon the flag to-commotion proportion in the information, the nature of the choice or deduction issue to be explained, and from the earlier learning of whether the model particular is right. There are likewise efficiencies to be picked up while evaluating different coefficients all the while from similar information. Be that as it may, thinking as far as information focuses per coefficient is still a valuable rude awakening, especially when the example size is little and the flag is feeble.

At the point when fitting regression models to regular time arrangement information and utilizing sham factors to gauge month to month or quarterly impacts, there is somewhat decision about the quantity of parameters the model should incorporate. The occasional example in some mold, regardless of how little the specimen ought to be assessed, and ought to dependably incorporate the full set, i.e., don't specifically evacuate regular shams whose coefficients are not altogether unique in relation to zero. When in doubt, it regards have no less than 4 seasons of information. More would be better however long time histories may not be accessible or adequately applicable to what is occurring now, and utilizing a gathering of occasional sham factors as a unit does not convey an indistinguishable sort of danger of overfitting from utilizing a comparable number of regressors that are arbitrary factors looked over a huge pool of hopefuls. On the off chance that it is consistent for the arrangement to have an occasional example, then there is no doubt of the pertinence of the factors that measure it.
Despite the fact that the certainty interims for one-stage ahead conjectures are construct completely in light of RMSE, the certainty interims for the more extended skyline figures that can be created by time-arrangement models depend intensely on the fundamental demonstrating suspicions, especially suppositions about the fluctuation of the pattern. The certainty interims for a few models extend generally gradually as the estimate skyline is stretched (e.g., basic exponential smoothing models with little estimations of "alpha", straightforward moving midpoints, regular irregular walk models, and direct pattern models). The certainty interims extend much quicker for different sorts of models (e.g., non-regular arbitrary walk models, occasional irregular pattern models, or direct exponential smoothing models). The rate at which the certainty interims enlarge is not a dependable manual for model quality: what is vital is the model ought to make the right suspicions about how dubious what's to come is. It is imperative that the model ought to breeze through the different leftover indicative tests and "eyeball" tests all together for the certainty interims for longer-skyline estimates to be considered important.

So, the primary concern is that more weightage must be given on the error measures in the estimation time frame - frequently the RMSE (or standard blunder of the relapse, which is RMSE balanced for the relative intricacy of the model), however in some cases MAE when looking at among models. The MAE measurement gives an exceptionally valuable rude awakening for a model fitted to time arrangement information: is it any superior to anything an innocent model.

To analyze the performance of the random forest for regression analysis, the training data is given as input to models with different node size and is presented in Fig. 6.4.
Lower values of RMSE in the graph indicate the better fit of the model for the data. The accuracy in predicting the response can be measured with the RMSE value. It is the most important criterion for fit if the main purpose of the model is prediction. In regression analysis problem, the mean squared error is rarely utilized to denote the unbiased estimate of error variance which is equal to the residual sum of squares divided by the number of degrees of freedom. Prediction accuracy can be defined as the correlation between the predicted value and the actual value. Accuracy of 1 indicates a perfect accuracy, whereas the accuracy of 0 indicates a random guess.

**6.5 SUMMARY**

The random forest based regression analysis is preformed to predict the performance of the software system. The dataset, used for training and testing the developed
prediction model, is composed of categorical data i.e. they represent the presence of a feature in the software application or not. The random forest based regression analysis is best suited for categorical kind of data. The proposed method for predicting the performance of the configurable software system exhibits high accuracy and low error rate. Also, this method is capable of presenting the variable importance as a plot. The error rate during the training phase of the regression model is also very less.