Edureka (www.edureka.co)
Data Science Certification Training

Report of Certification Project

By: Sushil Kumar Verma
sushilverma208016@gmail.com

About Data:

It is a company's human resource data where data of employee's salary, bonus, promotion etc. and his stay in company is given. The task is to create a predicting model for VP of company to predict stay rate of current employees.

Size: 14999 rows and 10 columns

Columns:

- Employee satisfaction level (satisfaction_level : numeric)
- Last evaluation (last_evaluation : numeric)
- Number of projects (numer_project : numeric)
- Average monthly hours (average_monthly_hours : numeric)
- Time spent at the company (time_spend_company : numric)
- Whether they have had a work accident (work_accident : numeric)
- Whether the employee has left (left : categorical)
- Whether they have had a promotion in the last 5 years (promotion_last_5years : numeric)
- Department (department : categorical)
- Salary (salary : categorical)

Target column is left which is 7^{th} column of dataset.

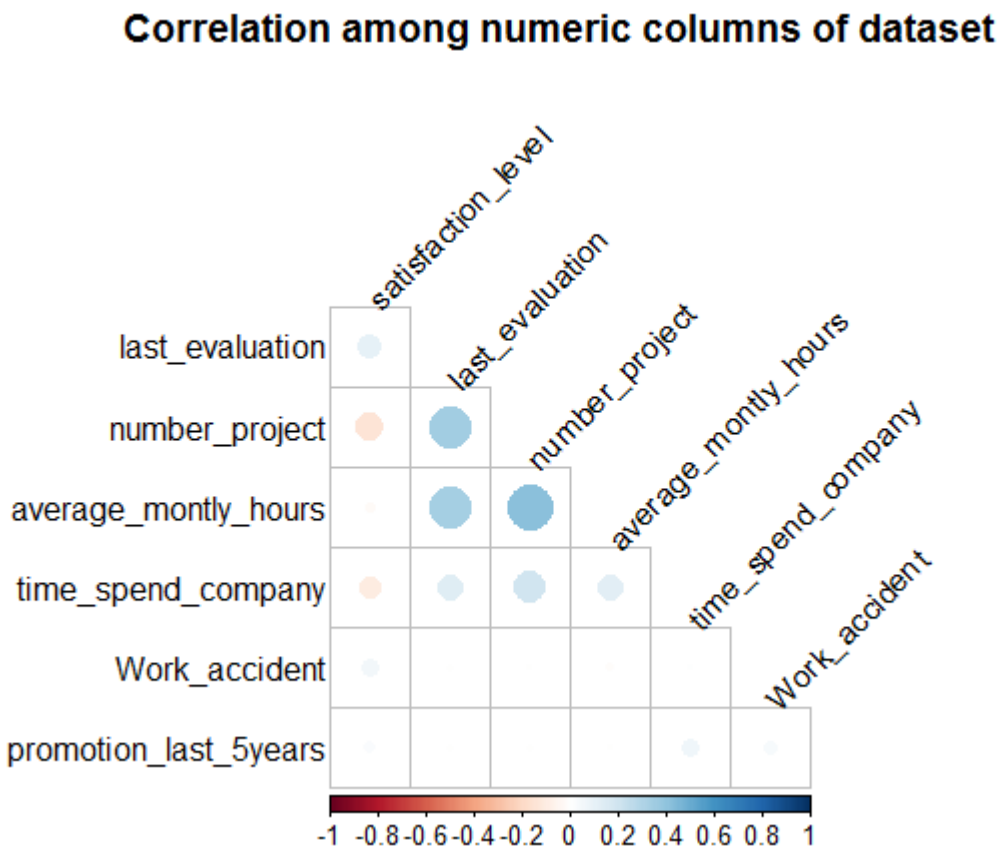
Data Wrangling:

- Factorized categorical columns: left, department and salary.
- Scaled remaining numeric columns.
- As all columns are potential, so dimensionality reduction must not do.
- In each classifier, ratio of training and test data is taken as 80:20

Correlation among variables:

