

**School of InfoComm Technology**

**Machine Learning**

Diploma in Data Science (DS)

Diploma in Information Technology (IT)

October 2022 Semester

**INDIVIDUAL ASSIGNMENT 2**

(40% of Machine Learning Module)

**Deadline for Submission:**

**Presentation: 29th Jan 2023 (Sunday), 2359 Hours**

**Report: 11th Feb 2023 (Saturday), 2359 Hours**

|  |  |  |
| --- | --- | --- |
| Student Name | : | Brian Low De Quan |
| Student Number | : | S10228107D |

**Penalty for late submission:**

10% of the marks will be deducted every day after the deadline.

**NO** submission will be accepted after 18th Feb 2023, 23:59.

Contents

[1. Introduction 3](#_Toc126765178)

[2. HR Analytics 3](#_Toc126765179)

[2.1. Problem Understanding 3](#_Toc126765180)

[2.2. Build Model 3](#_Toc126765181)

[2.2.1. Load & Sample Data 3](#_Toc126765182)

[2.2.2. Logistic Regression 4](#_Toc126765183)

[2.2.3. Decision Tree 4](#_Toc126765184)

[2.2.4. Artificial Neural Network 5](#_Toc126765185)

[2.2.5. Random Forest 5](#_Toc126765186)

[2.2.6. Support Vector Machine 6](#_Toc126765187)

[2.3. Evaluate & Improve Model 6](#_Toc126765188)

[2.3.1. Logistic Regression 6](#_Toc126765189)

[2.3.2. Decision Tree 6](#_Toc126765190)

[2.3.3. Artificial Neural Network 7](#_Toc126765191)

[2.3.4. Random Forest 7](#_Toc126765192)

[2.3.5. Support Vector Machine 7](#_Toc126765193)

[2.4. Summary 7](#_Toc126765194)

[3. Airbnb 7](#_Toc126765195)

[3.1. Problem Understanding 7](#_Toc126765196)

[3.2. Build Model 7](#_Toc126765197)

[3.2.1. Load & Sample Data 7](#_Toc126765198)

[3.2.2. Logistic Regression 8](#_Toc126765199)

[3.2.3. Decision Tree 8](#_Toc126765200)

[3.2.4. Artificial Neural Network 9](#_Toc126765201)

[3.2.5. Random Forest 9](#_Toc126765202)

[3.2.6. Support Vector Machine 10](#_Toc126765203)

[3.3. Evaluate & Improve Model 10](#_Toc126765204)

[3.3.1. Logistic Regression 10](#_Toc126765205)

[3.3.2. Decision Tree 10](#_Toc126765206)

[3.3.3. Artificial Neural Network 10](#_Toc126765207)

[3.3.4. Random Forest 10](#_Toc126765208)

[3.3.5. Support Vector Machine 10](#_Toc126765209)

[3.4. Summary 10](#_Toc126765210)

[4. Conclusion 10](#_Toc126765211)

[4.1. HR Analytics 10](#_Toc126765212)

[4.2. Airbnb 10](#_Toc126765213)

[5. Reflection 11](#_Toc126765214)

[5.1. Further Improvement 11](#_Toc126765215)

[5.2. Reflect 11](#_Toc126765216)

# Introduction

In this assignment, Python will be used to solve classification and regression issues using machine learning models. to create machine learning models to address the issues with Airbnb and HR analytics. Next, train the models by modifying various hyperparameters. The performance of the model would then be assessed, and findings would be reported. To accurately select the best model for each of the problems, compare the various models.

I must first move the two datasets from the earlier assignment to the assignment 2 folder so I can load the data later on for modeling before I can start putting models to work on the problems.

# HR Analytics

# Problem Understanding

Human resources departments are operating in a new way thanks to HR analytics, which increases productivity and produces better outcomes overall. Analytics have been used by human resources for a long time. Due to the manual nature of data collection, processing, and analysis, as well as the dynamics of human resources and HR KPIs, HR has been constrained by this method. To help them predict the likelihood that employees will be promoted, which turns out to be a classification problem, they would like to use a machine learning model. Features containing employee personal information, education background, past performance, places they currently work at and the department they belong in will bring importance towards employee promotion prediction for this classification problem Which the dataset has been cleanse and encoded from assignment one as shown below is the description of the cleanse dataset.

