**A Summer Internship Project Report On**

****

**A Study of Deep Learning technique using Data mining tool (orange)**

**For Online News Popularity using regression and**

**Bank marketing analysis using classification**

**Under the guidance of:**

**Gaurav Bhatt Dr. Ankush Mittal**

**Prepared by:**

**VARUN SINGHAL MANSI BANSAL**

Computer science Computer science

& Engineering & Engineering

**INDEX**

**INTRODUCTION----------------------------------------------------------**

**DATA SET DESCRIPTION----------------------------------------------**

**PRELIMINARIES----------------------------------------------------------**

* **CLASSIFIERS:**
* **Support Vector Machine (SVM)**
* **K-Nearest Neighbours (KNN)**
* **Random Forest**
* **Neural Network**
* **REGRESSION**
* **Linear Regression**

**EXPERIMENTS------------------------------------------------------------**

**RESULT-----------------------------------------------------------------------**

**CONCLUSION--------------------------------------------------------------**

**INTRODUCTION**

**1.1 Problem statement**

1. **CLASSIFICATION PROBLEM** : We have a bank marketing classification problem in which we are predicting if the client will subscribe a term deposit . All bank marketing campaigns are dependent on customers huge electronic data. The size of these data sources is impossible for a human analyst to come up with interesting information that will help in the decision-making process. Data mining models are completely helping in the performance of these campaigns and also helps in predicting the best campaign contact with the clients for subscribing deposit.
2. **REGRESSION PROBLEM :** Here, we are analyzing the number of shares depending on the attributes and predict if the news will get popular on internet or not from dataset of online news popularity. The dataset is divided into two sets that is training and testing dataset. This type of analysis is used for the purpose for finding out what elements makes an article or news popular. Since, social media is a popular way to share news and interesting articles, it is not uncommon for a person to be active on both facebook and twitter to share an article on both platforms. Thus, one measure for popularity of an article could be how many times it has been shared. Thus, an article can be neutral (no shares), unpopular (low shares), popular (high shares) or viral based on how many times it has been shared.

**1.2 Deep Learning**

Deep learning is a machine learning technique that teaches computers to do what comes naturally to humans: learn by example. Deep learning is a key technology behind driverless cars, enabling them to recognize a stop sign, or to distinguish a pedestrian from a lamppost. It is the key to voice control in consumer devices like phones, tablets, TVs, and hands-free speakers. Deep learning is getting lots of attention lately and for good reason. It’s achieving results that were not possible before.

In deep learning, a computer model learns to perform classification tasks directly from images, text, or sound. **Deep learning** models can achieve state-of-the-art accuracy, sometimes exceeding human-level performance.

**1.3 Machine Learning**

**Machine learning** is a method of data analysis that automates analytical model building. Using algorithms that iteratively learn from data, **machine learning** allows computers to find hidden insights without being explicitly programmed where to look.

Some popular machine learning methods

Two of the most widely adopted machine learning methods are supervised **learning** and **unsupervised learning** – but there are also other methods of machine learning as follows-

* **Supervised learning**algorithms are trained using labelled examples, such as an input where the desired output is known. Supervised learning is commonly used in applications where historical data predicts likely future events. For example, it can anticipate when credit card transactions are likely to be fraudulent or which insurance customer is likely to file a claim.

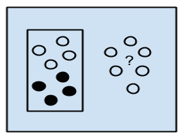


Fig1.1: Supervised Learning Algorithm

* **Unsupervised learning**is used against data that has no historical labels. The system is not told the "right answer." The algorithm must figure out what is being shown. The goal is to explore the data and find some structure within. Unsupervised learning works well on transactional data.

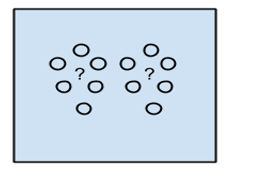


Fig1.2: Unsupervised Learning Algorithm

* **Reinforcement learning** is often used for robotics, gaming and navigation. With reinforcement learning, the algorithm discovers through trial and error which actions yield the greatest rewards.

**1.4 Applications of Machine Learning**

It is very interesting to know the applications of machine learning. Google and Facebook uses ML extensively to push their respective ads to the relevant users. Here are a few applications that you should know:

* **Banking & Financial services**: ML can be used to predict the customers who are likely to default from paying loans or credit card bills. This is of paramount importance as machine learning would help the banks to identify the customers who can be granted loans and credit cards.
* **Healthcare**: It is used to diagnose deadly diseases (e.g. cancer) based on the symptoms of patients and tallying them with the past data of similar kind of patients.

**1.5 Confusion matrix in machine learning**

In the field of machine learning and specifically the problem of statistical classification, a confusion matrix, also known as an error matrix. A confusion matrix is a table that is often used to describe the performance of a classification model (or “classifier”) on a set of test data for which the true values are known. It allows the visualization of the performance of an algorithm. It allows easy identification of confusion between classes e.g. one class is commonly mislabeled as the other. Most performance measures are computed from the confusion matrix.

A confusion matrix is a summary of prediction results on a classification problem

.  
The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix.  
The confusion matrix shows the ways in which your classification model is confused when it makes predictions. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made.



Positive (P): Observation is positive (for example: is an apple).  
 Negative (N): Observation is not positive (for example: is not an apple).  
 True Positive (TP): Observation is positive, and is predicted to be positive.  
 False Negative (FN): Observation is positive, but is predicted negative.  
 True Negative (TN): Observation is negative, and is predicted to be negative.  
 False Positive (FP): Observation is negative, but is predicted positive.

Classification Rate or Accuracy is given by the relation:  


**1.6 SOFTWARE USED ORANGE**

Orange is an open-source software package released under GPL. Versions up to 3.0 include core components in C++ with wrappers in Python are available on [github](https://github.com/biolab/orange). From version 3.0 onwards, Orange uses common Python open-source libraries for scientific computing, such as numpy, scipy and scikit-learn, while its graphical user interface operates within the cross-platform Qt framework. Orange3 has a separate [github](https://github.com/biolab/orange3).

The default installation includes a number of machine learning, preprocessing and data visualization algorithms in 6 widget sets (data, visualize, classify, regression, evaluate and unsupervised). Additional functionalities are available as add-ons (bioinformatics, data fusion and text-mining).

Orange is supported on macOS, Windows and Linux and can also be installed from the Python Package Index repository (*pip install Orange3*).

**FEATURES**

Orange consists of a canvas interface onto which the user places widgets and creates a data analysis workflow. Widgets offer basic functionalities such as reading the data, showing a data table, selecting features, training predictors, comparing learning algorithms, visualizing data elements, etc. The user can interactively explore visualizations or feed the selected subset into other widgets.

* **Canvas**: graphical front-end for data analysis
* **Widgets**:
  + **Data**: widgets for data input, data filtering, sampling, imputation, feature manipulation and feature selection
  + **Visualize**: widgets for common visualization (box plot, histograms, scatter plot) and multivariate visualization (mosaic display, sieve diagram).
  + **Classify**: a set of supervised machine learning algorithms for classification
  + **Regression**: a set of supervised machine learning algorithms for regression
  + **Evaluate**: cross-validation, sampling-based procedures, reliability estimation and scoring of prediction methods
  + **Unsupervised**: unsupervised learning algorithms for clustering (k-means, hierarchical clustering) and data projection techniques (multidimensional scaling, principal component analysis, correspondence analysis).
  + **Add-ons**:
    - **Associate**: widgets for mining frequent item sets and association rule learning
    - **Bioinformatics**: widgets for gene set analysis, enrichment, and access to pathway libraries
    - **Data fusion**: widgets for fusing different data sets, collective matrix factorization, and exploration of latent factors
    - **Educational**: widgets for teaching machine learning concepts, such as k-means clustering, polynomial regression, stochastic gradient descent, ...
    - **Geo**: widgets for working with geospatial data
    - **Image analytics**: widgets for working with images and Image Net embeddings
    - **Network**: widgets for graph and network analysis
    - **Text mining**: widgets for natural language processing and text mining
    - **Time series**: widgets for time series analysis and modeling

**OBJECTIVES:**

in science, it is used as a platform for testing new machine learning algorithms and for implementing new techniques in genetics and bioinformatics. In education, it was used for teaching machine learning and data mining methods to students of biology, biomedicine and informatics.