Used ‘corrr’, ‘Hmisc’ and ‘corrplot’ to find and plot correlation among numeric variables. The result is below:

	last_evaluation	number_project	average_montly	time_spend_company	Work_accident	promotion_last_5years
satisfaction_level	1.00000000	0.105021214	-0.142969586	-0.020048113	-0.100866073	0.058697241	0.025605186
last_evaluation	0.10502121	1.000000000	0.349332589	0.339741800	0.131590722	-0.007104289	-0.008683768
number_project	-0.14296959	0.349332589	1.000000000	0.417210634	0.196785891	-0.004740548	-0.006063958
average_montly	-0.02004811	0.339741800	0.417210634	1.000000000	0.127754910	-0.010142888	-0.003544414
time_spend_company	-0.10086607	0.131590722	0.196785891	0.127754910	1.000000000	0.002120418	0.067432925
Work_accident	0.05869724	-0.007104289	-0.004740548	-0.010142888	0.002120418	1.000000000	0.039245435
promotion_last_5years	0.02560519	-0.008683768	-0.006063958	-0.003544414	0.067432925	0.039245435	1.000000000



Decision Tree Classifier:

- Used package ‘rpart’ for decision tree
- Average accuracy of 5-fold cross validation: 90.89%
- The plot between accuracy of decision tree and its accuracy parameter is below:

Confusion Matrix

	0	1
0	11217	1156
1	211	2415

Random Forest Classifier:

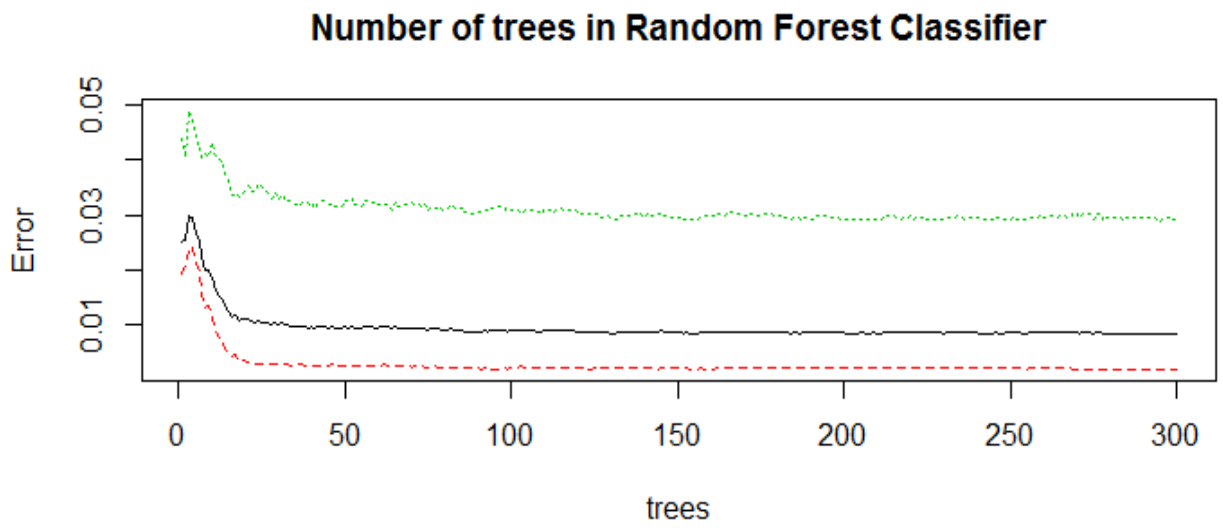
- Used library 'randomForest' for random forest classifier
- With number of trees (ntree)=300, achieved 98.23% accuracy

  Confusion Matrix

	0	1
0	2283	3
1	50	664
- With 10-fold cross validation achieved on average 99.16% accuracy.

  Accuracies of these ten iteration are:

  99.20000 99.13333 99.20000 99.13333 99.13333 99.13333 99.20000 99.13333 99.20000 99.13333
- The plot between error in random forest classifier and number of tree is:



Support Vector Machine (SVM):

- Used library ‘e1071’ to build SVM model
- Achieved 77.37% accuracy with linear kernel SVM

  Confusion Matrix

	0	1
0	2147	139
1	540	174

- Polynomial kernel SVM of degree four gave 93.83% accuracy

  Confusion Matrix

	0	1
0	2222	64
1	121	593

- Sigmoid kernel SVM gave 57.87% accuracy

  Confusion Matrix

	0	1
0	1677	609
1	655	59

k-Nearest Neighbor (k-NN) Classifier:

- Library ‘class’ is used for k-NN classifier

- Achieved accuracy of 94.97% from k-NN algorithm using k=5
  Confusion Matrix

	0	1
0	2187	99
1	52	662

- Varied value of k from 1 to 10 and below are accuracies for each k:
  # Value of K: 1 Accuracy: 0.97
  # Value of K: 2 Accuracy: 0.952
  # Value of K: 3 Accuracy: 0.954666666666667
  # Value of K: 4 Accuracy: 0.952666666666667
  # Value of K: 5 Accuracy: 0.949666666666667
  # Value of K: 6 Accuracy: 0.949666666666667
  # Value of K: 7 Accuracy: 0.952
  # Value of K: 8 Accuracy: 0.947666666666667
  # Value of K: 9 Accuracy: 0.948333333333333
  # Value of K: 10 Accuracy: 0.947666666666667