The cleansed dataset only has numerical values and potentially meaningful features. To understand the meaning of each feature /column, you can refer to the below short descriptions (equivalent to a data dictionary).

* **department**: department that the employee working in (e.g. 1~9)
* **education** (study level): 1 = Bachelors, 2 = Master's & above Education, 3 = Below Secondary
* **gender**: 0 = Female, 1 = Male
* **recruitment\_channel** (channel that employee get recruited from): 1 = sourcing, 2 = referred, 3 = other
* **no\_of\_trainings**: # of trainings
* **age**: normalised to 0 ~ 1, i.e. the percentile of the age
* **previous\_year\_rating** the rating given to employee performance last year
* **length\_of\_service**: (employee length of service): 0 = (0 ~ 3), 1 = (3 ~ 5), 2 = (5 ~ 7), 3 = (above 7)
* **KPIs\_met >80%**: 0 = lower than 80%, 1= higher than 80%
* **awards\_won?**: 0 = no awards won, 1 = won an award
* **avg\_training\_score**: normalised to 0 ~ 1, i.e. the percentile of the average training score
* **is\_promoted**: The target variable, 0 = not promoted, 1 = promoted
* **region\_zone**: regional zones of the employee working at (e.g. 1~26)

# Build Model

# Load & Sample Data

Table

Description automatically generated

Before building models for evaluation, we must first load the dataset from the previous assignment into the new Jupyter notebook in assignment 2 folder. Followed by copying the data frame into a new copy for usage and having the old one as backup in case of wrong implementation. Also, we must define the inputs and output of the dataset by giving it a X and Y variable respectively, having the y being the target value “is\_promoted?” and the remaining values will be the input that will be the X. Finally with the split of 70% training data and 30% testing data with the random state of dataset being 1.

# Logistic Regression

Graphical user interface, text, application

Description automatically generated

First, I attempted logistic regression which is commonly used in Predictive analytics and classification, also referred to as a logit model. Based on a given dataset of independent variables, logistic regression calculates the likelihood that an event will occur, such as promoted or not promoted. In this method, the response or dependent variable is dichotomous, meaning that there are only two possible outcomes (e.g., 0 or 1). Hence it is known as binary logistic regression. As we can see, the accuracy of the training and testing data outcome turn out to be very close to each other having training of 0.729 and testing of 0.722.

# Decision Tree

Text

Description automatically generated

Timeline

Description automatically generated with medium confidence

I then attempted using a classification decision tree, which involves splitting the data set into smaller sets depending on the descriptive features until we get a set that is sufficiently tiny and contains only data points that belong to a single label. The leaf[child] nodes reflect the results, while each feature of the data set is represented by a root[parent] node. For instance, this straightforward decision tree can be used to forecast whether it will be promoted. This is useful when trying to predict the class that a set of features should be placed into. Hence, we were able to fit our model and get the scoring of training and testing accuracy where training is 0.693 and testing is 0.682.

# Artificial Neural Network

Graphical user interface, text, application

Description automatically generated

Next, Classification Artificial Neural Network will be built to classify an observation as belonging to some discrete class as a function of the inputs. The input features (independent variables) can be categorical or numeric types; however, we require a categorical feature as the dependent variable. For the parameter we have applied on the default ANN model, we will be using a max iteration of 2000 for the model to fully converge without any issue and a random state of 2 to keep it standardized as the other model random state. With this, I was able to achieved training accuracy of 0.825 and testing accuracy of 0.801.