**DATASET DESCRIPTION**

1. **BANK MARKETING DATASET***:*

The characteristics of data set composed of two kinds: nominal and numeral

Attributes as shown below:

Input variables:  
# bank client data:  
1 - age (numeric)  
2 - job : type of job (categorical: 'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')  
3 - marital : marital status (categorical: 'divorced','married','single','unknown'; note: 'divorced' means divorced or widowed)  
4 - education (categorical: 'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown')  
5 - default: has credit in default? (categorical: 'no','yes','unknown')  
6 - housing: has housing loan? (categorical: 'no','yes','unknown')  
7 - loan: has personal loan? (categorical: 'no','yes','unknown')  
# related with the last contact of the current campaign:  
8 - contact: contact communication type (categorical: 'cellular','telephone')   
9 - month: last contact month of year (categorical: 'jan', 'feb', 'mar', ..., 'nov', 'dec')  
10 - day\_of\_week: last contact day of the week (categorical: 'mon','tue','wed','thu','fri')  
11 - duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.  
# other attributes:  
12 - campaign: number of contacts performed during this campaign and for this client (numeric, includes last contact)  
13 - pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means client was not previously contacted)  
14 - previous: number of contacts performed before this campaign and for this client (numeric)  
15 - poutcome: outcome of the previous marketing campaign (categorical: 'failure','nonexistent','success')  
# social and economic context attributes  
16 - emp.var.rate: employment variation rate - quarterly indicator (numeric)  
17 - cons.price.idx: consumer price index - monthly indicator (numeric)   
18 - cons.conf.idx: consumer confidence index - monthly indicator (numeric)   
19 - euribor3m: euribor 3 month rate - daily indicator (numeric)  
20 - nr.employed: number of employees - quarterly indicator (numeric)  
  
Output variable (desired target):  
21 - y - has the client subscribed a term deposit? (binary: 'yes', 'no')

Here the attribute named Job, there exist many kinds of jobs belonging to this attribute as (admin, unknown, unemployed, management, housemaid, entrepreneur, student, blue-collar, self-employed, retired, technician, and services). The attribute Marital can be illustrated in classes as (married, divorced, and single) where the class divorce means divorced or widowed. The Education classes are divided into unknown, secondary, primary, and tertiary; however, in attributes Default, Housing, Loan, and the output attribute has only two classes (yes, and no). The contact communication classes in the Contact attribute are: unknown, telephone, and cellular. Clearly, in the attribute Month the classes are month’s names Jan, Feb, etc. The attribute Poutcome presents the outcome of the previous marketing campaign like: unknown, other, failure, and success. The last column in table 2 introduces the duration for each range in the numerical kind of attributes; for example, Age attribute has (18:95) in duration; that means all ages for customers or samples range between 18 and 95 years.

1. **ONLINE NEWS POPULARITY DATASET:**

We study popularity of a news using the online news popularity dataset from UCI .The original data has 61 variable in all including the response variable “shares” and 39644 observations in which 58 are predictive attributes ,2 are non-predictive and 1 goal field .Out of these the first two variables containing the URL of the article and time delta to data acquisition are not useful for analysis. Of the rest, many are binary variables and by visual inspection some seem to be correlated.

Here is the detail attribute information:

*Non predictive features* :

0. url: URL of the article (nonpredictive)

1. timedelta: Days between the article publication and the dataset acquisition (nonpredictive)

*Predictive features* :

2. n\_tokens\_title: Number of words in the title

3. n\_tokens\_content: Number of words in the content

4. n\_unique\_tokens: Rate of unique words in the content

5. n\_non\_stop\_words: Rate of nonstop

words in the content

…

. ..

56. title\_subjectivity: Title subjectivity

57. title\_sentiment\_polarity: Title polarity

58. abs\_title\_subjectivity: Absolute subjectivity level

59. abs\_title\_sentiment\_polarity: Absolute polarity level

*Goal predicted attribute* :

60. shares: Number of shares (target)

Of the binary variables, Saturday and Sunday information is aggregated into the variable is\_weekend so, is\_weekday\_<Saturday-Sunday> can be safely removed. Of the variables is\_weekday\_<Monday- Friday> are atomic variables. Aggregating information from these five variables into a new variable “is\_weekday” will improve quality of the explanatory variable.

Variable 19-27 and 50-55 which have information about the number of keywords and positive words. Here, the variables are average, minimum and maximum. The average encompasses information from the minimum and maximum so, there is some collinearity between these variables. Thus, to reduce effects of singularity I removed the variables with suffix “\_min\_” and “\_max\_” while keeping “\_avg\_.” Thus, reducing variables and avoiding possible ill-effects after visually inspection.

**CLASSIFIERS**

**K-Nearest Neighbors(KNN)**

“In [pattern recognition](https://en.wikipedia.org/wiki/Pattern_recognition), the k-nearest neighbors algorithm (k-NN) is a [non-parametric](https://en.wikipedia.org/wiki/Non-parametric_statistics) method used for [classification](https://en.wikipedia.org/wiki/Statistical_classification) and [regression](https://en.wikipedia.org/wiki/Regression_analysis). In both cases, the input consists of the k closest training examples in the [feature space](https://en.wikipedia.org/wiki/Feature_space). The output depends on whether k-NN is used for classification or regression:

* In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive [integer](https://en.wikipedia.org/wiki/Integer), typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor.
* In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors.

k-NN is a type of [instance-based learning](https://en.wikipedia.org/wiki/Instance-based_learning), or [lazy learning](https://en.wikipedia.org/wiki/Lazy_learning), where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all [machine learning](https://en.wikipedia.org/wiki/Machine_learning) algorithms.”[14]

**Support Vector Machine (SVM)**

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well.

[](https://www.analyticsvidhya.com/wp-content/uploads/2015/10/SVM_1.png)

Fig 4.4(a) :SVM Image

Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

* **Identify the right hyper-plane:**Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.



Fig 4.4(b): Identify the right hyper plane

* **Identify the right hyper-plane:**Here, we have three hyper-planes (A, B and C) and all are segregating the classes well. Now, to identify right hyper plane.



Fig 4.4(c): Identify the right hyper plane

* Maximizing the distances between nearest data point and hyper-plane will help us to decide the right hyper-plane. This distance is called as **Margin**

[[](https://www.analyticsvidhya.com/wp-content/uploads/2015/10/SVM_4.png)](https://www.analyticsvidhya.com/wp-content/uploads/2015/10/SVM_4.png)Fig 4.4(d) : Decide Right Hyper Plane

Above, you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C.

* **Identify the right hyper-plane:** Use the rules as discussed in previous section to identify the right hyper-plane

**[](https://www.analyticsvidhya.com/wp-content/uploads/2015/10/SVM_5.png)**

Fig 4.4(e): Identify the Right Hyper Plane

**Identify two classes:**Here, its unable to segregate the two classes using a straight line, as one of star lies in the territory of other(circle) class as an outlier.

**[](https://www.analyticsvidhya.com/wp-content/uploads/2015/10/SVM_61.png)**

Fig 4.4(f): Identify two classes

* **Identify hyper-plane to segregate to classes:**In the scenario below, we can’t have linear hyper-plane between the two classes, so how does SVM classify these two classes? here we have only looked at the linear hyper-plane .[13]

**[](https://www.analyticsvidhya.com/wp-content/uploads/2015/10/SVM_8.png)**

Fig : Identify hyper plane to segregate classes

**RANDOM FOREST:**

“Random Forest is a [statistical](https://simple.wikipedia.org/wiki/Statistics) [algorithm](https://simple.wikipedia.org/wiki/Algorithm) that is used to cluster [points of data](https://simple.wikipedia.org/wiki/Information) in functional groups. When the data set is large and/or there are many [variables](https://simple.wikipedia.org/wiki/Variable) it becomes difficult to cluster the data because not all variables can be considered, therefore the algorithm can also give a certain chance that a data point belongs in a certain group. It is an [ensemble learning](https://en.wikipedia.org/wiki/Ensemble_learning) method for [classification](https://en.wikipedia.org/wiki/Statistical_classification), [regression](https://en.wikipedia.org/wiki/Regression_analysis) and other tasks, that operate by constructing a multitude of [decision trees](https://en.wikipedia.org/wiki/Decision_tree_learning) at training time and outputting the class that is the [mode](https://en.wikipedia.org/wiki/Mode_(statistics)) of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of [overfitting](https://en.wikipedia.org/wiki/Overfitting" \o "Overfitting) to their training set.”[15]

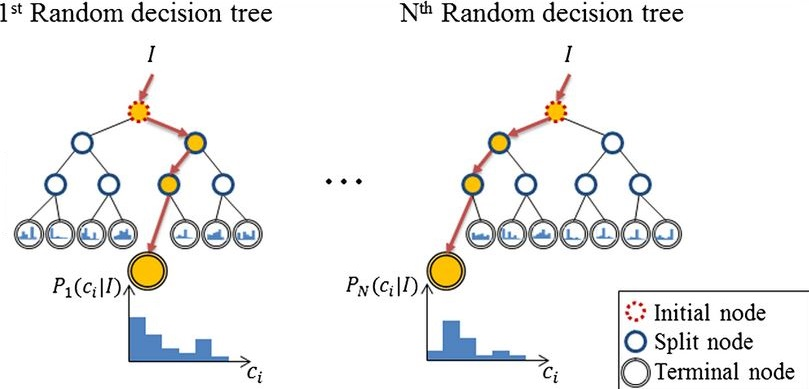


Fig 4.5:This is how the clustering takes place.

* Of the entire set of data, a subset is taken (training set).
* The algorithm clusters the data in groups and subgroups. If you would draw lines between the data points in a subgroup, and lines that connect subgroups into group etc. the structure would look somewhat like a tree. This is called a [decision tree](https://simple.wikipedia.org/w/index.php?title=Decision_tree&action=edit&redlink=1).
  + At each split or node in this cluster/tree/dendrogram variables are chosen at random by the program to judge whether data points have a close relationship or not.
* The program makes multiple trees for a forest. Each tree is different because for each split in a tree, variables are chosen at random.
* Then the rest of the dataset (not the training set) is used to predict which tree in the forests makes the best classification of the data points (in the dataset the right classification is known).
* The tree with the most predictive power is shown as output by the algorithm”

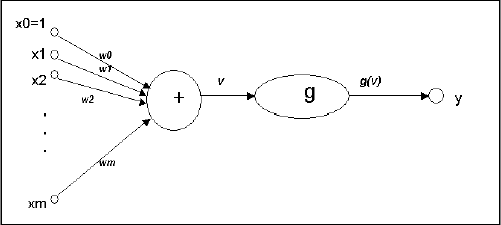
**NEURAL NETWORK**

Artificial neural networks are relatively crude electronic networks of neurons based on the neural structure of the brain. They process records one at a time, and learn by comparing their classification of the record (i.e., largely arbitrary) with the known actual classification of the record. The errors from the initial classification of the first record is fed back into the network, and used to modify the networks algorithm for further iterations.

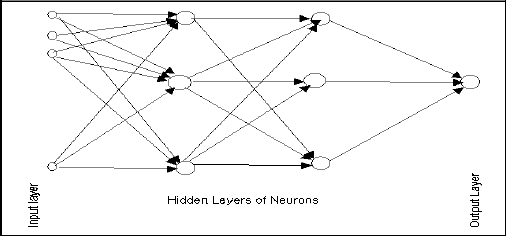
A neuron in an artificial neural network is

1. A set of input values (xi) and associated weights (wi).

2. A function (g) that sums the weights and maps the results to an output (y).



Neurons are organized into layers: input, hidden and output. The input layer is composed not of full neurons, but rather consists simply of the record's values that are inputs to the next layer of neurons. The next layer is the hidden layer. Several hidden layers can exist in one neural network. The final layer is the output layer, where there is one node for each class. A single sweep forward through the network results in the assignment of a value to each output node, and the record is assigned to the class node with the highest value.



In the training phase, the correct class for each record is known (termed supervised training), and the output nodes can be assigned correct values -- 1 for the node corresponding to the correct class, and 0 for the others. (In practice, better results have been found using values of 0.9 and 0.1, respectively.) It is thus possible to compare the network's calculated values for the output nodes to these correct values, and calculate an error term for each node (the Delta rule). These error terms are then used to adjust the weights in the hidden layers so that, hopefully, during the next iteration the output values will be closer to the correct values.

**LINEAR REGRESSION**

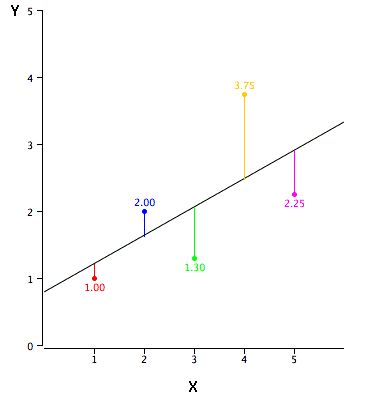
In linear regression, we predict scores on one variable from the scores on a second variable. The variable we are predicting is called the *criterion variable* and is referred to as Y. The variable we are basing our predictions on is called the *predictor variable* and is referred to as X. When there is only one predictor variable, the prediction method is called *simple regression*. In simple linear regression, the topic of this section, the predictions of Y when plotted as a function of X form a straight line.

The example data in Table 1 are plotted in Figure 1. You can see that there is a positive relationship between X and Y. If you were going to predict Y from X, the higher the value of X, the higher your prediction of Y.

|  |  |
| --- | --- |
| **X** | **Y** |
| 1.00 | 1.00 |
| 2.00 | 2.00 |
| 3.00 | 1.30 |
| 4.00 | 3.75 |
| 5.00 | 2.25 |

Linear regression consists of finding the best-fitting straight line through the points. The best-fitting line is called a *regression line*. The black diagonal line in Figure 2 is the regression line and consists of the predicted score on Y for each possible value of X. The vertical lines from the points to the regression line represent the errors of prediction. As you can see, the red point is very near the regression line; its error of prediction is small. By contrast, the yellow point is much higher than the regression line and therefore its error of prediction is large.

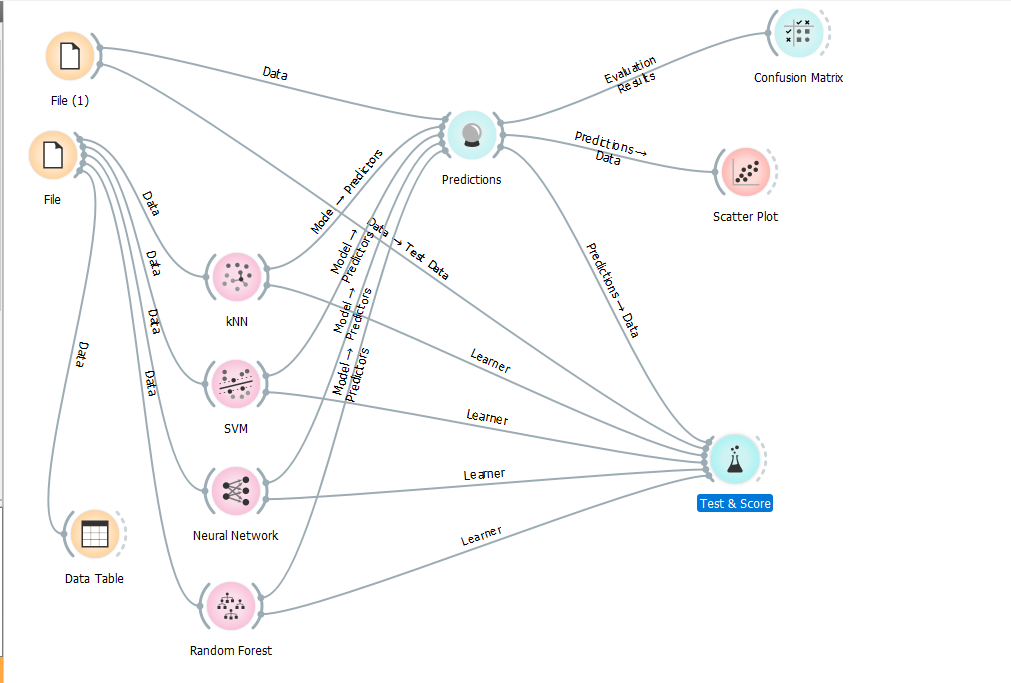
A scatter plot of the example data is shown below. The black line consists of the predictions, the points are the actual data, and the vertical lines between the points and the black line represent errors of prediction.



**EXPERIMENTS**

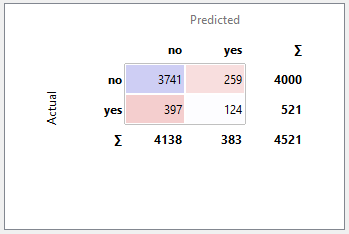
**1. CLASSIFICATION :**

We have performed machine learning by training our machine through different type of classifiers Support Vector Machine, K-Nearest and Random Forest, neural network and the pictorial representation of our model is shown below:



**Fig (1)**

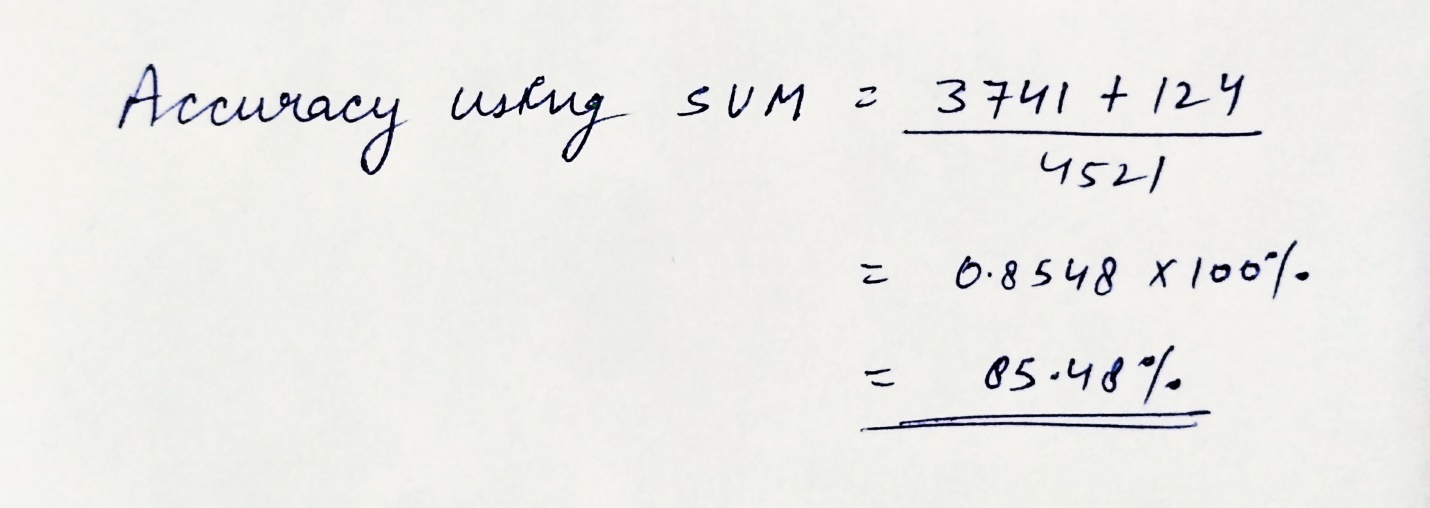
* **ANALYZING WITH SVM CLASSIFIER :**

****

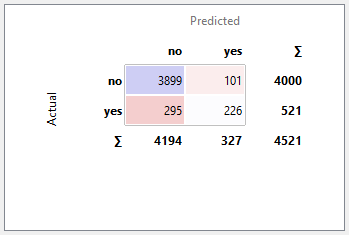
**The above image shows the confusion matrix after applying SVM classifier.**

SVM classifier is defined as a separating hyperplane method that is widely used for data analyzing pattern recognition. Supervised learning is performed on the training set, which outputs an optimal hyperplane for categorization of new examples, that is, the testing data to indicate which class it belongs to.

**ACCURACY FOUND IS:**

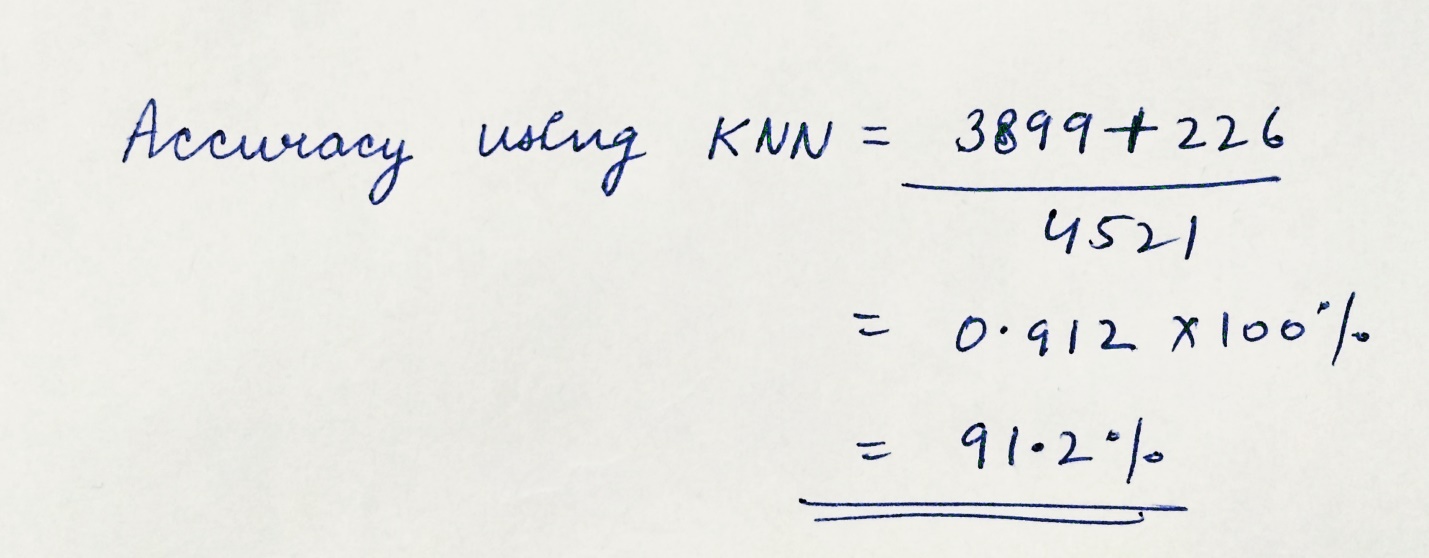
****

* **ANALYZING WITH KNN:**

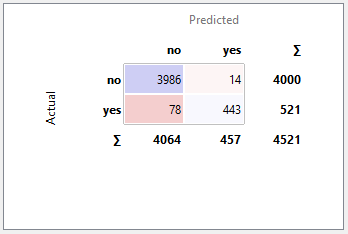
****

**THE ABOVE FIGURE SHOWS CONFUSION MATRIX WITH KNN CLASSIFIER**

The model for KNN is the entire training dataset. When a prediction is required for an unseen data instance, the KNN algorithm will search through the training dataset for the k-most similar instances. The prediction attribute of the most similar instances is summarized and returned as the prediction for the unseen instance. The similarity measure is dependent on the type of data. Other types of data such as categorical or binary data, hamming distance can be used. In the case of regression problems, the average of the predicted attribute may be returned.

**ACCURACY WITH KNN**

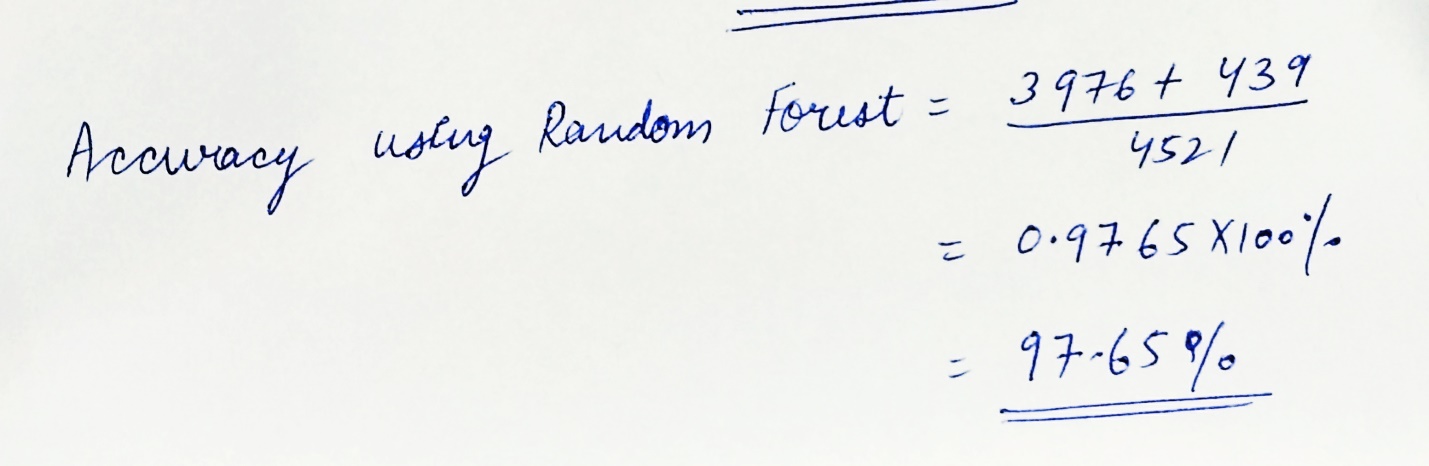
* **ANALYZING USING RANDOM FOREST :**

****

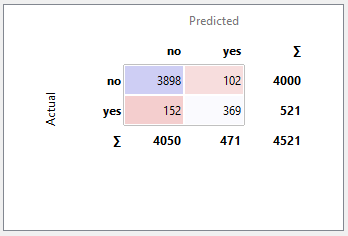
**This the confusion matrix by random forest classifier**

Random Forest is a [statistical](https://simple.wikipedia.org/wiki/Statistics) [algorithm](https://simple.wikipedia.org/wiki/Algorithm) that is used to cluster [points of data](https://simple.wikipedia.org/wiki/Information) in functional groups. When the data set is large and/or there are many [variables](https://simple.wikipedia.org/wiki/Variable) it becomes difficult to cluster the data because not all variables can be taken into account, therefore the algorithm can also give a certain chance that a data point belongs in a certain group. The function can also train random subspace ensembles of KNN or discriminant analysis classifiers

**ACCURACY WITH RANDOM FOREST** :

****

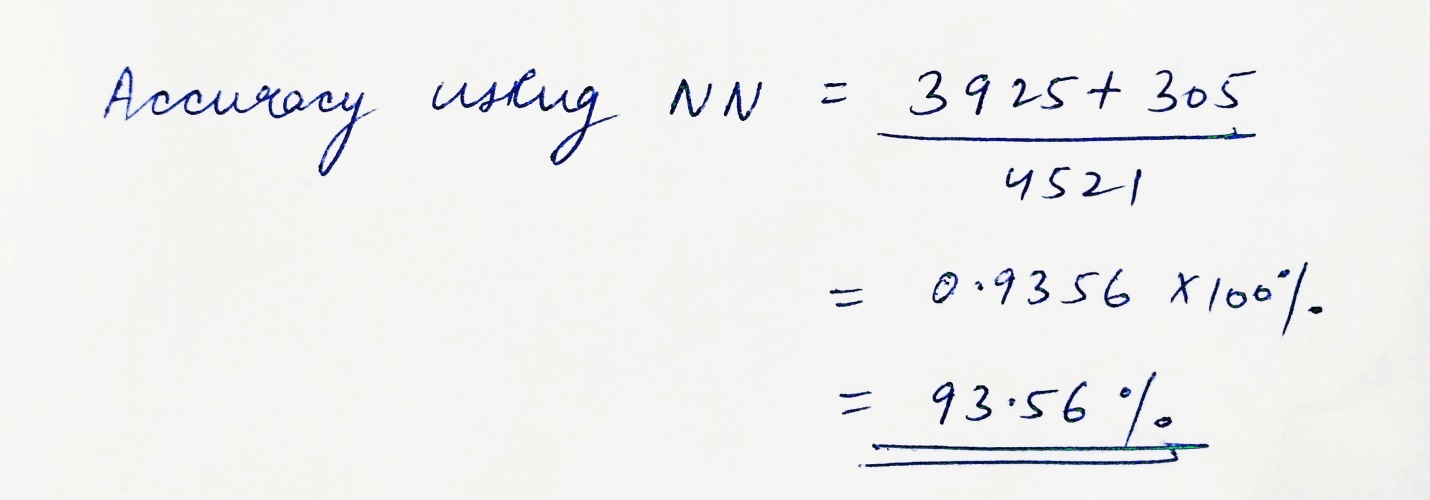
* **ANALYZING WITH NEURAL NETWORK:**

****

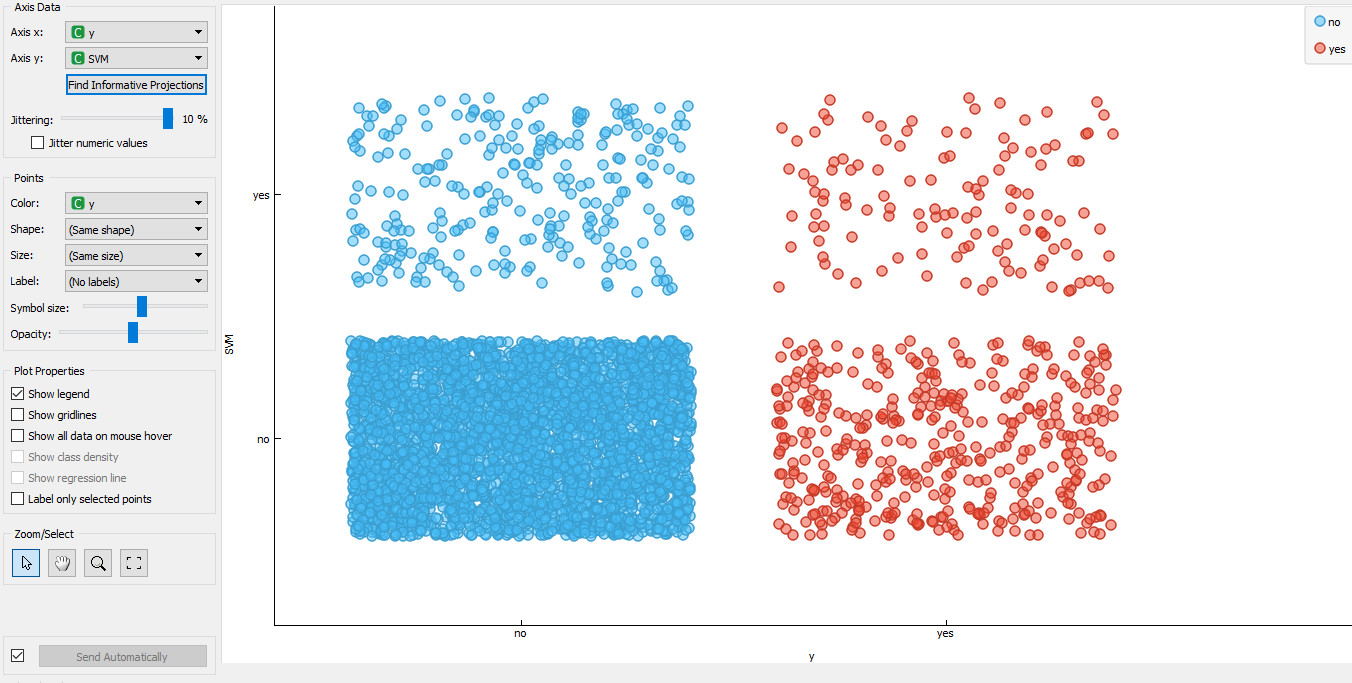
**This is the confusion matrix with neural network classifier.**

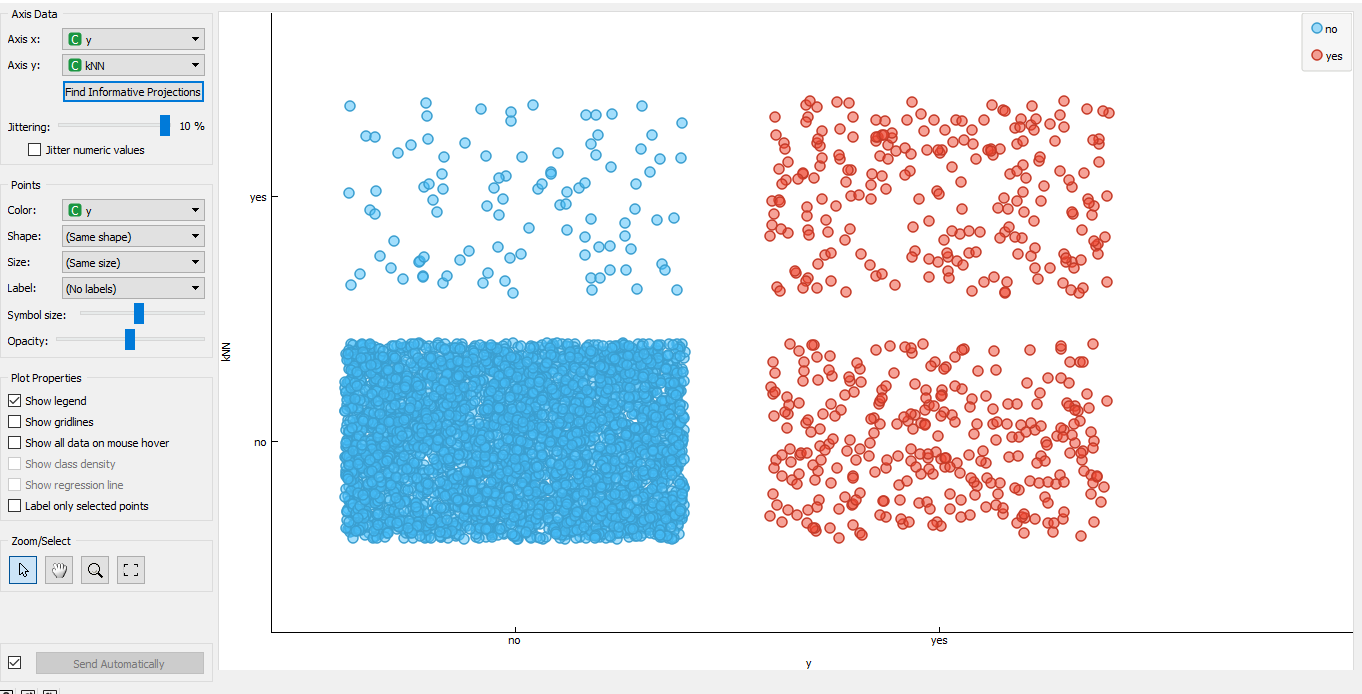
Neural network process records one at a time, and learn by comparing their classification of the record (i.e., largely arbitrary) with the known actual classification of the record. The errors from the initial classification of the first record is fed back into the network, and used to modify the networks algorithm for further iterations.

**ACCURACY WITH NEURAL NETWORKS:**

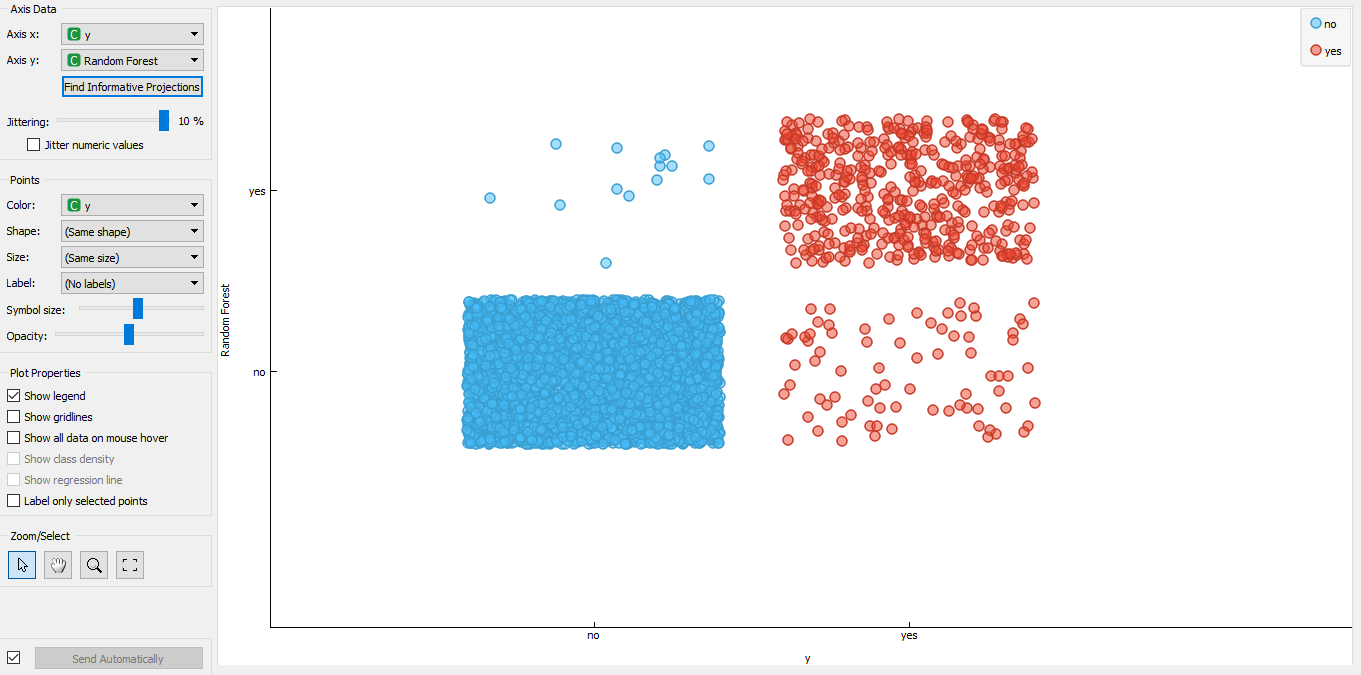
****

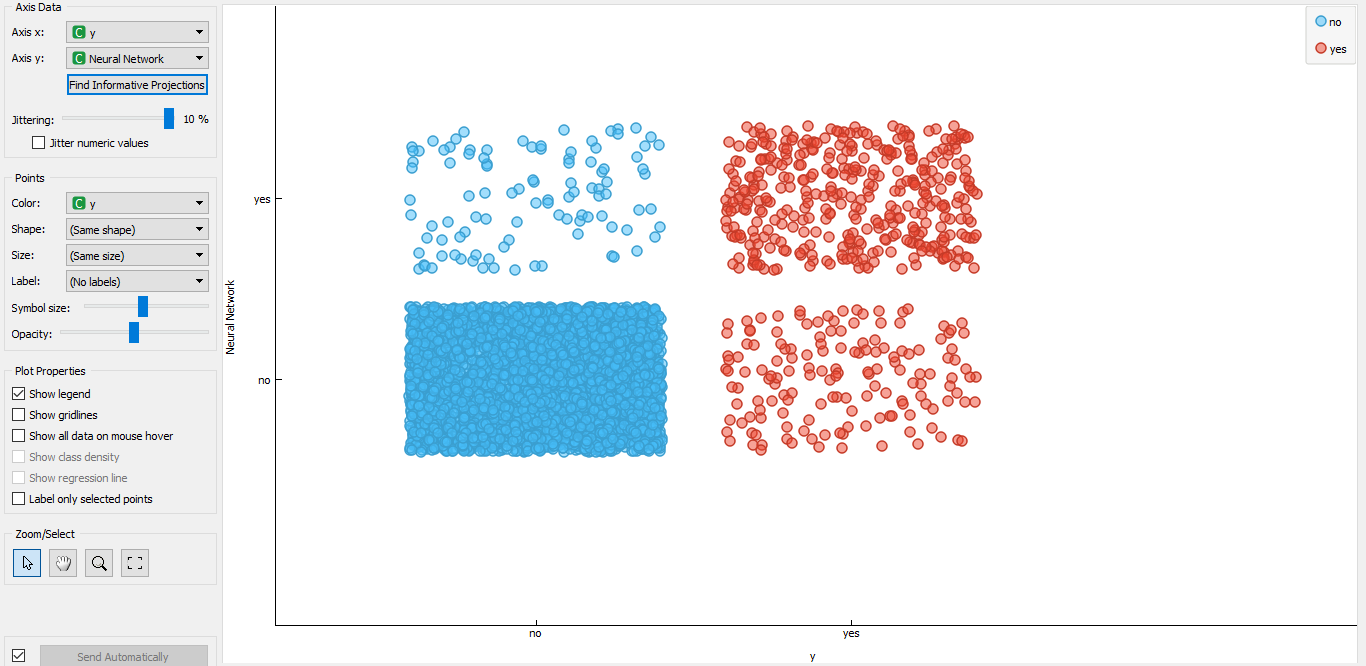
**SCATTER PLOT GRAPHS OF DIFFERENT CLASSIFICATION ALGORITHM**

**SVM SCATTER PLOT GRAPH**

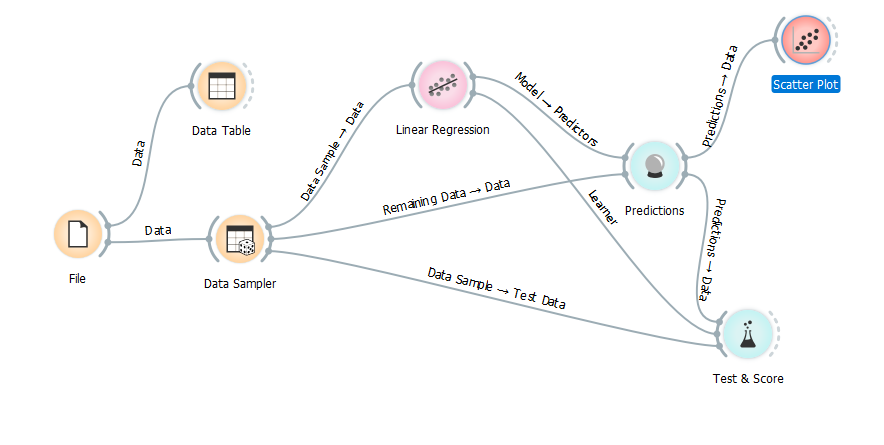
****

**KNN SCATTER PLOT GRAPH**

** RANDOM FOREST SCATTER PLOT GRAPH**

** NEURAL NETWORK SCATTER PLOT GRAPH**

1. **REGRESSION MODEL:**

****

**Inputs**

**Data**

 input dataset

**Preprocessor**

 preprocessing method(s)

**Outputs**

**Learner**

 linear regression learning algorithm

**Model**

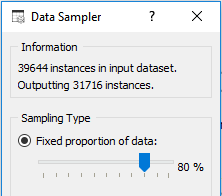
 trained model

**Coefficients**

 linear regression coefficients

**Linear regression works only on regression tasks.**

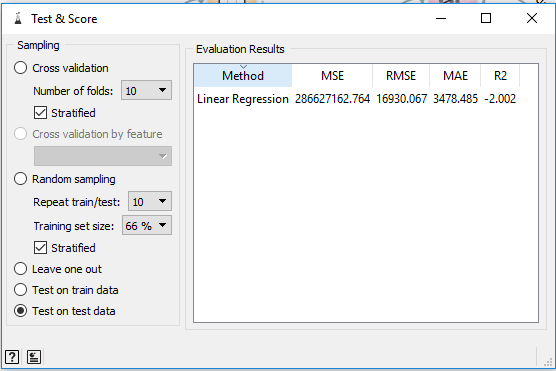
The **Linear Regression** widget constructs a learner/predictor that learns a linear function  from its input data. The model can identify the relationship between a predictor xi and the response variable y.

****

The **Data Sampler** widget implements several means of sampling data from an input channel. It outputs a sampled and a complementary dataset (with instances from the input set that are not included in the sampled dataset). The output is processed after the input dataset is provided and Sample Data is pressed.

Here we have a file with 39644 instances and using data sampler with ratio 80:20 we divide input file into two parts i.e. training file and test file.

Training file have 31716 instances and remaining instances for test file.

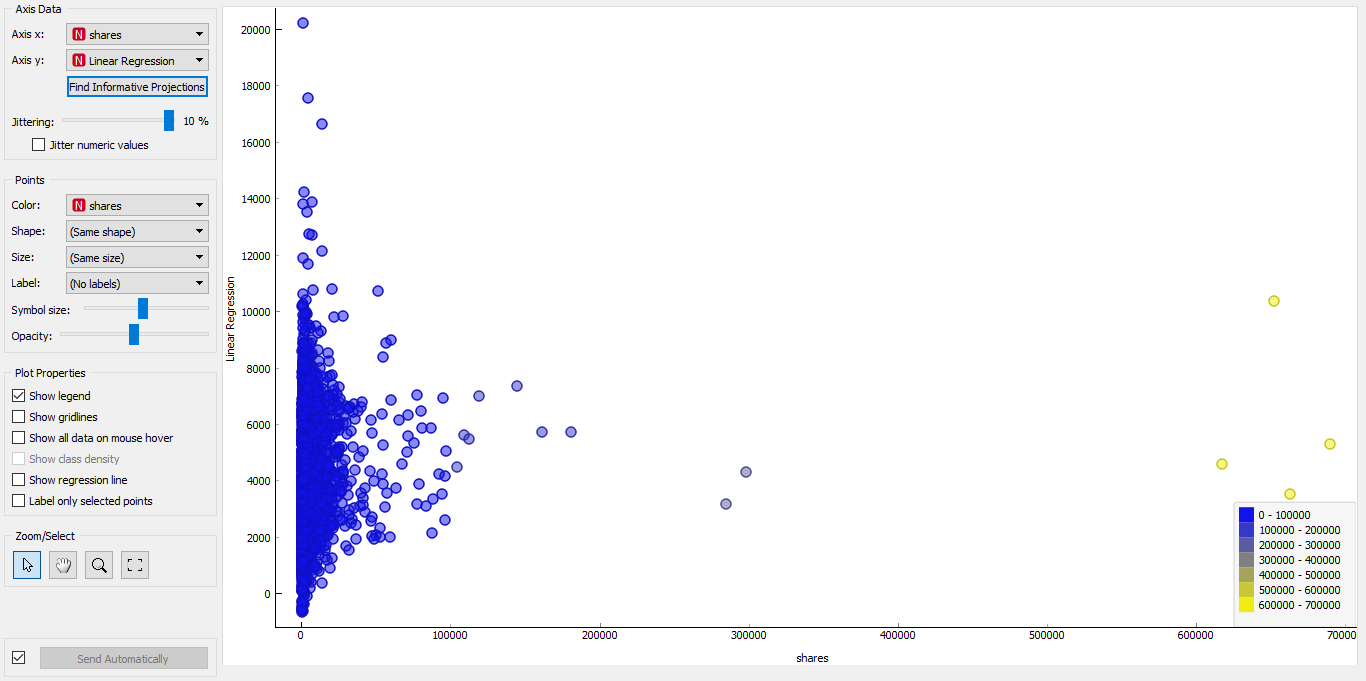
****

**Results of testing Linear regression algorithm is shown above**

Some important terms :

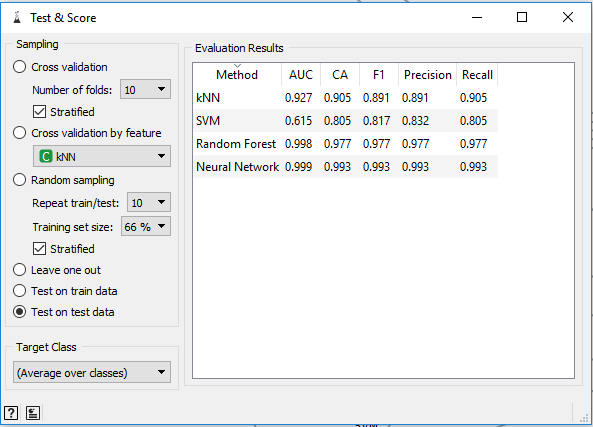
* [MSE](https://en.wikipedia.org/wiki/Mean_squared_error) measures the average of the squares of the errors or deviations (the difference between the estimator and what is estimated).
* [RMSE](https://en.wikipedia.org/wiki/Root_mean_square) is the square root of the arithmetic mean of the squares of a set of numbers (a measure of imperfection of the fit of the estimator to the data)
* [MAE](https://en.wikipedia.org/wiki/Mean_absolute_error) is used to measure how close forecasts or predictions are to eventual outcomes.
* [R2](https://en.wikipedia.org/wiki/Coefficient_of_determination) is interpreted as the proportion of the variance in the dependent variable that is predictable from the independent variable.

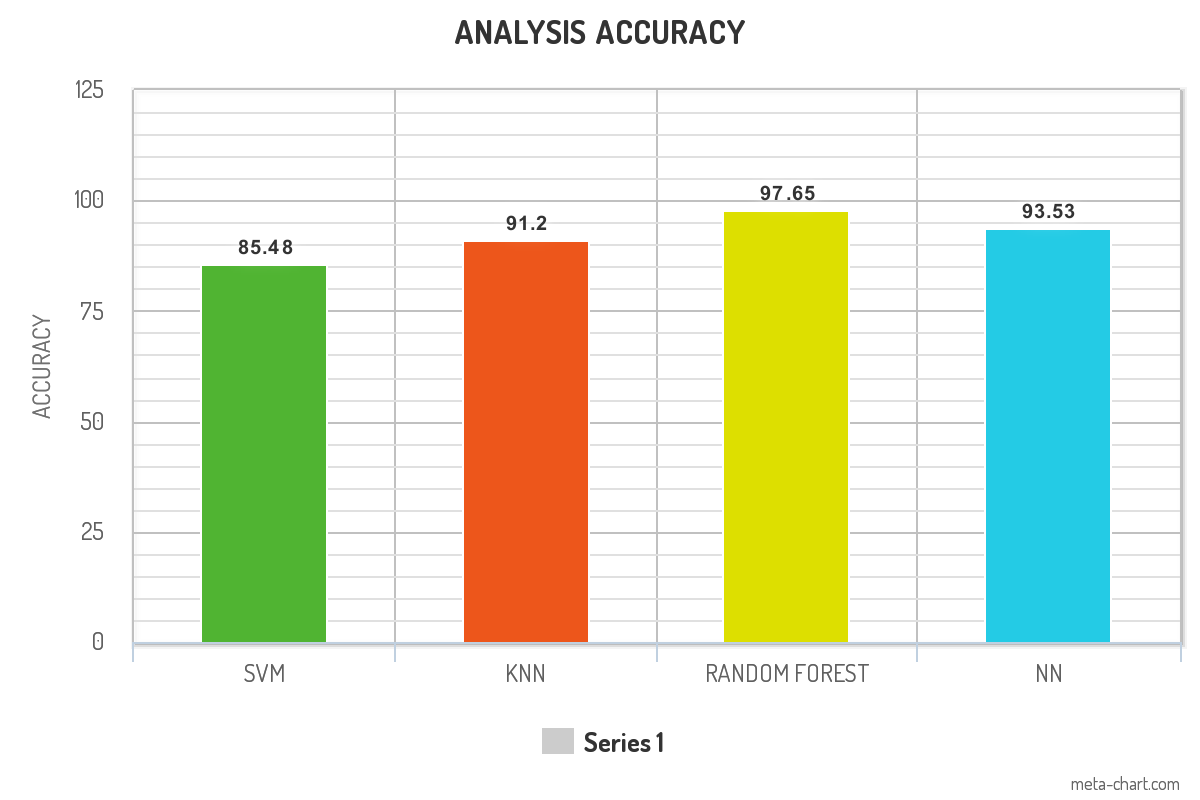
**SCATTER PLOT OF LINEAR REGRESSION MODEL**

****

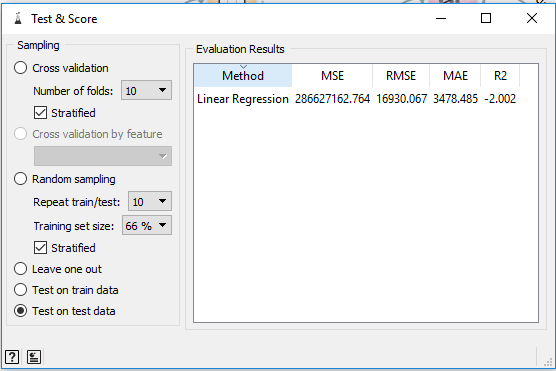
**RESULT**

**1.1 RESULT OF CLASSIFICATION**

****

****

**1.2 RESULT OF REGRESSION**

****

**CONCLUSION**

**“A Study of Deep Learning technique using Data mining tool (orange)**

**For Online News Popularity using regression and”**

This project was undertaken under guidance of skilled professionals who led the team members to pursue a better prospect of Deep Learning

While studying about deep learning concepts, many algorithms should be studied like Support Vector Machine (SVM), Random Forest (RF), K- Nearest Neighbors (KNN), and their combinations can also be used to determine accuracy and error with testing data

In classification, we have two file one is training file and other is test file and we had passed both the file separately ,the training file is trained over different classification problems like knn , svm etc whereas testing file is directly predicted on the basis of above algorithm.

The best result was obtained by random forest with accuracy 97.65%.

In regression, Dataset is the data which is usually divided into training and testing datasets. Generally, dataset is divided in the ratio 80:20. It is because that training needs more data so that our proposed model can learn more and can predict best in testing data.

The variance of each variable is -2.002.

**REFRENCES**

**[1]** K. Fernandes, P. Vinagre and P. Cortez. A Proactive Intelligent Decision Support System for Predicting the Popularity of Online News. Proceedings of the 17th EPIA 2015 - Portuguese Conference on Artificial Intelligence, September, Coimbra, Portugal.

**[2]** S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014  
S. Moro, R. Laureano and P. Cortez. Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology. In P. Novais et al. (Eds.), Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121, Guimaraes, Portugal, October, 2011. EUROSIS. [bank.zip]

**[3] Bank Marketing Data Set** (classification)

https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

[4] **Online News Popularity Data Set(regression)**

https://archive.ics.uci.edu/ml/datasets/Online+News+Popularity

[5] <https://docs.orange.biolab.si/3/visual-programming/widgets/model/knn.html>

[6] <https://docs.orange.biolab.si/3/visual-programming/widgets/model/svm.html>

[7] <https://docs.orange.biolab.si/3/visual-programming/widgets/model/randomforest.html>

[8] <https://docs.orange.biolab.si/3/visual-programming/widgets/model/neuralnetwork.html>

[9] <https://en.wikipedia.org/wiki/Orange>

[10] <https://docs.orange.biolab.si/3/visual-programming/widgets/model/linearregression.html>

[11] <https://docs.orange.biolab.si/3/visual-programming/widgets/evaluation/confusionmatrix.html>

[12] <https://docs.orange.biolab.si/3/visual-programming/widgets/visualize/scatterplot.html>

[13] <https://docs.orange.biolab.si/3/visual-programming/widgets/evaluation/testandscore.html>