Logistic Regression Classifier:

- Used function 'glm' to create a logistic regression classifier.
- Achieved accuracy of 79.1% with 'binomial' family and probability threshold of 0.5
  Confusion Matrix

	0	1
0	2117	169
1	458	256

- Tried different family of function with logistic regression and their accuracies are:
  # Family : binomial Accuracy: 0.793333333333333
  # Family : gaussian Accuracy: 0.793333333333333
  # Family : Gamma Accuracy: 0.793333333333333
  # Family : inverse.gaussian Accuracy: 0.793333333333333
  # Family : poisson Accuracy: 0.793333333333333
  # Family : quasi Accuracy: 0.793333333333333
  # Family : quasibinomial Accuracy: 0.793333333333333
  # Family : quasipoisson Accuracy: 0.793333333333333

Naïve Bayes Classifier:

- Used library ‘e1071’ to implement naïve Bayes classifier
- Simple naïve Bayes classifier gave 78.67% accuracy
  Confusion Matrix

	0	1
0	1854	432
1	208	506

- Naïve Bayes classifier with ‘laplace’ value=2000, ‘eps’=0.1 and ‘threshold’=0.5 gave 79.67% accuracy
  Confusion Matrix

	0	1
0	1899	387
1	223	491

Neural Network:

- Library ‘h2o’ is used for neural network
- Obtained asccuracy of 94.53% with ‘Rectifier’ activation function, 5 hidden layers and 100 epochs
  Confusion Matrix

	0	1
0	2215	71
1	93	621

- Obtained accuracy of 96.43% with ‘Rectifier’ activation function, 100 hidden layers and 100 epochs
  Confusion Matrix

	0	1
0	2231	55
1	52	662

- Obtained accuracy of 97% with ‘tanh’ activation function, 50 hidden layers and 1000 epochs
  Confusion Matrix

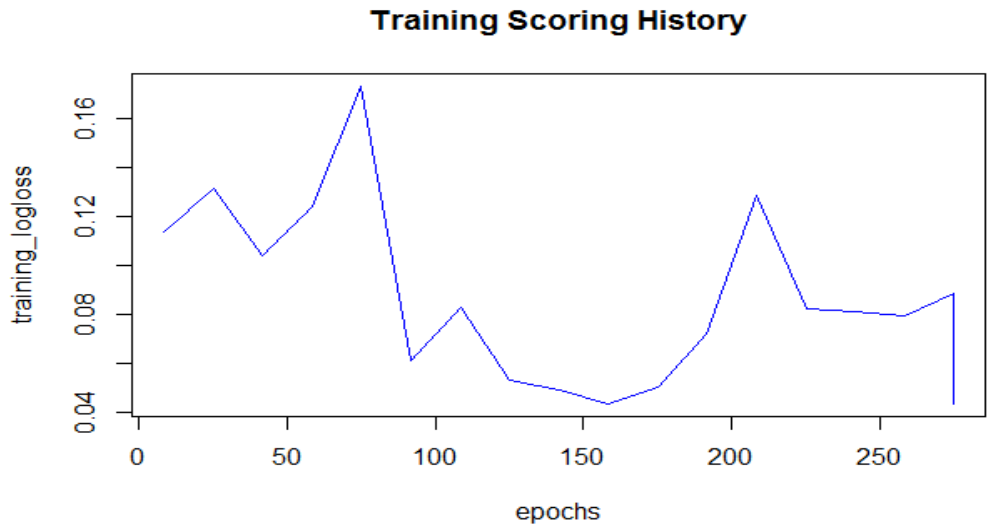
	0	1
0	2245	41
1	49	665

- Obtained accuracy of 97.33% with ‘tanh’ activation function, 50 hidden layers and 150 epochs

  Confusion Matrix

	0	1
0	2247	39
1	41	673

- Training Scoring History of neural network classifier is given in the below plot:



Summary of accuracies of different classifiers:

- Decision Tree: 90.89%
- Random Forest: 99.20%
- SVM: 93.83%
- K-NN: 97%
- Logistic Regression: 79.33%
- Naive Bayes Classifier: 79.67%
- Neural Network: 97.33%