# Random Forest

Text, letter

Description automatically generated

For the 4th classification model, I will be building Random Forest Classification model. Which is an advance modeling which consists of a large number of individuals decision trees that operate as an ensemble. Where each individual decision tree in the random forest spits out a class prediction and the class with the most votes becomes our model’s prediction. It required to have low correlation between each model for the random forest classifier to be working together. In the default model, I have applied number estimator of 100 to declare a total of 100 decision tree activated within random forest with the random state of 2. With this, we achieved training accuracy of 0.999 and testing accuracy of 0.793.

# Support Vector Machine

Lastly, we will be applying Support Vector Machine, and the objective would be finding hyperplane in a N-dimensional space that distinctly classifies the data points.

Graphical user interface, text, application

Description automatically generated

Previously, I have made a mistake where I implemented 2 svc scoring based off train dataset while I am supposed to implement test on the second accuracy scoring hence, I got similar output as shown above.

Graphical user interface, text, application

Description automatically generated

After implementing the correct input as shown above, I was able to get training accuracy of 0.708 while testing accuracy of 0.696, this would mean that there is a slight variance between the training and testing accuracy.

# Evaluate & Improve Model

# Logistic Regression

Text, letter

Description automatically generated

The first improvement model I have applied towards logistic regression is k-fold validation. By applying k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. Which is shown in the array above with 5 different accuracy and to get a distinct accuracy from k-fold, one would need to combine the 5 outcome and divide it to get the average accuracy of the 5 outcome for it to be cross validation accuracy. Having a training accuracy of 0.727 and testing accuracy of 0.725.

Graphical user interface, text, application, email

Description automatically generated

From the above model summary, we can see that all features have close to ZERO **p-value** (i.e., the column P>|z|). It means we have very high confidence on the coefficients of these features, or say these features are the strongest indicators to predict is\_promoted.

However, features such as education and length of service has a small amount of p-value, but it wouldn’t make a huge impact on the accuracy of the models as the p-value is not in the range that is close the 1, hence there isn’t any improvement needed to be made towards the logistic regression model.

# Decision Tree

Text

Description automatically generated

For improvement of decision tree, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different accuracy and to get a distinct accuracy from k-fold, one would need to combine the 5 outcome and divide it to get the average accuracy of the 5 outcomes for it to be cross validation accuracy. Having a training accuracy of 0.690 and testing accuracy of 0.690.

Graphical user interface, application, Word

Description automatically generated

I have also implemented Ada Boost which is similar to random forest modeling where it is an ensemble modelling technique used to applied on decision tree classification only. This algorithm improves the prediction power by converting several weak learners. Which result a boosted training accuracy of 0.807 and testing accuracy of 0.802, which is a big improvement from the base decision tree model.

Graphical user interface, text, application

Description automatically generated

A picture containing chart

Description automatically generated

Next, I have improved my decision tree using grid search to find the best parameter from the different combinations of parameter I have implemented in the grid search to and build a new model based off the grid search parameters given to me. As we can see from the above, the decision tree has increased depth and results in an increase of training accuracy and testing accuracy as compared to the base accuracy. However, by having more layers towards the tree will result in overfitting of model due to noisy data. Hence, having more depth will not only result in better accuracy but will affect the data integrity due to highly fragmented due to insufficient of notes/features within the dataset.

Graphical user interface, text, application, chat or text message

Description automatically generated 

Lastly, I have tried removing insignificant features by applying feature importance and removed the highlighted features as it provide0 signficant towards the target variable, I was able to get the same accuracy as the base model.

# Artificial Neural Network

Text

Description automatically generated

For improvement of ANN model, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different accuracy and to get a distinct accuracy from k-fold, one would need to combine the 5 outcome and divide it to get the average accuracy of the 5 outcomes for it to be cross validation accuracy. Having a training accuracy of 0.812 and testing accuracy of 0.786.

Text

Description automatically generated



Next, I have improved my ANN model using grid search to find the best parameter from the different combinations of parameter I have implemented in the grid search to and build a new model based off the grid search parameters given to me. As we can see from the above, the parameter I used have up to a total of 10 hidden layers using a total of 2000 iteration to converge my model. With this, we can achieve a best score of 0.80 from the grid search best combination. Finally, when fitting the values with the new parameters, I was able to get accuracy of 0.809 for training and accuracy of 0.805 for testing.

# Random Forest

Text

Description automatically generated with medium confidence

For improvement of Random Forest model, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different accuracy and to get a distinct accuracy from k-fold, one would need to combine the 5 outcome and divide it to get the average accuracy of the 5 outcomes for it to be cross validation accuracy. Having a training accuracy of 0.999 and testing accuracy of 0.795. This may not be a good model as the variance is huge meaning overfitting has occur and might not be able to predict with high accuracy using new dataset.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Next, I have improved my Random Forest model using grid search to find the best parameter from the different combinations of parameter I have implemented in the grid search to and build a new model based off the grid search parameters given to me. As we can see from the above, the parameter I used 8 depths for our random forest using a total of 20 decision tree within my random forest model. With this, we can achieve a best score of 0.80 from the grid search best combination. Finally, when fitting the values with the new parameters, I was able to get accuracy of 0.795 for training and accuracy of 0.787 for testing. However, the same issue might happen to random forest as by having more depth towards the tree will result in overfitting of model due to noisy data.

Graphical user interface, text, application

Description automatically generated

Lastly, I have tried removing insignificant features by applying feature importance and removed the highlighted features as it provide least signficant towards the target variable. However, I end up getting slightly lesser accuracy for my testing due to lost of data and drop from the base model accuracy of 0.79 training to 0.78 testing..

# Support Vector Machine

Graphical user interface, text, application

Description automatically generated

For improvement of SVM model, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different accuracy and to get a distinct accuracy from k-fold, one would need to combine the 5 outcome and divide it to get the average accuracy of the 5 outcomes for it to be cross validation accuracy. Having a training accuracy of 0.716 and testing accuracy of 0.795.

Graphical user interface, text, application, email

Description automatically generated

Next, I have improved my SVM model using grid search to find the best parameter from the different combinations of parameter I have implemented in the grid search to and build a new model based off the grid search parameters given to me. As we can see from the above, the parameter I have applied only has a few options such as gamma and C that is for controlling of error. After applying gamma of 0.01 and C of 10, I was able to obtain training accuracy of 0.77 and testing of 0.74.

# Summary

Table

Description automatically generated

As mention during the presentation, I will be choosing Decision Tree with the improvement of Ada Boost as my best model for HR Analytics classification problem as it has the lowest variance out of all the different combinations of models I have did and the lowest bias due to it not having a wide range of gap between training and testing data. Which means that the model will be able to predict the outcome of promotion almost 80% of the time.

# Airbnb

# Problem Understanding

Since 2008, Airbnb has operated as an online marketplace that connects property owners looking to rent out their spaces with travelers looking for affordable lodging while visiting cities or towns. Due to the large amount of data contain in Airbnb dataset, travelers would need a machine learning model that could aid them in predicting a regression problem of the rental price of listed properties to ensure they would get the best condition that suited their needs when choosing which listed properties to stay at. As a result, our model will include elements like the neighborhoods that represent the desired location, the precise location of properties, feedback provided by previous guests while they were staying at particular properties, the availability of particular listed properties, as well as the daily rental price, which will serve as the target variable for this regression problem. Which the dataset has been cleanse and encoded from assignment one as shown below is the description of the cleanse dataset.

The cleansed dataset only has numerical values and potentially meaningful features. To understand the meaning of each feature /column, you can refer to the below short descriptions (equivalent to a data dictionary).

* **neighbourhood\_group**: ("North Region": 1, "Central Region": 2, "East Region": 3, "West Region": 4, "North-East Region": 5)
* **latitude**: normalised to 0 ~ 1, i.e. the percentile of the latitude
* **longitude**: normalised to 0 ~ 1, i.e. the percentile of the longitude
* **room\_type**: ("Private room": 1, "Entire home/apt": 2, "Shared room": 3)
* **price**: Target for regression problem (the rental price of listings properties)
* **minimum\_nights**: 0 = (0 ~ 1), 1 = (1 ~ 3), 2 = (3 ~ 10), 3 = (above 10)
* **number\_of\_reviews** : 0 = (0), 1 = (0 ~ 2), 2 = (2 ~ 10), 3 = (above 10)
* **reviews\_per\_month**: 0 = (0 ~ 0.33), 1 = (0.33 ~ 0.55), 2 = (0.55 ~ 0.85), 3 = (above 0.85)
* **calculated\_host\_listings\_count**: 0 = (0 ~ 2), 1 = (2 ~ 9), 2 = (9 ~ 48), 3 = (above 48)
* **availability\_365**: 0 = not open all time throughout the year, 1 = open 365days a year
* **season**: Jan-March = 1, April-June = 2, July-September = 3, October-December = 4
* **weekend?**: whether the review is on the weekend (0 = weekdays, 1 = weekends)

# Build Model

# Load & Sample Data

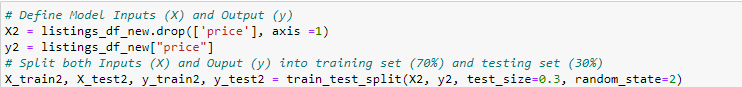
Table

Description automatically generated

A picture containing chart

Description automatically generated Chart, box and whisker chart

Description automatically generated



As for Airbnb regression problem, I must first load the dataset from the previous assignment into the new Jupyter notebook in assignment 2 folder. Followed by copying the data frame into a new copy for usage and having the old one as backup in case of wrong implementation.

However, I have realized that the outlier for price has value of up to 10000 when the average of Airbnb price is around 0 to 400, hence I will be applying removal of outlier as it will help us scale down the dataset to the average value and we will not have a high rate of mse/rmse else it will affect the prediction model due to the extreme high value.

Also, I must define the inputs and output of the dataset by giving it a X and Y variable respectively, having the y2 being the target value “Price” which is a continuous variable, and I will be predicting the prices using the remaining values which will be the input that will be the X2. Finally with the split of 70% training data and 30% testing data with the random state of dataset being 2.

# Linear Regression

Text

Description automatically generated

Unlike logistic regression, which is commonly used in Predictive analytics and classification, I will be applying linear regression towards regression problem Linear Regression is a supervised learning technique that involves learning the relationship between the features and the target. The target values are continuous, which means that the values can take any values between an interval. As we can see, the R2 of the training and testing data outcome turn out to be very close to each other having training of 0.108 and testing of 0.096. We can see the outcome of R2 is relatively low as compared to the average r2 value, which is normally above 0.50. This issue could be due to many factors, such as the dataset I have applied, or the techniques used in assignment 1 to cleanse my dataset.

# Decision Tree

Text

Description automatically generated

A picture containing diagram

Description automatically generated

As for decision tree, I have applied the regression tree which is used to response on variable that is continuous. In a regression tree, a regression model is fit to the target variable using each of the independent variables. The data is then split into several points for each independent variable. Through this, I was able to achieve rmse value of 65 for training and 67 for testing data. While the r2 value is around 0.40 for training and 0.37 for testing data.

# Artificial Neural Network

Text

Description automatically generated

Next, Regression Artificial Neural Network will be built for learning of complex non-linear relationships that linear regression couldn’t achieve. To learn the complex non-linear relationship between the features and target, we are in need of other techniques. One of those techniques is to use Artificial Neural Networks. Artificial Neural Networks could learn the complex relationship between the features and target due to the presence of activation function in each layer. Using the configuration of max iteration of 2000 to converge the model successfully and a random state of 2. I was able to achieve training rmse of 62 and testing rmse of 65, with training r2 of 0.45 and testing r2 of 0.41.

# Random Forest

Text

Description automatically generated

I will be building Random Forest Regression model. Which is an advance modeling which consists of many individuals’ decision trees that operate as an ensemble. Where each individual decision tree in the random forest spits out a class prediction and the class with the most votes becomes our model’s prediction. It required to have low correlation between each model for the random forest classifier to be working together. In the default model, I have applied number estimator of 20 to declare a total of 20 decision tree activated within random forest with the random state of 1. I was able to achieve rmse of 65 for training and rmse of 67 for testing with r2 of 0.40 for training and r2 of 0.37 with testing.

# Support Vector Machine

Text

Description automatically generated

Lastly, I will be applying SVM regression as SVM regression is considered a nonparametric technique because it relies on kernel functions. The set of training data includes predictor variables and observed response values. Using the parameters shown above, I was able to get rmse of 81 for training and rmse of 84 for testing with r2 of 0.06 training and r2 of 0.03 of testing. This the lowest r2 value out of all the models and svm is probably not a great model when attempting regression problem.

# Evaluate & Improve Model

# Linear Regression

Text

Description automatically generated

For improvement of Linear Regression model, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different r2 values and to get a distinct r2 value from k-fold, one would need to combine the 5 outcome and divide it to get the average r2 value of the 5 outcomes for it to be cross validation r2 value. Having a training r2 of 0.105 and testing r2 of 0.043.

Graphical user interface, table

Description automatically generated A screenshot of a computer

Description automatically generated with medium confidence

From the above model summary, we can see that all features have close to ZERO **p-value** (i.e., the column P>|z|). It means we have very high confidence on the coefficients of these features, or say these features are the strongest indicators to predict price.

However, features such as availability\_365 and weekend? has a significant amount of p-value, hence I will be removing them before building another linear regression without these 2 features, my training r2 has drop to 0.106 while testing r2 has risen to 0.097.

# Decision Tree

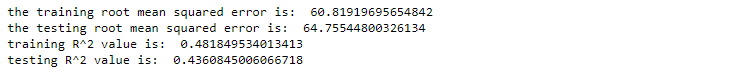
Text

Description automatically generated

For improvement of decision tree model, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different r2 values and to get a distinct r2 value from k-fold, one would need to combine the 5 outcome and divide it to get the average r2 value of the 5 outcomes for it to be cross validation r2 value. Having a training r2 of 0.395 and testing r2 of 0.366.

Graphical user interface, text, application

Description automatically generated



Next, I have improved my Decision Tree model using grid search to find the best parameter from the different combinations of parameter I have implemented in the grid search to and build a new model based off the grid search parameters given to me. As we can see from the above, after applying the best parameter, I was able to obtain training r2 of 0.48 and testing of 0.43.

Graphical user interface, text, application

Description automatically generated Text

Description automatically generated

Lastly, I have tried removing insignificant features by applying feature importance and removed the highlighted features as it provide least signficant towards the target variable. However, I end up getting slightly lesser r2 value for my testing due to lost of data and drop from the base model r2 of 0.47 training to 0.42 testing.

# Artificial Neural Network

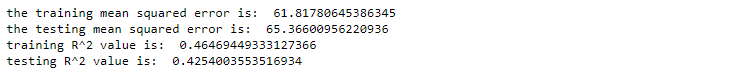
Text

Description automatically generated

For improvement of ANN model, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different r2 values and to get a distinct r2 value from k-fold, one would need to combine the 5 outcome and divide it to get the average r2 value of the 5 outcomes for it to be cross validation r2 value. Having a training r2 of 0.462 and testing r2 of 0.383.

Text, application

Description automatically generated



Next, I have improved my ANN model using grid search to find the best parameter from the different combinations of parameter I have implemented in the grid search to and build a new model based off the grid search parameters given to me. As we can see from the above, after applying the best parameter, I was able to obtain training r2 of 0.46 and testing of 0.42.

# Random Forest

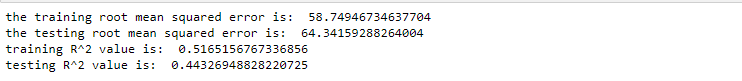
Text

Description automatically generated

For improvement of random forest model, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different r2 values and to get a distinct r2 value from k-fold, one would need to combine the 5 outcome and divide it to get the average r2 value of the 5 outcomes for it to be cross validation r2 value. Having a training r2 of 0.396 and testing r2 of 0.366.

Graphical user interface, text, application

Description automatically generated



Next, I have improved my random forest model using grid search to find the best parameter from the different combinations of parameter I have implemented in the grid search to and build a new model based off the grid search parameters given to me. As we can see from the above, after applying the best parameter, I was able to obtain training r2 of 0.51 and testing of 0.44.

Graphical user interface, text, application

Description automatically generated Text

Description automatically generated

Lastly, I have tried removing insignificant features by applying feature importance and removed the highlighted features as it provide least signficant towards the target variable. To my surprise, I end up getting higher r2 value of 0.41 training and 0.38 testing from removal of feature that bring noises towards the data.

# Support Vector Machine

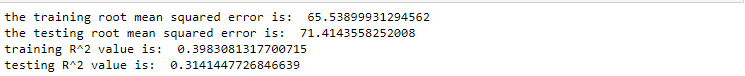
Text

Description automatically generated

For improvement of SVM model, I have also applied k fold validation, they will split the dataset into 5 separate smaller portions to validate and fit to get the result of all 5 scoring. which is shown in the array above with 5 different r2 values and to get a distinct r2 value from k-fold, one would need to combine the 5 outcome and divide it to get the average r2 value of the 5 outcomes for it to be cross validation r2 value. Having a training r2 of 0.156 and testing r2 of 0.103.

Graphical user interface, text, application

Description automatically generated



Next, I have improved my SVM model using grid search to find the best parameter from the different combinations of parameter I have implemented in the grid search to and build a new model based off the grid search parameters given to me. As we can see from the above, after applying the best parameter, I was able to obtain training r2 of 0.39 and testing of 0.31.

# Summary

Table

Description automatically generated

Out of all the combination of models I have tried, Decision Tree with the aids of Grid search to find my best hyper parameter which turns out to be training r2 value of 0.481 and testing r2 value of 0.436. I chose this as it has high variance and low bias within the train and test.

# Conclusion

# HR Analytics

To conclude my journey when implementing models to find the best fitted model for HR analytics classification problem, I have implemented decision tree with the implementation of Ada boost to help decision tree perform better. With training accuracy of 0.807 and testing accuracy of 0.802 which achieve the machine learning model standard of having high variance and low bias due to the difference between training and testing accuracy not being wide.

# Airbnb

To conclude my journey when implementing models to find the best fitted model for Airbnb regression problem, I have implemented decision tree with the implementation of Grid Search to find the best parameter for my decision tree model to perform better than the base version. With training r2 of 0.481 and testing r2 of 0.436 which achieve the machine learning model standard of having high variance and low bias due to the difference between training and testing accuracy not being wide.

# Reflection

# Further Improvement

# HR Analytics

Graphical user interface, text, application

Description automatically generated

As mentioned during the presentation, due to time constraint I couldn’t build xgboost ahead and decided to build it in further improvement as shown above. XGBoost is a scalable, distributed gradient-boosted decision tree (GBDT) machine learning library. It provides parallel tree boosting and is the leading machine learning library for regression, classification, and ranking problems. I was able to achieve training accuracy of 0.82 and testing accuracy of 0.81.

# Airbnb

Graphical user interface, text, application, email

Description automatically generated

For Airbnb, I have also attempted to use xgboost regressor on my dataset. However, with the twist of using untouched numerical features, I was able to reach accuracy of 0.52 for training and 0.44 for testing.

# Reflect

Could have test with much more variety of model such as XGBoost as having a poor time management cost me to not be able to test every single model option given. Which was implemented above during at the further improvement section. Another thing I would keep in mind would be the method I have implemented during cleansing and removal of outlier on my dataset as that is what affecting my R2 values of Airbnb to be lower than average. Overall, it was the learning that’s matter as it helps me understand more of what machine learning engineer out there do daily.

# Discussion Board

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated