The Project was implemented using the DRISP-DM Methodology and we as a team and individually were able to implement it.

Business and Data Understanding:

- We have searched multiple datasets where we can get some interesting problem-solving dataset which can create some impact on society or peoples life. We got the dataset which was useful and meeting our requirements
- After a lot of research for the selection of dataset, we came across the dataset of US census dataset which gives the information about the citizen's information like age, work class, education, marital status, occupation, sex, ethnicity, annual income, etc having records of about 50,000 individuals.
- We have an annual salary column that has the value in which the annual salary of a person less than 50,000 USD or greater than 50,000 USD.
- By using age, work class, education, occupation, sex, capital gains, and loss, hours/week we can predict the column income.
- In this project, we are considering the age column as the categorical which in the general case it's considered as a continuous variable.
- This Type of census data can be useful for companies or governments to take the decision on investment or to implement some social scheme for the people in that particular area.
- The company is of the FMCG sector, then they can easily work out whether to open the store or launch the product in that particular area depending on the salary of the people.
- If the restaurant wants to open a new store in that particular area they can easily find out whether that restaurant will get the response or not in that area.

Data Preparation:

Feature Selection:

- We have selected features depending on the results from the correlation matrix. We have taken the features that are correlated and which can contribute to the well working of our mode without compromising the assumptions. After the analysis of the correlation matrix, we decided some features and omitted the remaining features.
- Features are age, work class, education, marital status, occupation, sex, ethnicity, annual income, and dropping the insignificant variables such as Capital Gain, Capital Loss, Education Number, Native Country.
- We as the Team decided to implement the same project in the Rapid Miner and Python using the K fold cross-validation. From this most of the implantation is done by me and my teammates helped me as it was teamwork. Most of the selection of dataset and data gathering is done by my teammates in which I was involved too.
- In the data preparation stage, we have dropped the unwanted columns and we found that we have 22 NA values, so we decided to drop that 22 NA values instead of using the mean, median, mode.

Modeling:

Auto Model:

- Auto Model is an important feature of RapidMiner. Auto Model helps us to find three large classes of problems: Prediction, Clustering, Outliers.
- I have run the auto model on the given dataset after cleaning the dataset. After Running the Auto model I got different results of the different model which is the best suggestion we can get which model to use for the implementation of our dataset using the features. Naïve Bayes has an accuracy of 81.2%. Decision Tree has an accuracy of 81.8%. Gradient Boosted Trees has an accuracy of 81.1%.
- And I ran various Machine Learning Models which gave the different results which were not significant as the above models. So we decided to deploy those above models in our project. So I implemented the Decision tree and Naïve Bayes models in our project.

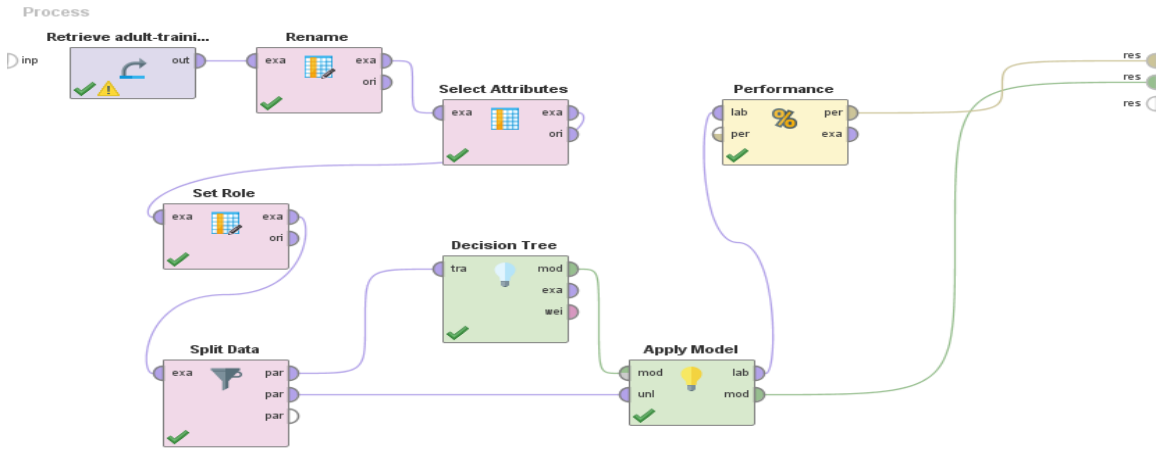

Fig.1. RapidMiner Decision Tree Implementation

Implementation:

- While implementing the model I have used rename function from which I had renamed all the column or attribute names of my dataset to meaningful names that are friendly to use.
- From the attributes, I have selected important features that are useful for us.
- In the set Role I have set the output that I want for my model. In my case, it's the salary column.
- And I have used the Split data function to split the data into Train and Test. In the training dataset, we have taken 70% of the data while for the test we have taken 30% of the data.
- I have used different models in our projects. But the main aim was to implement the Decision Tree. In RapidMiner to deploy the models is very easy. We have to keep the whole procedure as it is and just to change the model name.
- So I have changed the model name from Random Forest to the Gradient Boosted Tree and find the accuracy of the model
- And the same procedure was again performed and used the Model Decision Tree. In Decision Tree, we have got good accuracy but the random forest was the best to select as the model.

Evaluation:

We have got the accuracy as the evaluation criteria. We have recorded the accuracy of the models and we got the same accuracy as it was suggested in the auto model.

K fold Cross-Validation:

- I have implemented the k fold cross-validation in the python for the same dataset in python. I have used k=10 and K=5 to perform the same operation. In python, I got an accuracy of about 83.6%. While implementing the given project in python I have used features such as stratification so that I can get the proper selection of the dataset. So using the stratification Feature our accuracy has increase by 1 percent, which was a significant change as only 1 parameter was changed. So we as the team concluded that the K fold cross-validation and stratifications work constructively in our dataset.

Deployment: We can deploy this model using various applications in which the user can put their dataset and our model will function accordingly and will find the results appropriate for the business users.

Conclusion: After Implementing the project in RapidMiner and Python we have cross-checked the results which were in the range of 5%. So the Decision Tree as the model was the best choice available in our dataset to achieve the required results.