By

Satadipa Sarkar

Data Mining Project

Seattle Pacific University

ISM-6359 Data Mining

1. The data mining tool I have used – Orange.

Orange is a component based visual programming software for data mining, machine learning and data analysis. Workflows in the canvas are created by linking predefined or user defined components called widgets. The reasons why I chose Orange to solve my data mining problem are:

* Visual programming, no need of writing any code, everything is presented visually.
* Multitude of machine learning algorithm.
* It allows classification and clustering. The business problem I have analyzed is a classification problem where I had to train a dataset to predict if patients who have registered for a doctor’s appointment will show up or not. And then if not, what are the determining factors (informative variables) for not showing up.
* It is free / open source and got easily downloaded on my Mac.
* Interactive data visualizations widgets which helped me to visualize and interpret the results of the prediction easily.
* Functionality can be added through add-ons.
* I could write Python scripts to clean and transform the data in proper format which took lesser time as compared to using tool widgets to clean the data (ex: finding outliers). It has a built in Python library.

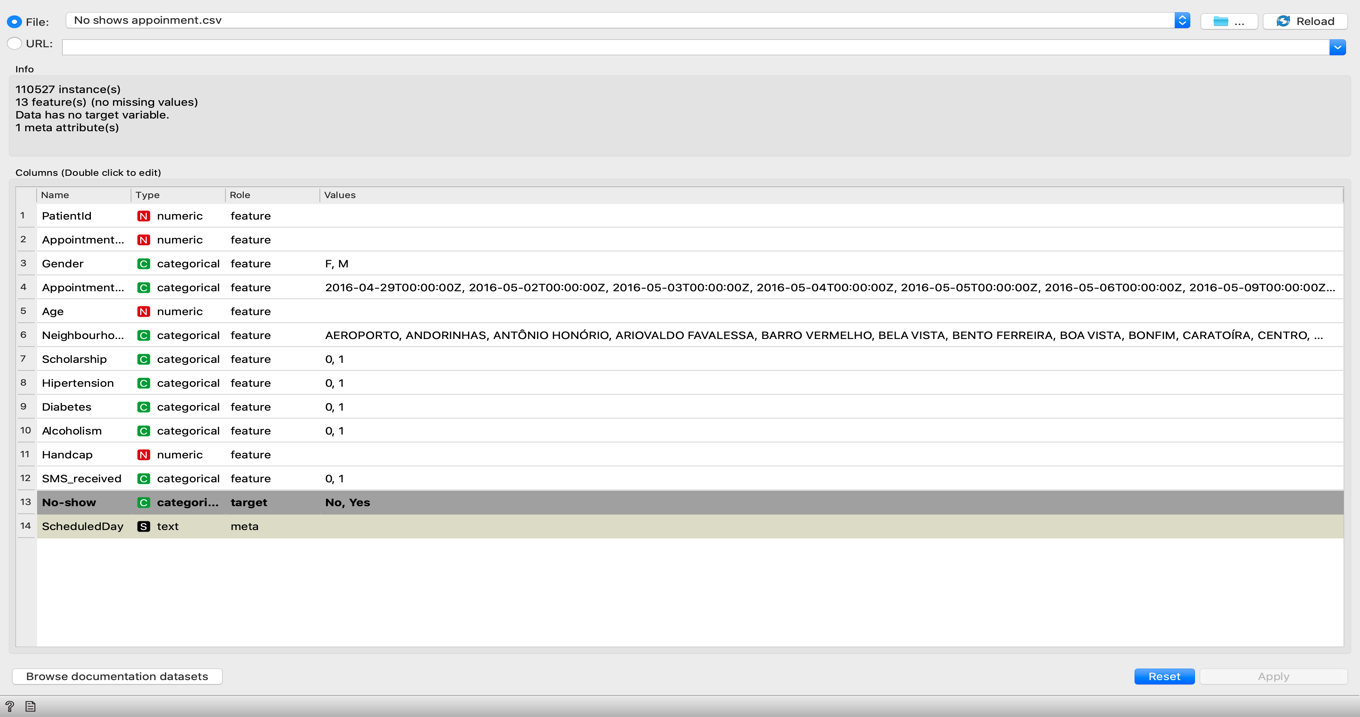
2. Feature of the data set:

The data set I used to predict the appointment no-shows have:

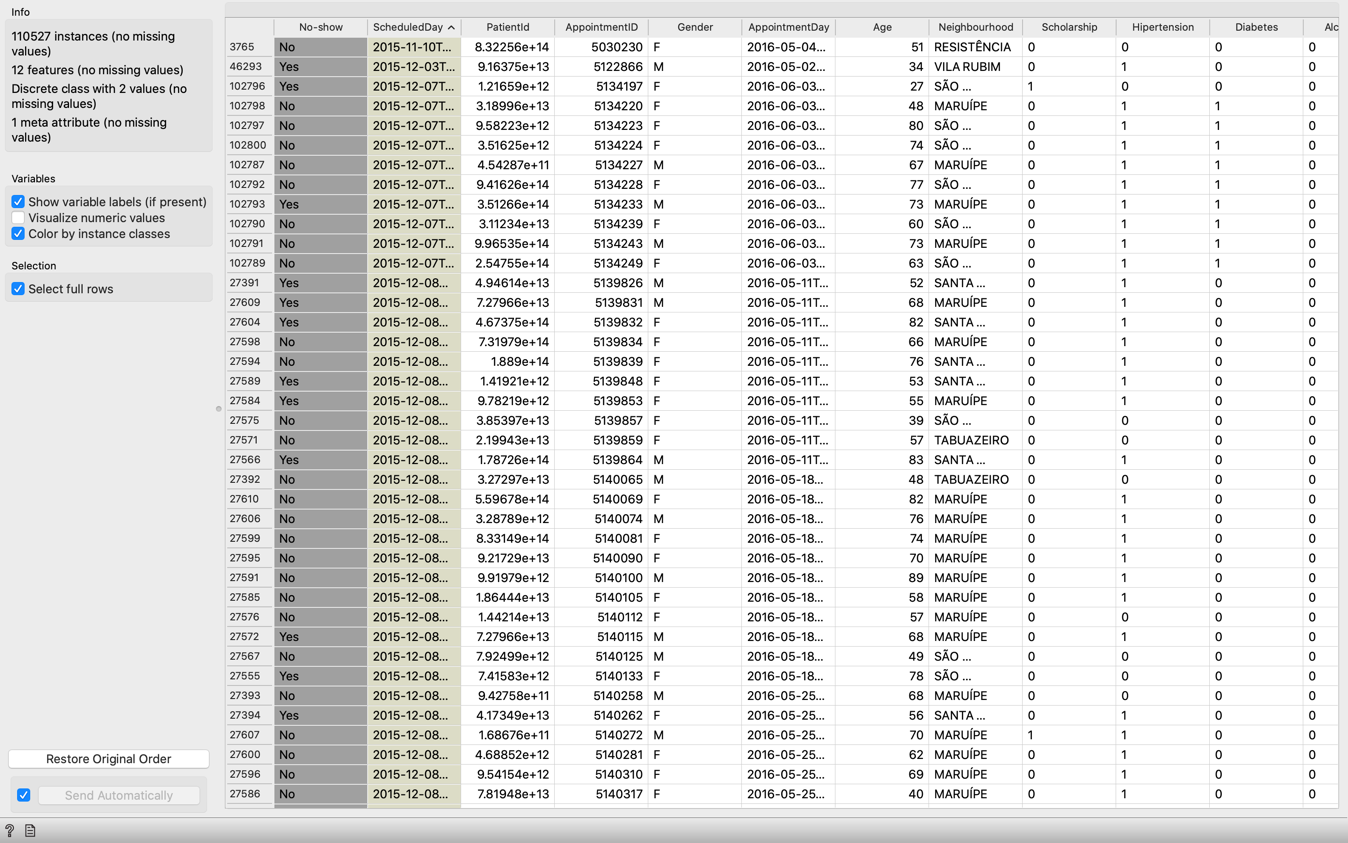
* Rows: 110527
* Columns: 14

3. Widgets I used to predict the appointment no-shows are as follows:

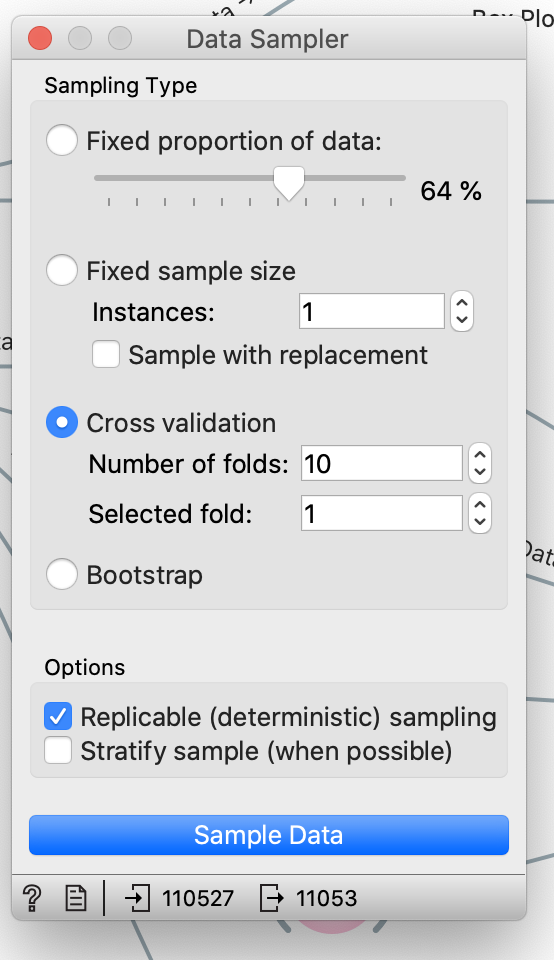
1.File – To load the data into the Orange workspace. If we double click on the file widget it opens and asks for the filename. My filename had .csv extension, although Orange is compatible with other formats like .txt etc. Once the dataset got loaded, select the target variable or label – in my dataset it was No-show. The data types of other columns can be changed at this time too.



2.Data Table - Once the target variable is set and the data is loaded in the Orange Canvas, I have used the data table widget to view the data in table format and understand the columns better. It also gives information about missing values and rows in the data set. In my data set there was none.

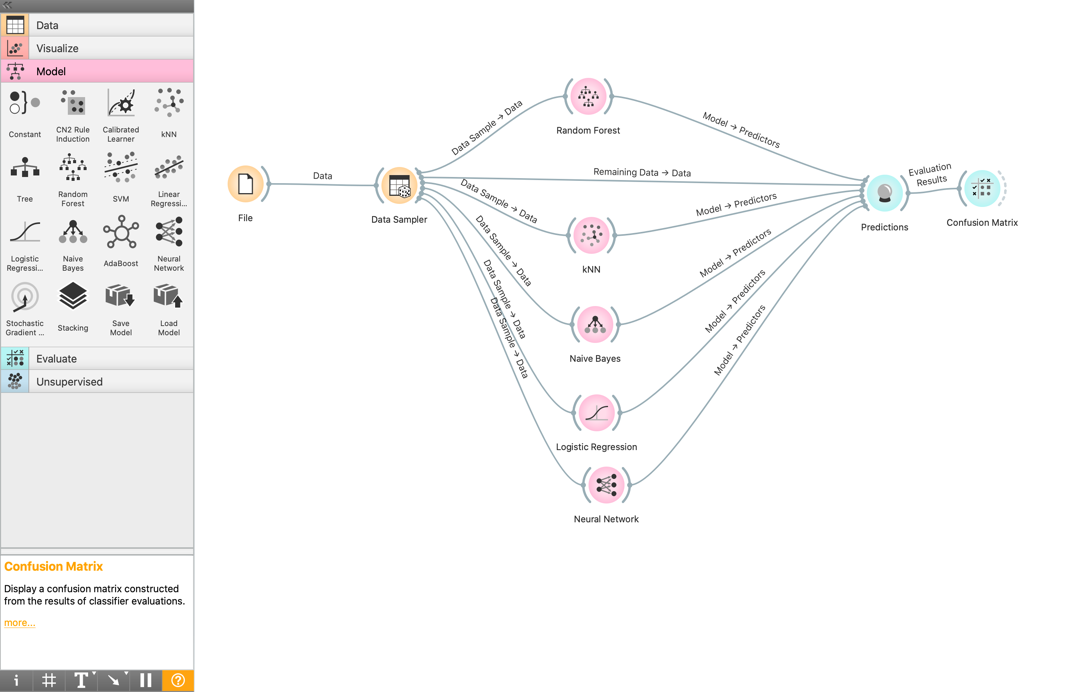


3.Data Sampler – The data sampler widget was used to sample the data into training and test data set. In my case I used a cross validation with 10 folds but there are other options like split ratio or boot strap (makes the process much slower).

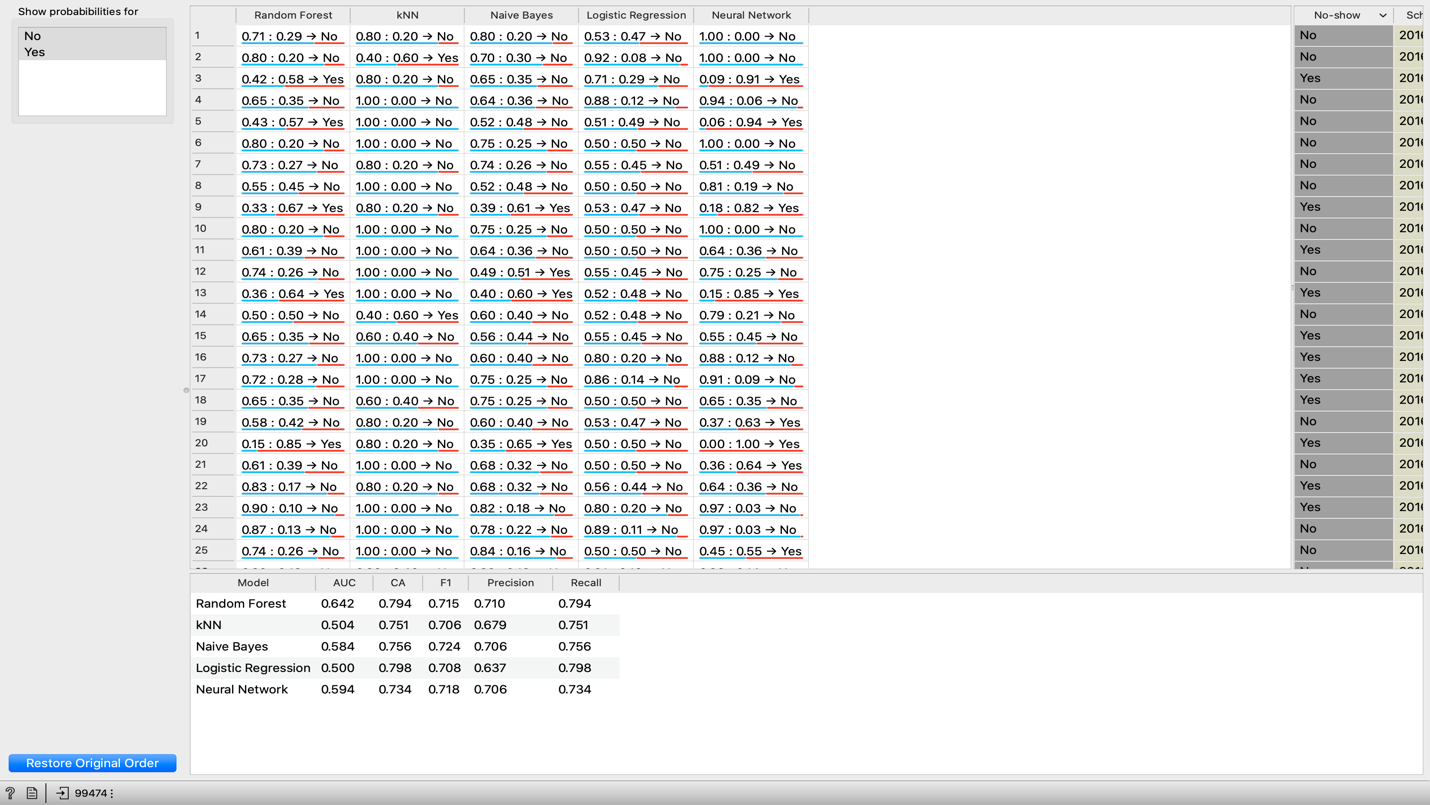


4.Learners / Model widgets – I have used various learners to evaluate and predict the precision of various classification algorithm ex: Random Forest, Naïve Bayes, kNN, Logistic Regression to find the most efficient model without any data preparation.

5.Predictions – The prediction widget predicts the model’s precision. The input (training data set from the data sampler) and the remaining test data set.

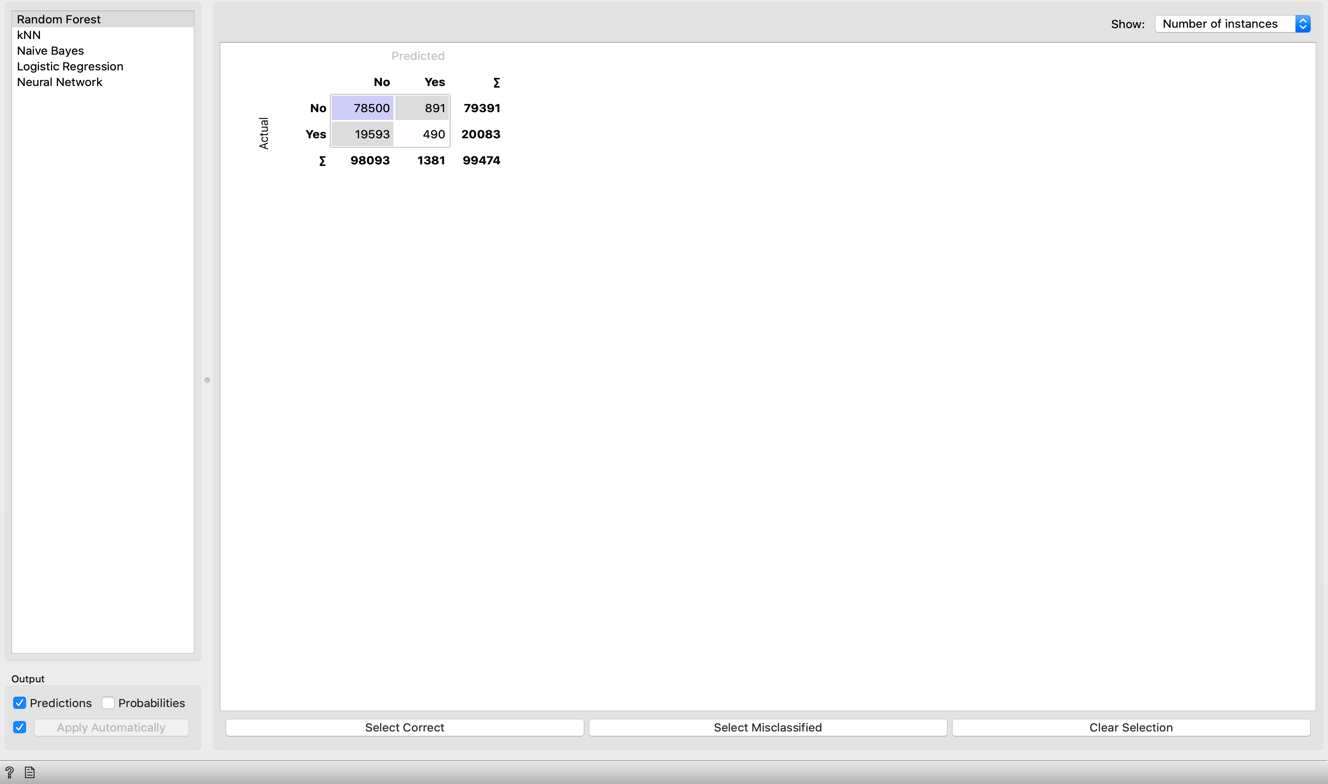


Once you double-click on the prediction widget it shows the precision and classification accuracy (CA) of the models in predicting the no-shows for the appointment (Yes/No).



From the screen shot we can see that Random Forest and Naïve Bayes have better classification accuracy and precision than other models.

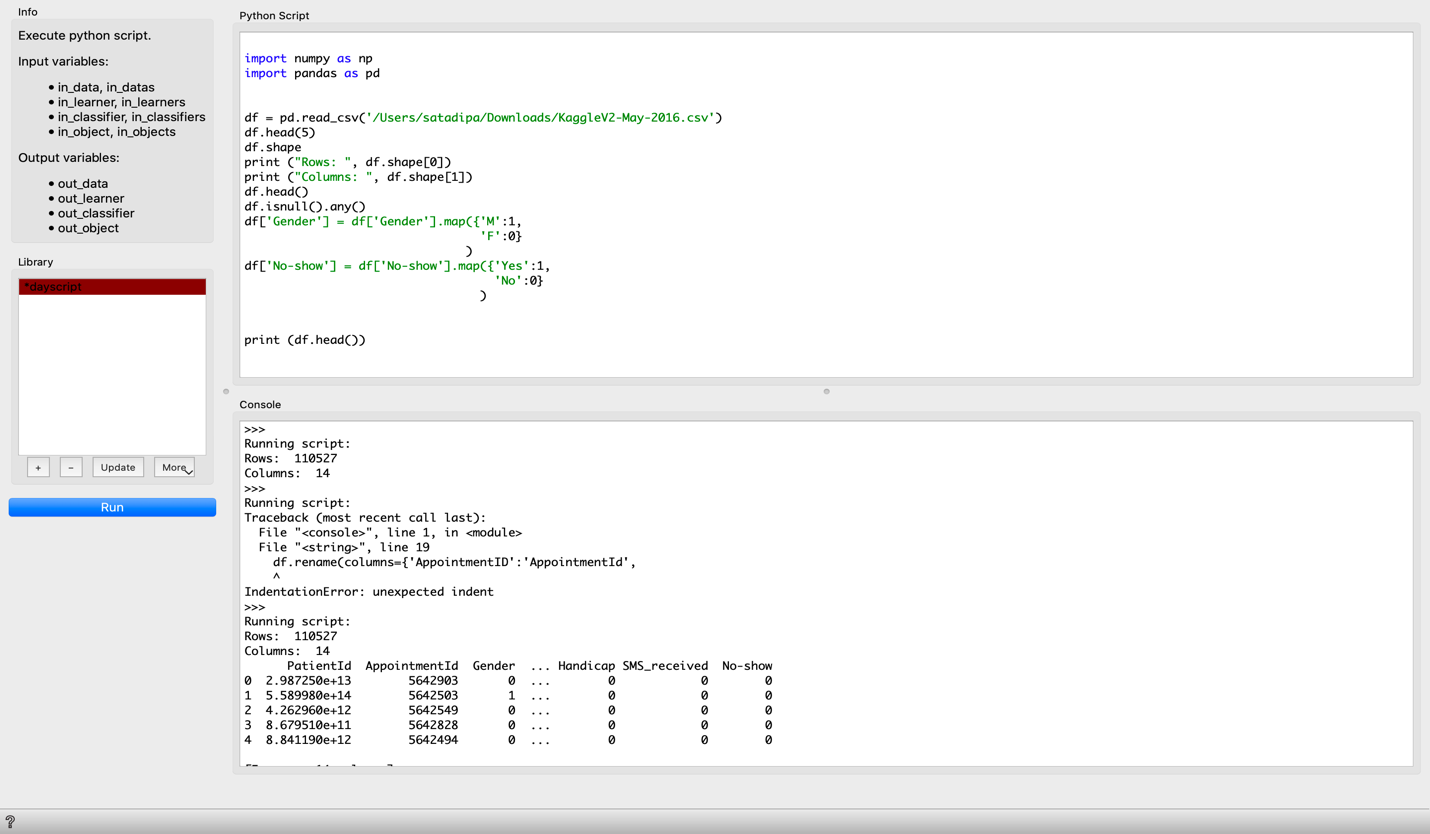
6.Confusion Matrix- The confusion matrix widget gives a better understanding of the correctly predicted and misclassified one’s in the form of a table called Truth table.



Data Preparation:

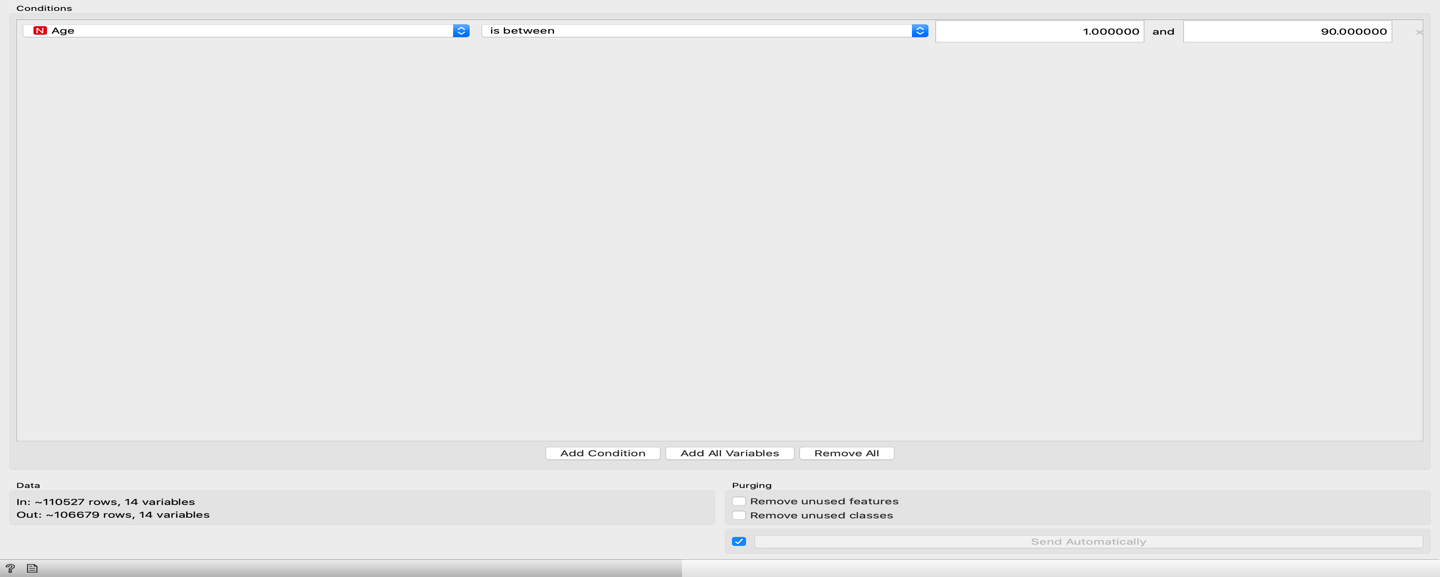
7.Select Columns – The “No-show appointment” data set didn’t have any missing values so there was no need of imputing any data. But the data set had columns like Patient Id and Appointment Id which did not seem to contribute much in identifying the target variable. So, I have selected the columns except the two mentioned.

8.Python Script – The Python script widget helped to write the script to understand the data columns more exhaustively like: the age data ranges from -1 to 115. The appointment date and the scheduled date was converted to datetime. Gender and No-shows column were converted to Boolean from string. The following python code was used to understand the shape of the data set and map Gender and No\_shows as 0 & 1.

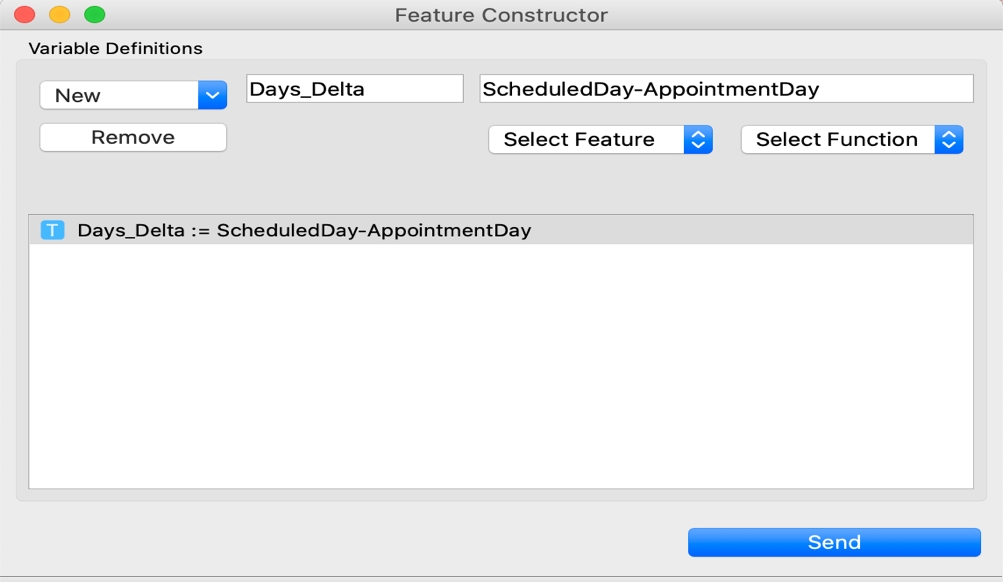


9.Outlier – The youngest patient is -1 years old and the oldest 115 years old. The outlier widget showed age as an outlier in the machine learning algorithm.

10.Select Rows – The select rows widget was used to select rows where age is between 1 and 90.

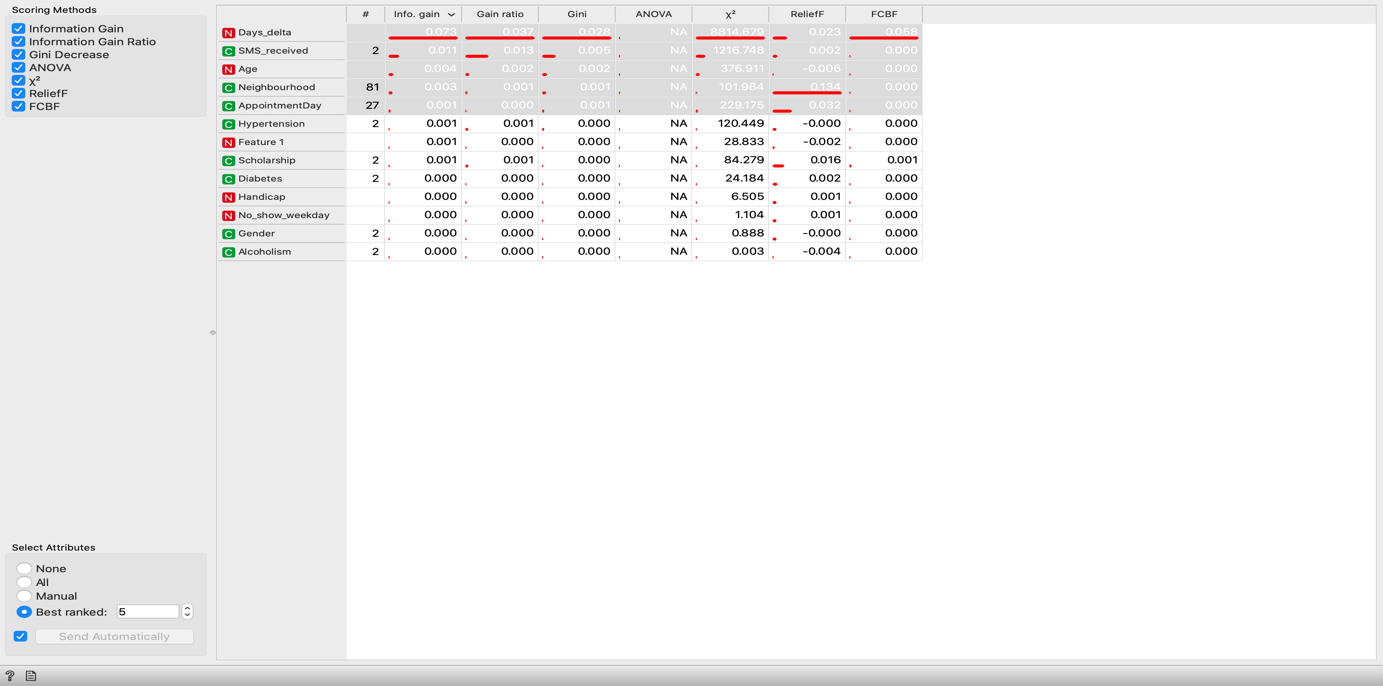


11.Feature Constructor – The feature constructor attribute generates attribute from the columns available. We created a column called “Days\_delta” shows the approximate waiting date after the appointment was scheduled.

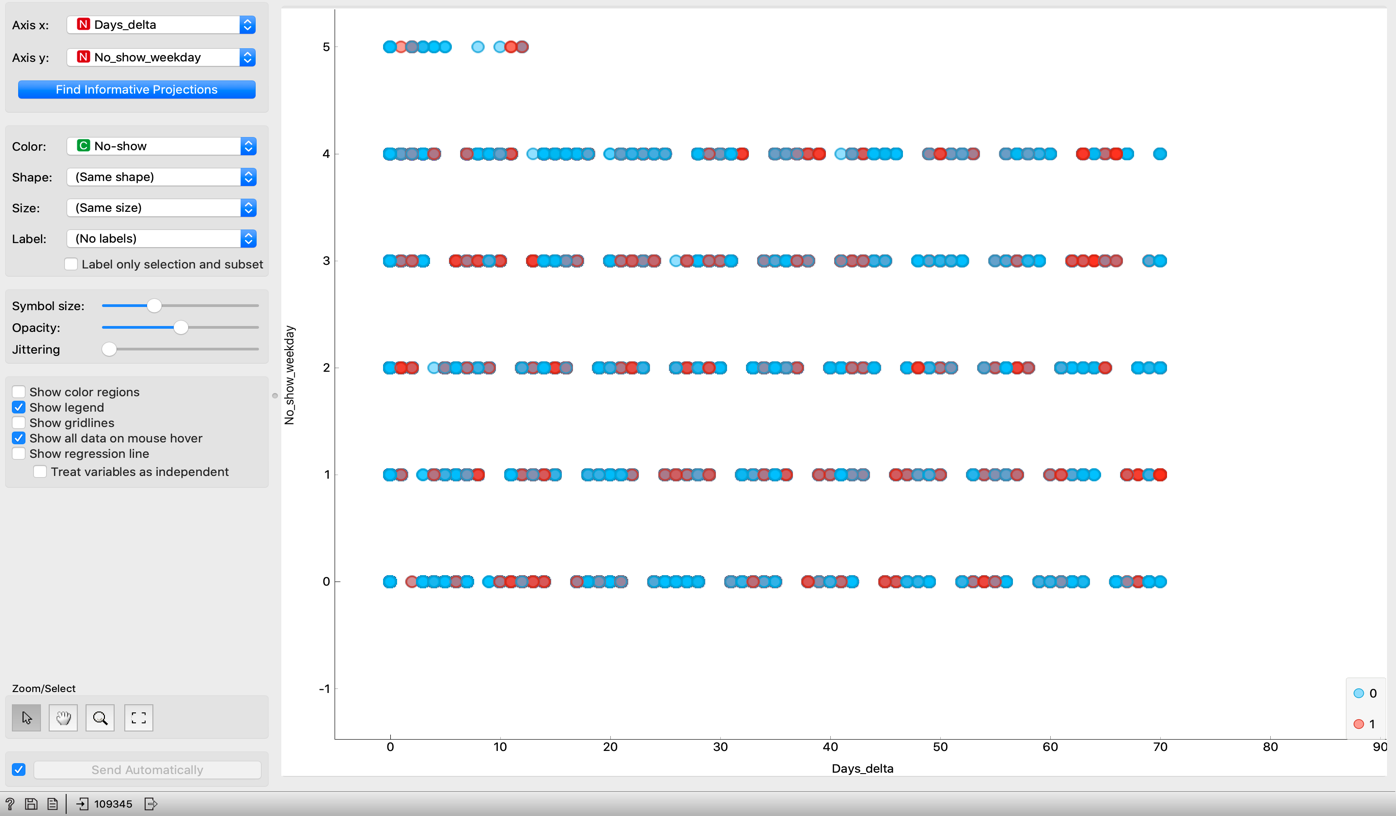


Data Visualization/ Exploratory data analysis:

12.Rank – The rank widget helped to rank the columns according to their correlation with the class. It helps to identify the important attribute. The output scores the top 5 attribute which is based on Information Gain, Gain Ratio, Gini, Annova index. The informative attributes for my dataset were: Age, days\_delta, sms\_received, neighborhood. These parameters are easier to intuitively understand and are more sensible to play an important role in predicting whether a patient is going to be a no\_show or not. A patient who gets a reminder before an upcoming appointment will at least remind those who have forgotten about it. Also, days\_delta i.e. no of days passed after they have initially made the appointment till the scheduled appointment date can be an important indicator because if too many days have passed then patient try to be a no\_show.

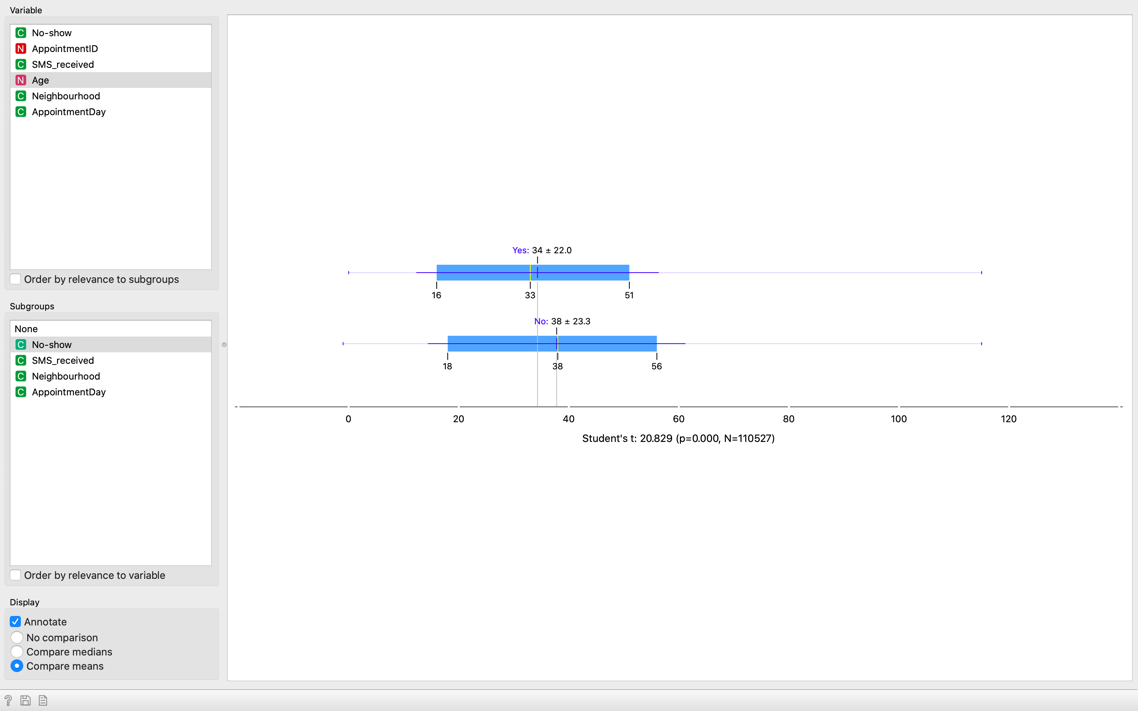


13.Scatter plot -The scatter plot widget was used to understand the relationship between no\_shows and days\_delta. It shows that it is between 0-70 and the rest are outliers and can be removed.

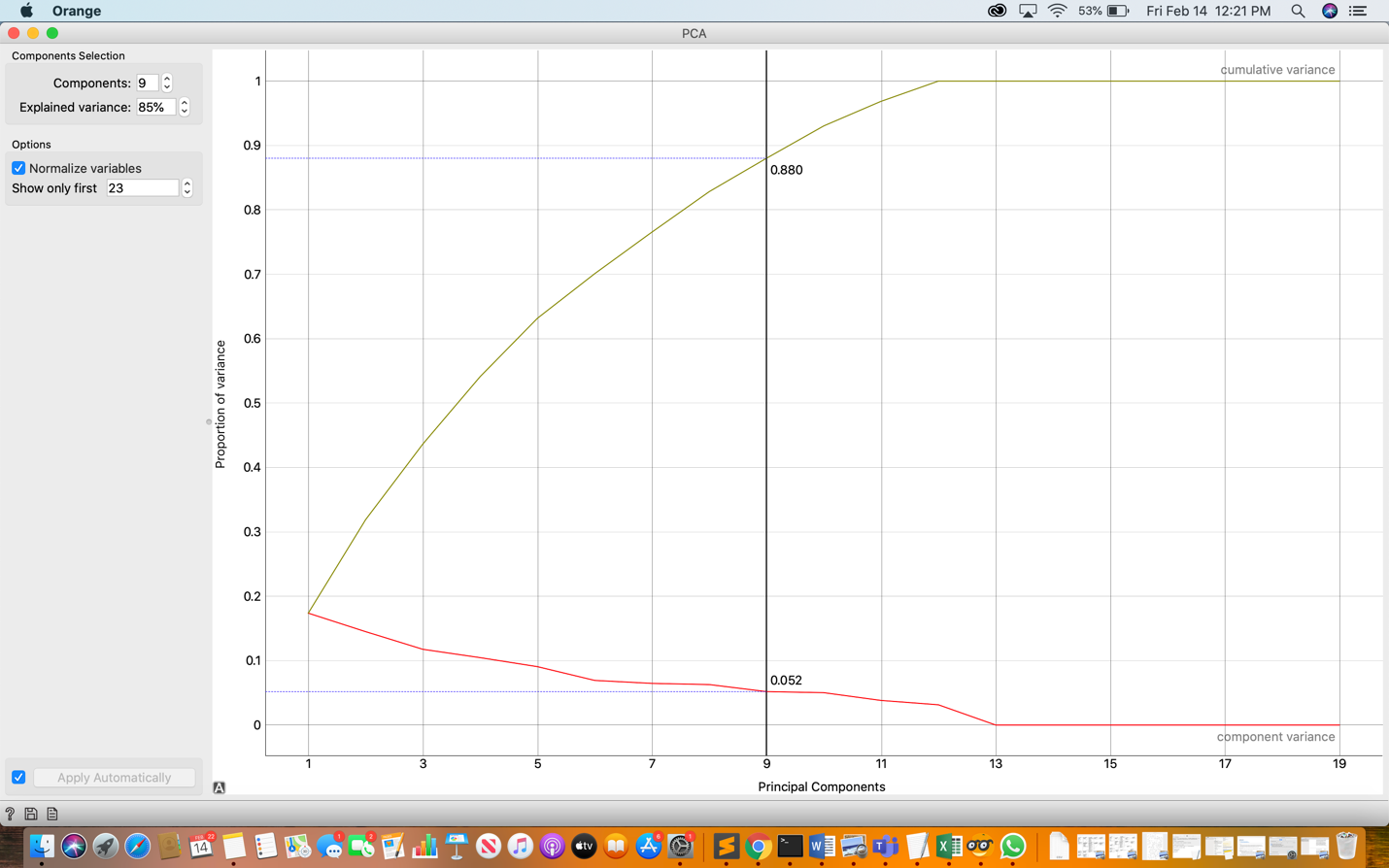


14.Distributions – The distribution widget helped with analyzing the data with histograms, charts, stack columns, and cumulative distribution.

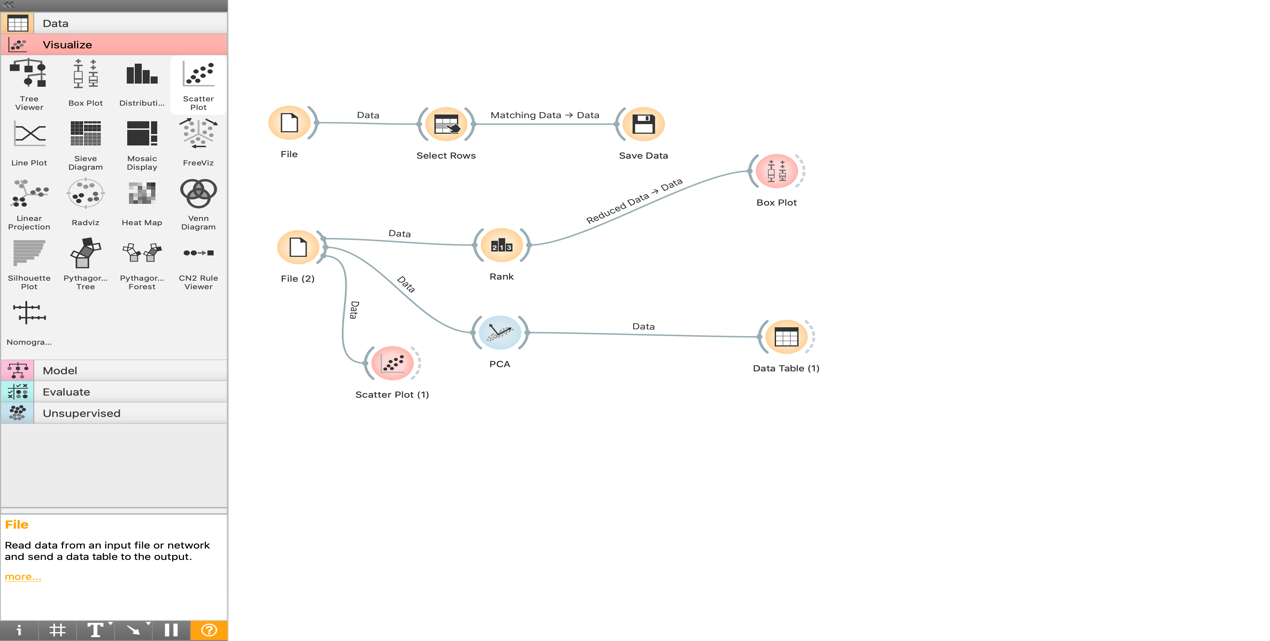
Box-plot – It is a method for graphically depicting groups of numerical data through their quartiles. The below diagram is the box plot of age and no\_shows:



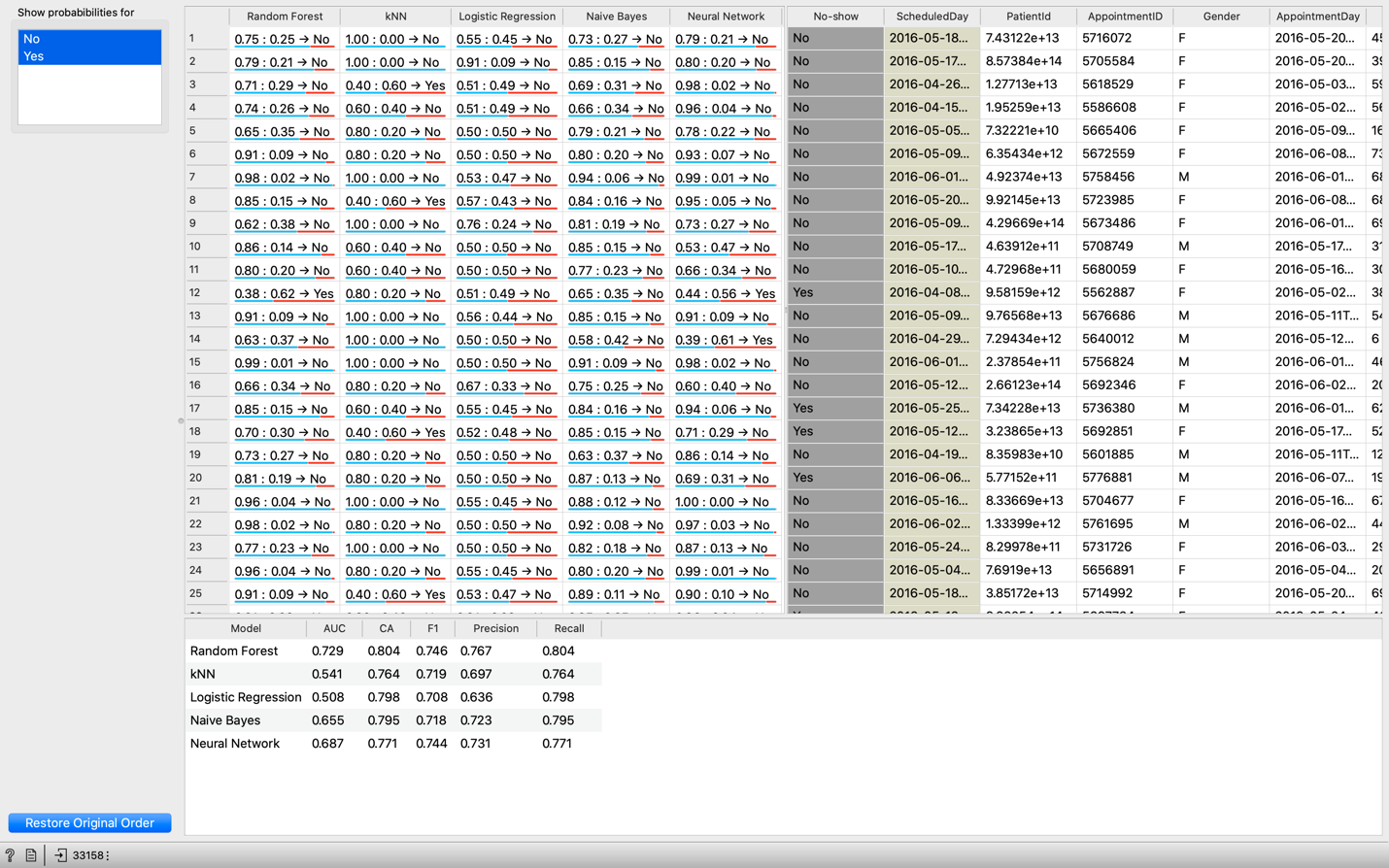
15.Principal Component Analysis: It is one of the most popular linear dimension data reduction method. It normalizes the data, calculates the covariance, finds eigen values and vectors and forms the new data set with reduced dimensions with top k values. In this data set 85% of the variance was explained by the top 9 components or variables.



The orange canvas for data visualization widgets used in my data mining project:



Results with data preparation:



Model Performance: Random Forest, with Classification Accuracy of 80.4% followed by Naïve Bayes with 79.5%.

3. Is it a classification or regression problem?

Ans: The data set was to predict whether a “show” or “no-show” in health care analytics. It is a classification model that will try to assign items to target class. The goal of classification is to accurately predict the target class for the data. The most important feature is to predict if a patient is going to show up to their medical appointment. There are various data mining algorithm like simple decision tree, Random forest, Naïve Bayes, kNN, Logistic regression, Neural Network etc. The data mining algorithm which had best performance and classification accuracy than the rest was Random forest with 10 folds and classification accuracy of 80.4%.

The data had a target variable called no\_shows with values yes/no, so a supervised learning algorithm would be a better fit. Supervised learning is the data mining task of inferring a function from labeled training data. In this case, there is a specific target value that is no\_shows that I would like to predict. The target value has two outcomes (Yes/No). In this there is a subset of data points for which the target value is already known and that data is being used to build the model which is then applied to the data for which the target variable is unknown.

3. Data Source:

Ans: The No\_show appointment data set is available in Kaggle. It is publicly available.

URL: <https://www.kaggle.com/joniarroba/noshowappointments>

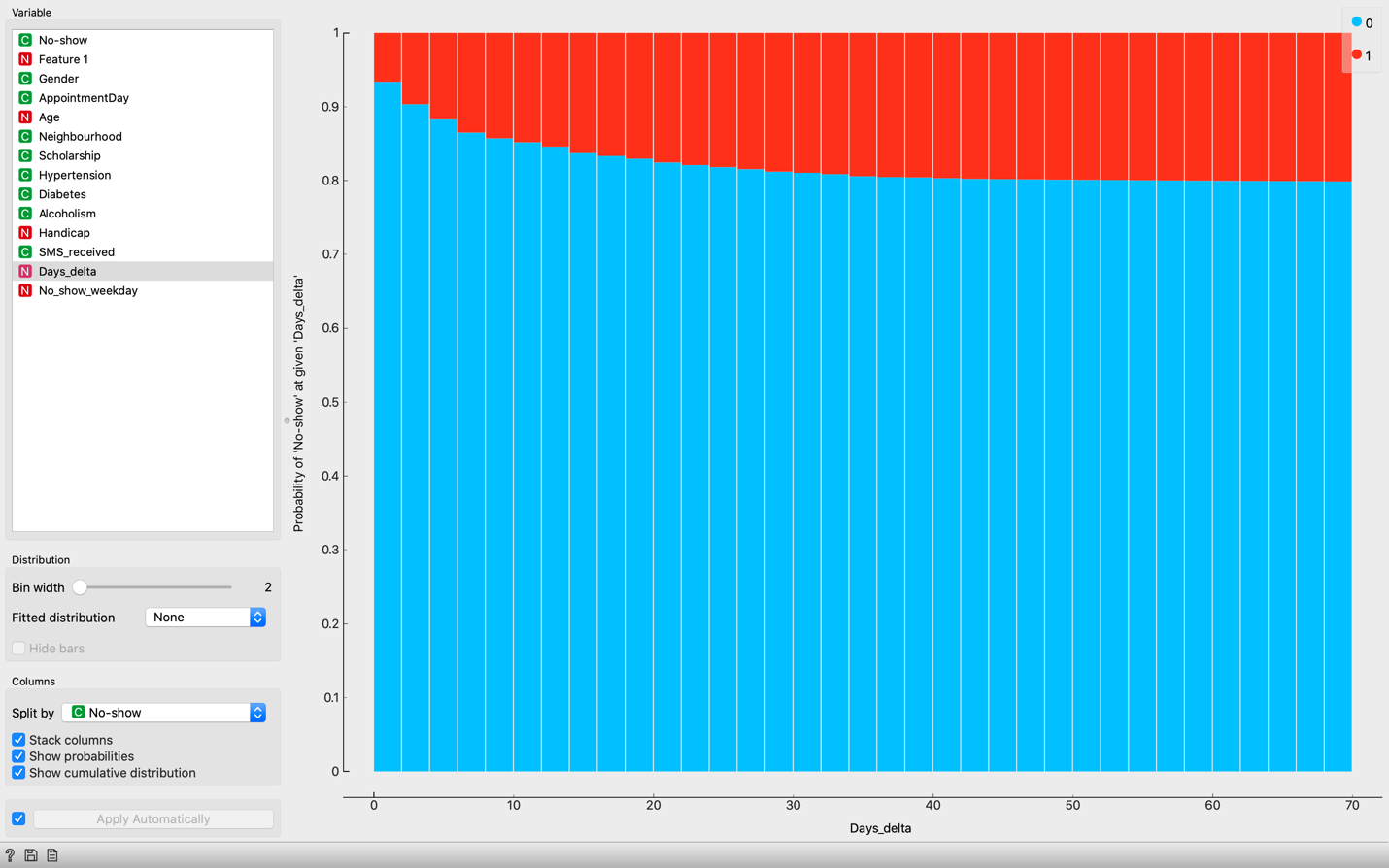
I chose this data set because the size of the data set isn’t very huge for analysis and this data had self-explanatory attributes or columns which made my analysis easier.

The data set has only 27 unique dates (April, May & June) therefore it does not provide a good representative sample to predict whether a patient will not show up outside those 27 days.

5. Output / Analysis of the result:

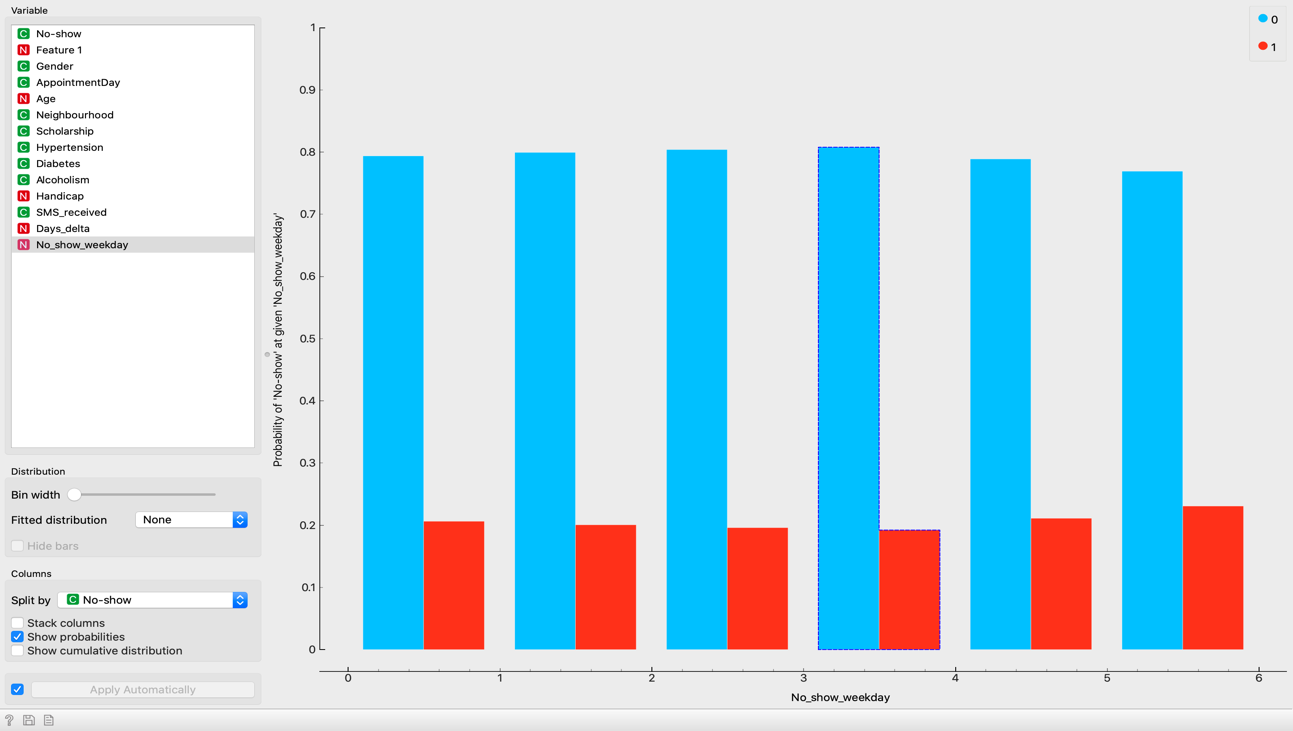
The analysis of the no\_shows classification problem is to predict the important features or reasons for not showing up for the doctor’s appointment. According to the different algorithm result Random Forest algorithm was the best classifier with classification accuracy of 80.4%. The methods used for feature selection can be an important criterion in selecting the informative features. Here the rank & scoring method features days\_delta (which was generated or engineered from the data available), age, sms\_received as the top four features which can be the reason for not showing up.

* The feature days\_delta which is the difference between the no of days passed between making the initial appointment and actual scheduled appointment can be an intuitive feature why patients don’t show up. If the days\_delta is more the probability of showing up for the doctor’s appointment can be less.

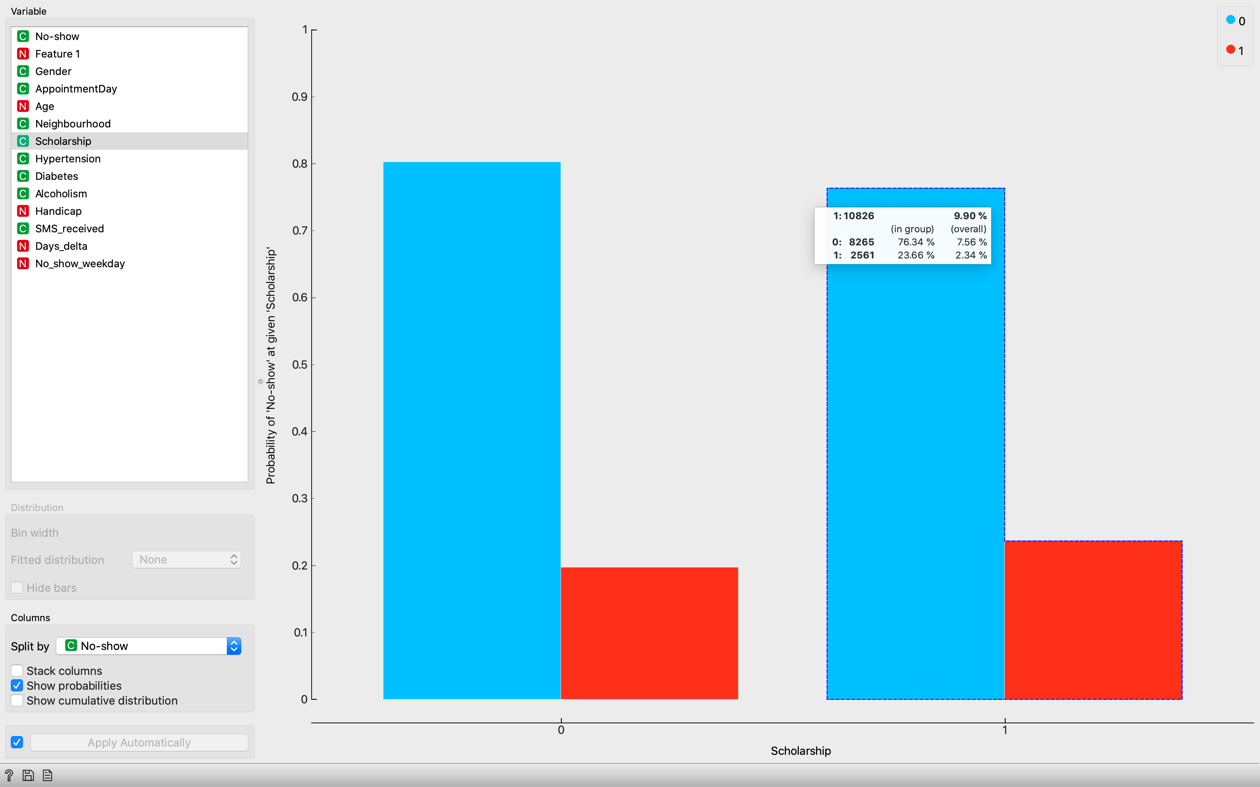


* Sms\_received can be helpful in understanding that if people have received SMS on or the day before the appointment there is a higher probability of showing up at the appointment.
* Younger individuals tend to show higher attendance rate as depicted through box plot before.
* The highest proportion of no\_shows coincided with when the appointment day is Friday/Saturday. There was no data for Sunday, for which the explanation can be that on

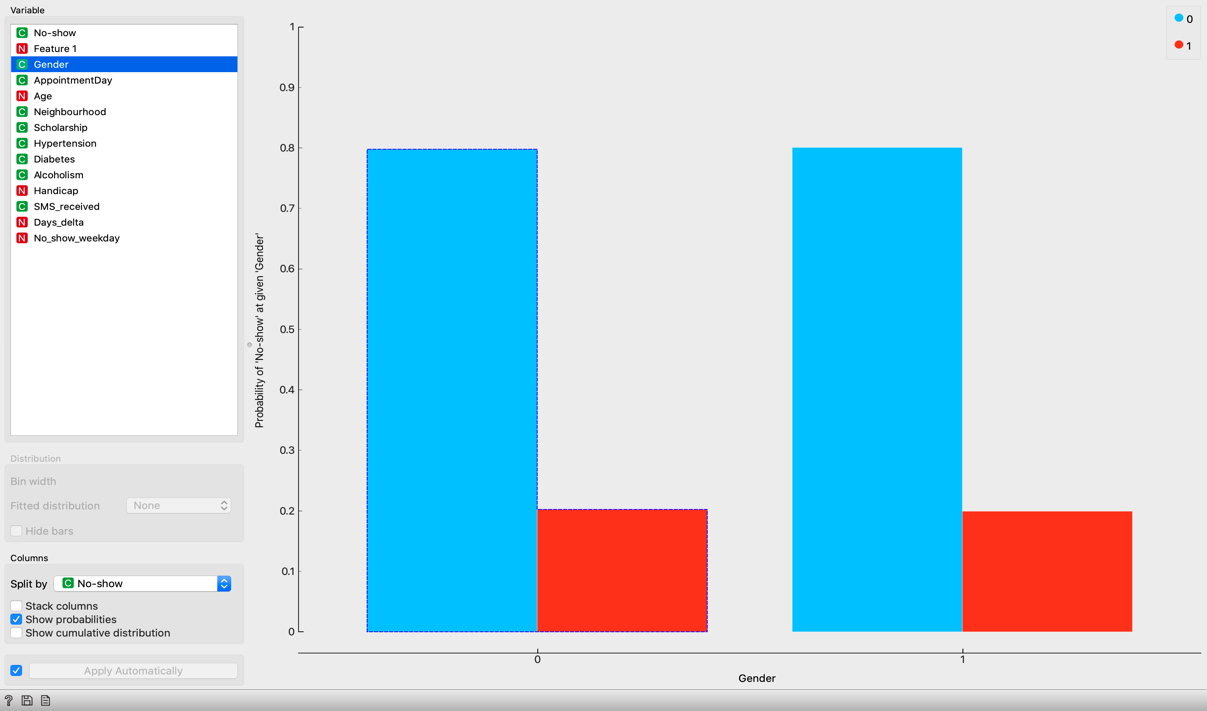
Sunday, the clinic remains closed.

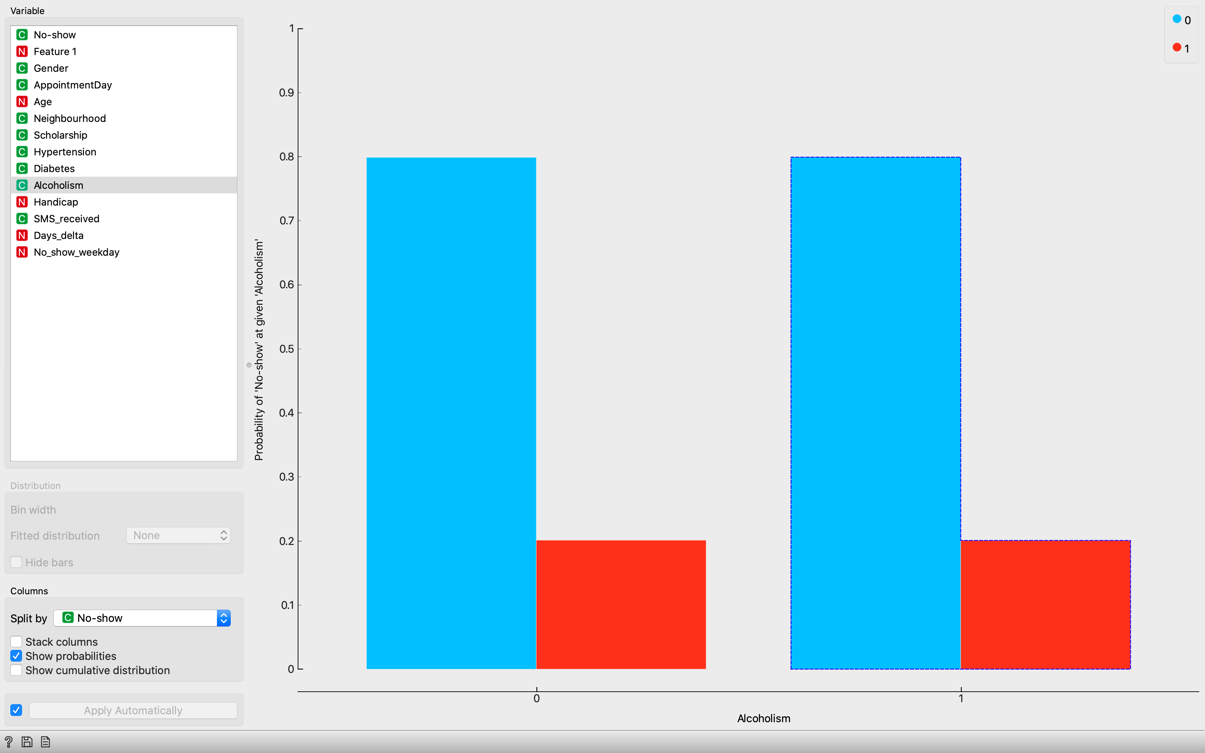


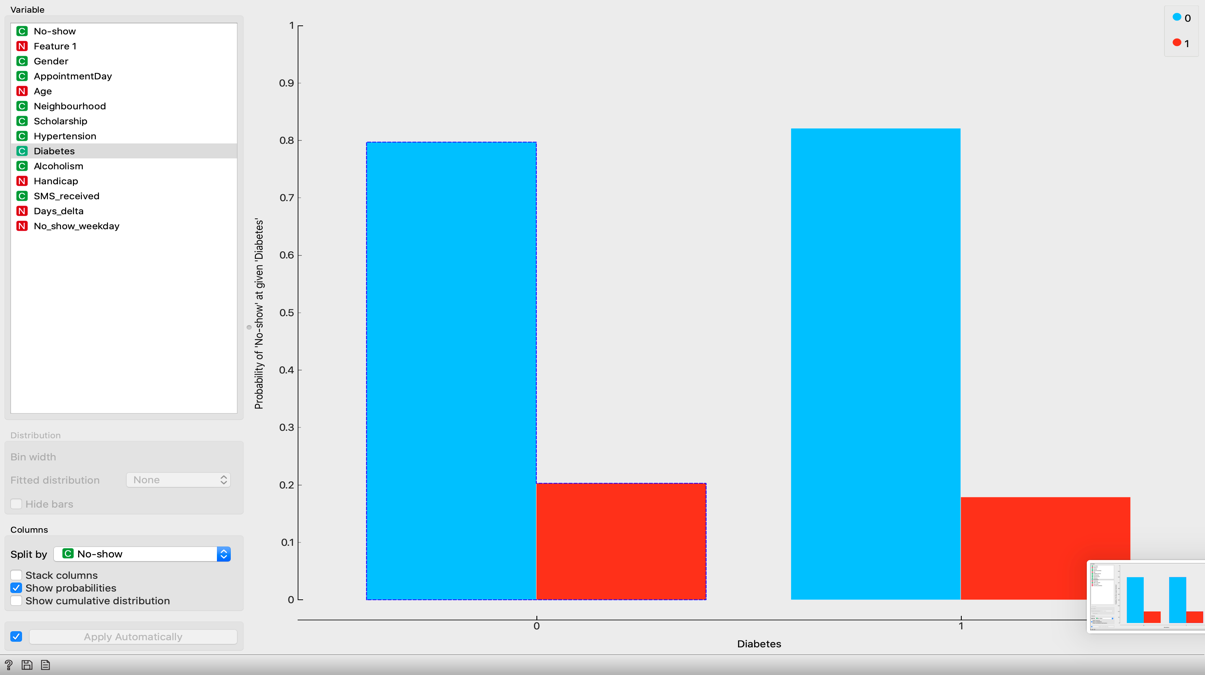
* People who have received welfare assistance (scholarship) have slightly higher no\_shows.



* Gender, alcoholism and diabetes did not have any effect on no shows as shown by the chart below:







What Went Well:

* The data visualizations with the distribution widget was pretty helpful in determining the factors and how it is affecting the no\_shows.
* In Orange, data analysis can be done by visual programming. It proposes the most frequent combinations.
* Orange was very easy, colorful and fun to use. The installation was quick and user-friendly widgets.
* The ranking of the features and selecting the informative variable for this data set was easy and helpful.
* The python script widget was very helpful in initial data cleaning process.
* Data Sampling / Cross-validation was quite easy.

What Did Not go Well:

* Loading data into Orange was very slow. It took plenty of time to load the data.
* Selecting the target variable was confusing since it didn’t have the widget for the same but needed to be set once you are loading the data.
* Ranking the algorithms w.r.t the prediction accuracy was very time consuming.
* The flow of the process starting from loading the data, selecting attributes, doing some data preparation and applying the data mining algorithm was not very smooth and streamlined.

What Would you do Differently Next Time:

* If I choose Orange as the data mining tool I will definitely choose a dataset which is smaller with lesser rows/examples.
* I would like to play more with the outliers and imputing missing values because the dataset I worked on didn’t have any missing values.
* I would like to focus more on data visualizations with different other widgets available like heatmap, linear projection and Venn-diagram.

Reference

Kaggle Data Set. Retrieved from URL: <https://www.kaggle.com/joniarroba/noshowappointments>