- The target function has discrete output values: Decision tree methods can easily extend to learning functions with more than two possible output values. A more substantial extension allows learning target functions with real-valued outputs, though the application of decision tree in this setting is less common.

3.1.4.2 When not to use decision tree

However, decision trees cannot be applied to solve all kinds of data set. Below is characteristics of data set and scenarios in which decision tree methods cannot be used; also, some weaknesses are involved and discussed (Rokach & Maimon 2014):

- Most of algorithms, such as ID3 and C4.5, require that the target attribute will have only discrete values.
- As decision tree methods use the "divide and conquer" method, they tend to perform will if a few highly relevant exist, but less so if many complicated interactions are existed. One of the reasons for this is that other classifiers can compactly describe a classifier that would be very challenging to represent using a decision tree.
- Another disadvantage of decision tree is that the over-sensitivity to the training set, to irrelevant attributes and to noise make decision trees unstable; this means small changes in one split close to the root can change the whole sub-tree below it. Because of small variations in the training set, the algorithm may choose an attribute which may not be the best one.
- If the data splits approximately equally on every split, then a univariate decision tree cannot test more than O(logn) features. This sets decision tree methods at a weakness for tasks with many relevant features.
- According to Quinlan (1996), the ability to handle missing data is an advantage for decision trees, but Friedman et al. (1996) believe that the efforts to handle those missing values is a disadvantage of these methods. According to Friedman el al. (1996), the correct branch to take is unknown if a feature tested is missing, and the algorithm must employ special mechanisms to handle missing values.

3.2 Random Forest

3.2.1 Background

Random Forest is an ensemble classifier of Decision Trees. The contributing principles for operating this particular classifier can be: Since a Decision Tree has low bias and extremely high variance, averaging several Decision Trees could be a method for reducing variation. By building a huge number of Decision Trees, it can fix the issues brought by overfitting of training sets. The error of this classifier can be assessed by using out-of-bag error. Specifically, the error of misclassification can be indicated in a designated diagram called confusion matrix. As TN and FP can be considered as an error, it enables the ability to command the actual accuracy of the model.

3.2.2 Variable Importance

Classifier like Random Forest can realign the order by the extent of importance of each attribute. The process of it could be done in a classification or sometimes a regression way. To define the extent of importance of variable a, there are several steps:

- Randomly distributing attribute “a”
- Recalculating the accuracy of attribute “a”
- If there is a big drop in accuracy of Random Forest, then attribute a is an important attribute.

Advantages of Random Forest can be listed as:

- Can assess the importance of attributes. We can discard the attributes which are relatively uncorrelated them we can make it concise enough for making an accurate prediction without wasting a lot of time.
- Can cope with a tremendous volume of data. As you can see, the data we are dealing with could be in an enormous amount which means the whole process is going to be time-taking and energy-consuming.
- Can generate a highly accurate classifier.
- Don’t require too many data preprocessing. The more sections a process has, the higher possibilities errors will occur. Simplicity is good.

Disadvantages of Random Forest are demonstrated below:

- There is a possibility of overfitting datasets.
- During data preprocessing, unrelated columns cannot be preprocessed.

3.2.3 Implementation – KNIME

For the first section, I trained the Random Forest Model. And the figure can be viewed below:

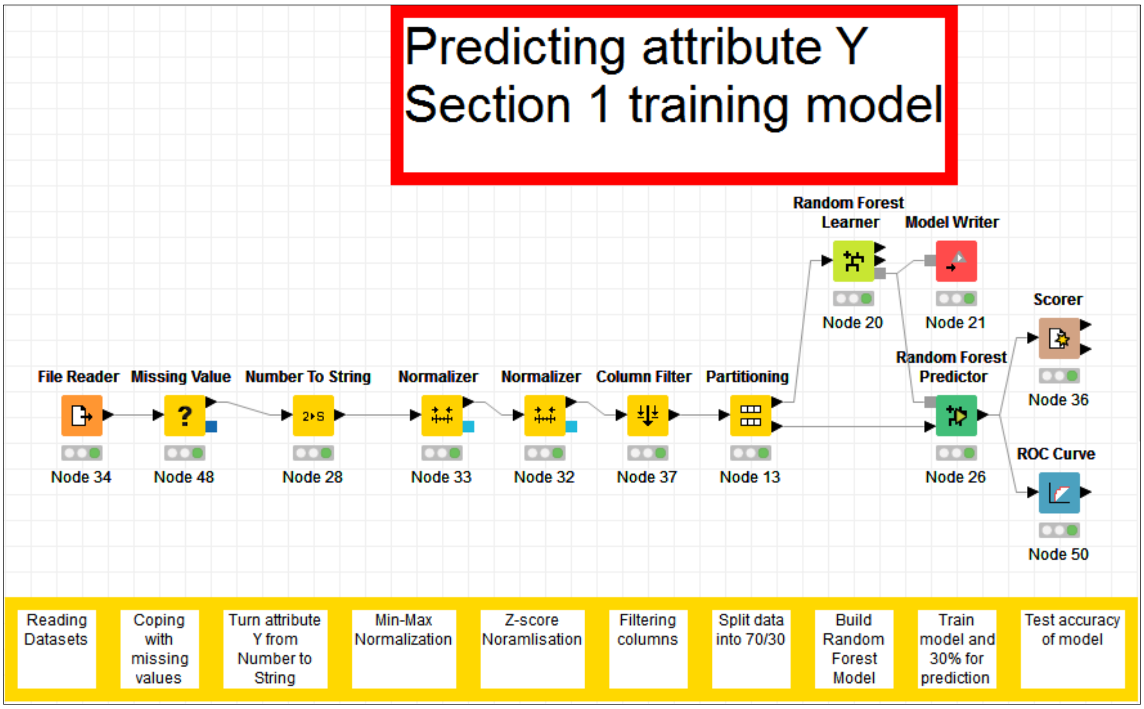
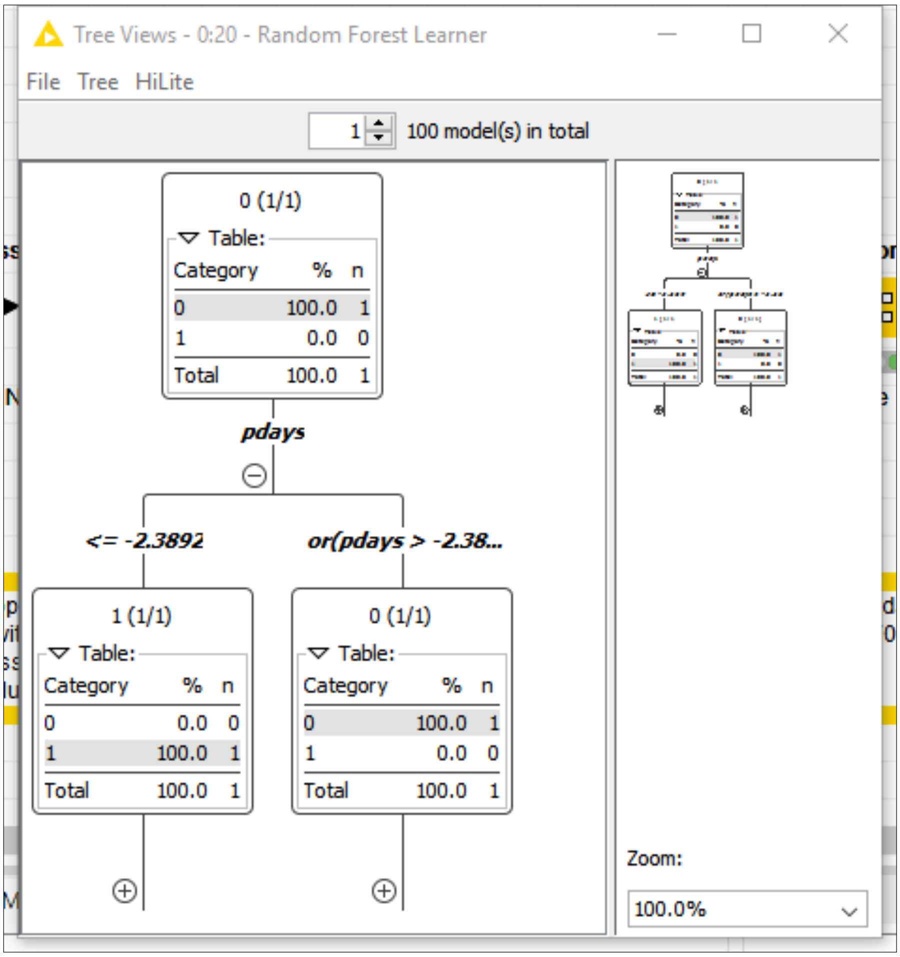


Figure 8: KNIME workflow for Random Forest classifier

In this section, we need to extract the datasets with a `File Reader` node. Then we need to do some data preprocessing to get rid of missing values. As we need to predict Y attribute while Y attribute is numeric and cannot be predicted, then turning it from numeric attribute into a string is a must. In order to deal with noisy data, normalisation must be implemented. Process Then we get the columns filtered. Spliting the datasets with 70 percent to upper row and 30 percent to lower row. For the upper row, as we have trained the model from `Random Forest Learner` node coming with model written by Model Writer.

Specifically explicating the node of `Random Forest Learner`, tree views is a critical role of visualising the model you have established. There is an example for tree views.



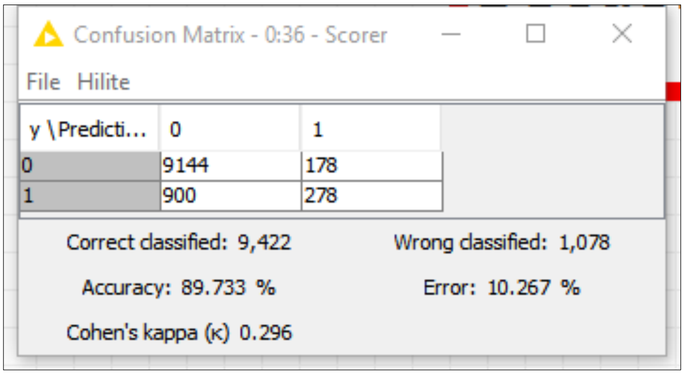
Category	%	n
0	100.0	1
1	0.0	0
Total	100.0	1

Category	%	n
0	0.0	0
1	100.0	1
Total	100.0	1

Category	%	n
0	100.0	1
1	0.0	0
Total	100.0	1

Figure 9: Decision Tree (1/100) used in the Random Forest classifier

What need to be highlighted is the consequence of executing Scorer, which is demonstrated as following:



y \ Predicti...	0	1
0	9144	178
1	900	278

Correct classified: 9,422 Wrong classified: 1,078

Accuracy: 89.733 % Error: 10.267 %

Cohen's kappa (κ) 0.296

Figure 10: Confusion matrix for the Random Forest classifier

For the second section, by using the model we have trained from the previous section, operating with the test.csv. The whole process can be viewed below.

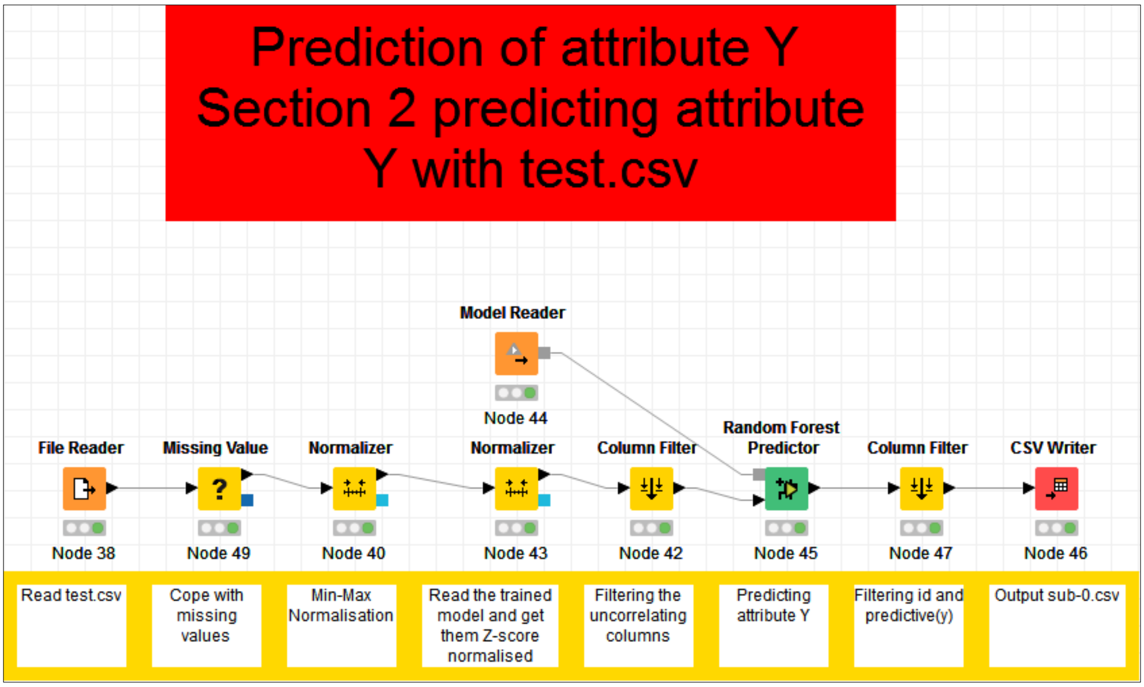


Figure 11: KNIME workflow for creating Kaggle® submission with Random Forest classifier

In this section, I have the test.csv read first then let it go through data preprocessing like filling missing values, getting it normalised and get the uncorrelated column filtered. After having a prediction on this model, we get attribute like id and prediction(y) included and output by a `CSV Writer` node.

3.2.4 Results

Table 2: Results of Random Forest classifier by attribute

Input Attribute	Result
`age`	● The mean value age of our clients is approximately around 39. ● Ranged from 22 years old to 60 years old, most of them didn't make their deposit into our company. ● For those who made their deposit, they are mostly around 22 years old to 64 years old. ● Most of our clients didn't have their money deposited
`job`	● Retired, technicians and admins showed their greatest interest in

	subscribing a term deposit
marital	● Married and single people display us an intense interest in making a deposit, while the one who is divorced doesn’t seemingly interested in it.
education	● Customers who have a degree ranked from university, high school, professional courses, basic 9y, basic 6y and basic 4y have their bank account subscribing a term deposit. ● The greater extent of education they have, the higher chances for them to make a subscription. ● Illiterates don’t have their habit of making a deposit, comparing with those who have been educated.
default	● For those who have credit will mostly accept their subscription. ● Comparing with people with credit in their bank account, those individuals who don’t have their credits comprise a small percentage of making a subscription.
housing	● Having a housing loan, those people who make their subscription possessed a superior percentage than those who don’t.
loan	● Without having a loan, people who say yes to making a term deposit comprise a significant percentage than those who don’t
contact	● For those people who would like to create a subscription, cellular dial is more popular than telephone
month	● Doesn’t seem too much correlated. Because data has been evenly distributed
day_of_week	● Mostly concentrating on Monday, Tuesday, Thursday, and Friday.
campaign	● The people who want to make the deposit mainly make their dials ranging from 1 to 7
pdays	● For those people who have contacted us, 15 days before have a higher chance of making a deposit ● People who don’t even contact us have great chance of declining our offer.
previous	● For people who previously contacted us at one time, have the highest chance of making a term deposit.

`poutcome`	● If the outcome is a successful one, the higher amount of users will make a subscription. If outcome is a failure, lower amount of subscription
`emp.var.rate`	● Rate between -0.4 to -3.4 have a high probability for making term deposit
`cons.price.idx`	● Correlations are not clear
`cons.conf.idx`	● Correlations are not very distinct
`euribor3m`	● Correlations are not significant
`nr.employed`	● 5113.6 to 4963.6, people in this interval will likely to make their deposit

3.2.5 Inferences Based on Analysis

After having to analyze the data with Random Forest Technique, the pattern of identifying the particular group of people who will make a term deposit will be defined. Several characteristics of determining pattern can be listed as below:

- For the age attribute, most of their age are notably concentrating between 22 to 60.
- For the job attribute, most of them are technicians, retired and admins.
- As we are talking about marital status, an enormous amount of them should be married or single.
- Education that they have is a determining factor of whether the chance of making a term deposit is high or low. From the highest level of education like the University to the lowest level of education like basic.4y, the possibilities of making a term deposit is decreasing even though they are still an excellent chance of making a deposit. Illiterates don't have a habit of making deposit
- For default, if people have credits, most of them will make a deposit.
- For housing, if people have a housing loan, they will have a greater chance of subscribing a deposit
- If people have loan, they will make a deposit in higher chance
- People who likely to use cellular has greater chance than telephone
- Those people frequently make their dial on Monday, Tuesday, Thursday and Friday.
- Their dials' times focuses on one to7.
- Most of them have contacted us 15 days before the campaign
- People who previously contacted us one time or haven't had a greater chance.
- Higher chances seemingly trend on a successful campaign.
- Emp.var.rate should be -0.4 to -3.4

- Some employees should be various from 5113.6 to 4963.6.

Within those characteristics, it can be presumed that clients who have those conditions will have the highest chance of making a successful term deposit.

3.2.6 Summary

3.2.6.1 Suitable usage

Suitable usage of Random Forest must meet the requirements as followed:

- The datasets you are going to preprocess don't require too much preprocessing like data cleaning, binning or normalisation. And this is explained by its own characteristics
- Datasets is in a tremendous capacity. For instance, the datasets we have analyzed has 35,000 data points.
- Random forest possesses the ability to classified uneven data and missing values with low classification error rate.

3.2.6.2 Unsuitable usage

Unsuitable for using Random Forest

- Most of columns/attributes in the datasets are uncorrelated or just has limited pertinent features which is not sufficient for deducing a precise prediction.
- Datasets are not suitable for the algorithm that doesn't comply with its characteristics.

3.3 Naive Bayes

3.3.1 Background

Naive Bayes is a simple and efficient method for constructing classifiers: models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. it includes a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of independence between every pair of features.

For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.

Abstractly, naive Bayes is a conditional probability model: