Data Mining and Predictive Modelling (082025-PRE)

GROUP ASSIGNMENT

Name TP Number	OOI DUN TZI TP071308
Name TP Number	**YAP BOON SIONG** **TP070171**
Name TP Number	**CHEAH JUN HENG** **TP071767**
Intake Code	**APU2F2502CS(DA)**
Lecturer Name	**Dr. Preethi Subramanian**

Contents

1.0 Introduction

Nowadays, modern banking relies heavily on predictive modelling, particularly in marketing efforts were gathering or comprehending client feedback can significantly enhance efficiency and decision-making. Furthermore, by evaluating client data, banks may make more informed predictions about which people are most likely to respond to various marketing initiatives, enabling for more targeted and cost-effective plans to emerge.

Furthermore, this report will be showing a predictive classification model that is developed by using SAS Model Studio to analyse the bank marketing dataset, and the target variable in this dataset is representing whether the customer subscribes to the offered services or not. However, the modelling process involves basic data preparation, data partitioning, and the creation of modelling pipeline, followed by the training and tuning of several classification algorithms. Nevertheless, the models that going to explore it Decision Tree, Bayesian Network, Neural Network, Logistic Regression, Forest, and Gradient Boosting. At the end of the project, it will outcome with the 7 models comparison to determine which one provides the highest accuracy in predicting customer response.

2.0 Business Case / Domain Knowledge (total word count 500)

2.1 Problem Statement / Business Goals

The customer retention and marketing campaigns targeted to specific customers are critical success factors in the banking industry today to determine financial viability and competitive standing. However, a key threat facing financial institutions is with regard to identification of those clientele who have a high likelihood of responding to a positive campaign on marketing campaigns. The data used in the current study was obtained with the help of Kaggle and was rawled by Gavrysh,2021. They are a comprehensive list of exhaustive reports generated through direct marketing programs pursued by a Portuguese bank. Every record refers to a client, who was reached out to in the course of the campaign, and includes numerous demographic, financial, and communication-related factors, along with a marker, whether or not signified whether the client subscribed to a term deposit.

Thea main problem of this study is to develop a predictive model that could identify the likely people who could possibly be potential customers and therefore adopt a term deposit product. Through the utilization of the data mining and machine learning tools, the research hypothesis aims to assist the bank in improving the effectiveness of its marketing campaigns, reducing unnecessary contacts with unresponsive clients, and redistributing the resources in the most efficient way (Ibrahim et al., 2024). As a result, the business aim is to use the dataset to derive patterns and key considerations to Customer subscription choices to create a sound model with the potential of boosting marketing performance and decision-making.

2.2 Aim & Specific Objectives

Objective

1) To analyze and interpret customer demographics, behaviors, and campaign results through descriptive statistics and data visualization to uncover key patterns.
2) To perform data preprocessing by handling missing values, encoding categorical variables, and removing inconsistent or irrelevant records to prepare the dataset for modeling.
3) To develop and compare various predictive models for customer term deposit subscription, evaluate their performance using accuracy metrics, and identify the most effective model.

Scope

The scope of this study is defined by the time period, geographical location and attributes covered in the dataset. The data were obtained on the direct marketing campaigns by a Portuguese banking institution in the period between May 2008 and November 2010 which integrates the customer reaction in that very economic and social environment. The data set has categorical and numerical variables that refer to demographic, financial, and communication-related factors.

Dataset - Bank marketing campaigns dataset

https://www.kaggle.com/datasets/volodymyrgavrysh/bank-marketing-campaigns-dataset

https://creativecommons.org/licenses/by-nc-sa/4.0/

2.3 Methodology Selection

The methodology embraced in this investigation will be the **CRISP-DM (Cross-Industry Standard Process for Data Mining)** as the general guideline to the undertaking of the data mining project. CRISP-DM can be considered one of the best-known and organized data analytics methodologies which provide a structured, iterative process that ensures a scheme of clarity, reproducibility, and reliability in all data mining pipelines. It consists of six major stages, Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. The choice of this methodology is based on the fact that the method is flexible and can be used impartially with a variety of analytical procedures and datasets, which makes it especially convenient in the process of comparative model analysis like the one conducted in the current research.

Phase 1:

The **Business Understanding** phase will focus on the statements of the project goals and requirements in business terms. In this case, the key objective is to predict the kind of bank clients who would mostly subscribe to a term deposit hence making marketing decision-making to be more effective.

Phase 2:

The **Data Understanding** phase will involve acquiring and analysing the bank marketing data obtained on Kaggle. A summary statistics and visualisations which are part of the exploratory data analysis are used to identify data quality anomalies, understand how the features were distributed, and discover inter- variable relationships.

Phase 3:

The **Data Preparation** stage involves data cleaning, filling in of missing data, conversion of categorical data sets into numerical ones and standardisation of data to be used in the modelling. This step will ensure that the dataset is correct, consistent, and best suited to utilize machine-learning algorithms.

Phase 4:

The **Modeling** step is to train and adapt a variety of machine-learning systems, such as logistic regression, decision tree, random forest, gradient boosting, random forest, neural network, bayesian network, and gradient boosting in order to formulate predictive systems. The different parameter settings are explored to attain the balance between accuracy and generalisation at the expense of over-fitting.

Phase 5:

The **Evaluation** phase involves comparative evaluation of all six models based on validation measure used accuracy, misclassification rate, area under the ROC curve (AUC) and log loss. Using the aggregate findings, the most effective or the champion model is identified to be deployed further.

Phase 6:

Lastly, the **Deployment** stage is concerned with the interpretation and reporting of the model outcomes, summary of the important findings and recommendations made on business decisions. CRISP-DM is also suitable in the project since it is conducive to data refining and model comparison, which assures the final results are both data-driven and objectives-driven.

3.0 Modeling

3.1 Exploratory Data Analysis (EDA)

Data exploration

Data Partition Summary

Partition Used	Number of Observations
All data	9,280

Figure 1: Data Partition Summary

This table shows the total size of the dataset we are using. It tells us that there are 9,280 rows of data available to work with for the data exploration and analysis. This is our starting point before we do anything else with the data.

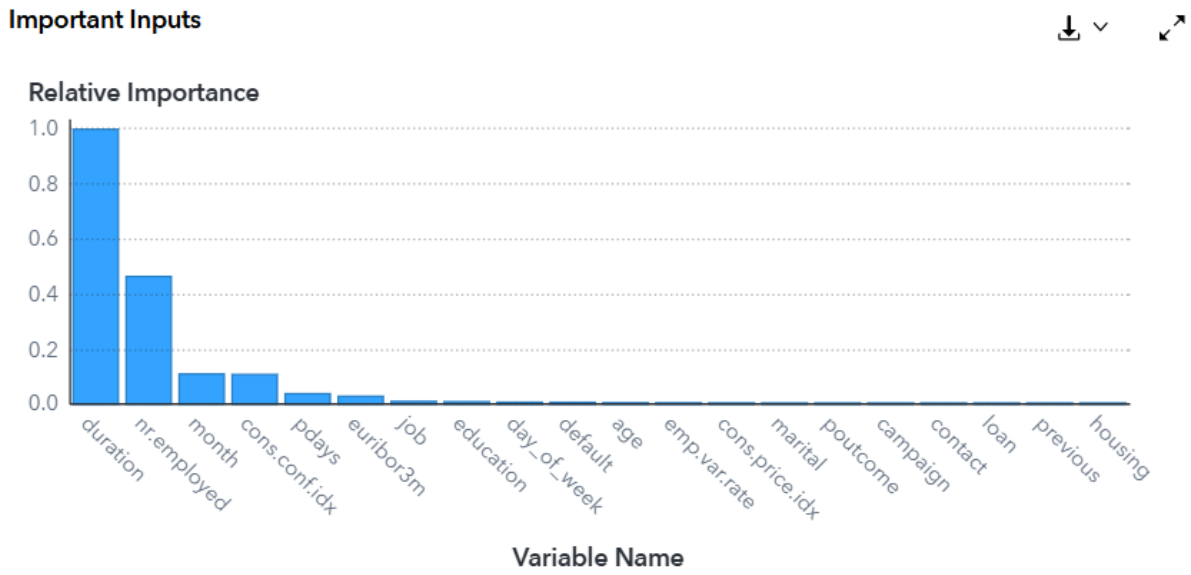

Figure 2: Important Inputs

This table shows the result of the variable importance analysis; we rank input factors from highest to lowest according to their relevance for prediction. Duration, the length of the last phone call is the clear top prize-winning variable (the importance score for duration is a perfect .It has more predictive power about whether or not a given customer will subscribe to a term deposit than any other single predictor). This is succeeded by macroeconomic determinants including "nr. employed" (employees) and "month", with medium relevance, indicating rather that the timing and economic background of the contact are a relevant secondary dimension.

The long tail of variables with very low scores (e.g., “loan”, “housing”, and possibly “poutcome” which gives the number contacts before this campaign, among others) indicates that these details regarding demographic of the client and historical contact have less impact on decision by how much you might have expected. Such a ranking is priceless as it not only confirms the model's rationale, but also offers actionable business insights, i.e. future strategies to prioritize call engagement and economic indicators rather than some of the customer background information.

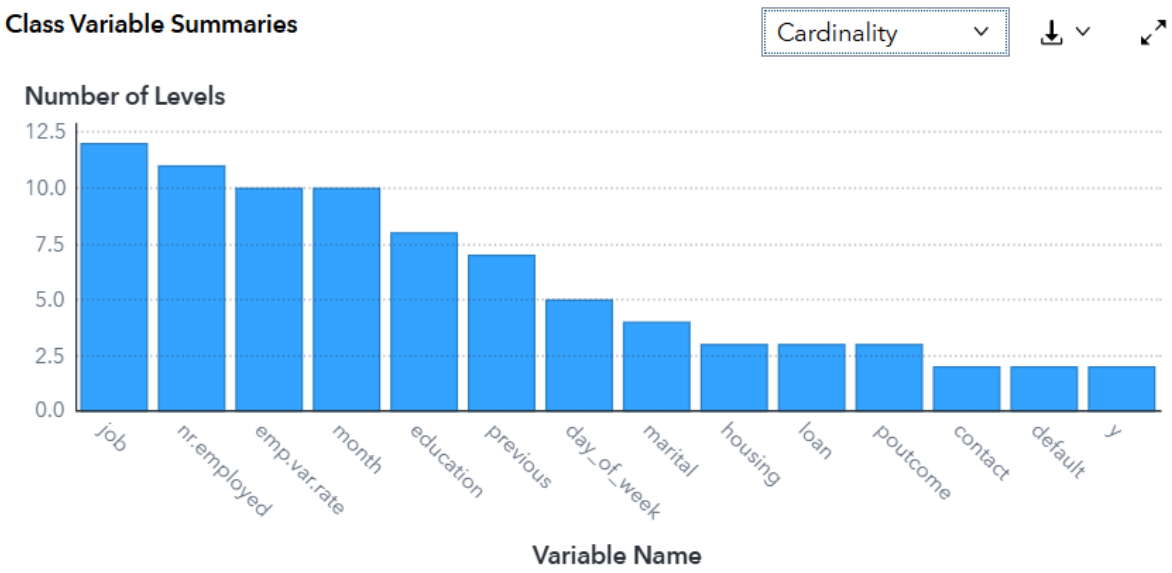

Figure 3: Class Variable Summaries

This summary of categorical variable shows the number of distinct categories that each of text-based variables have. For example, you can see that the number of categories within “job” variables is maximum i.e. 12 which is very large as compared to other variables such as ‘marital’, ‘housing’ etc for which it could be hardly 2-3 like 'yes' or 'no' or 'unknown’. This provides insight as to which features may need to eventually be reduced in the modelling process for performance gain and which are already in a simple, binary form.

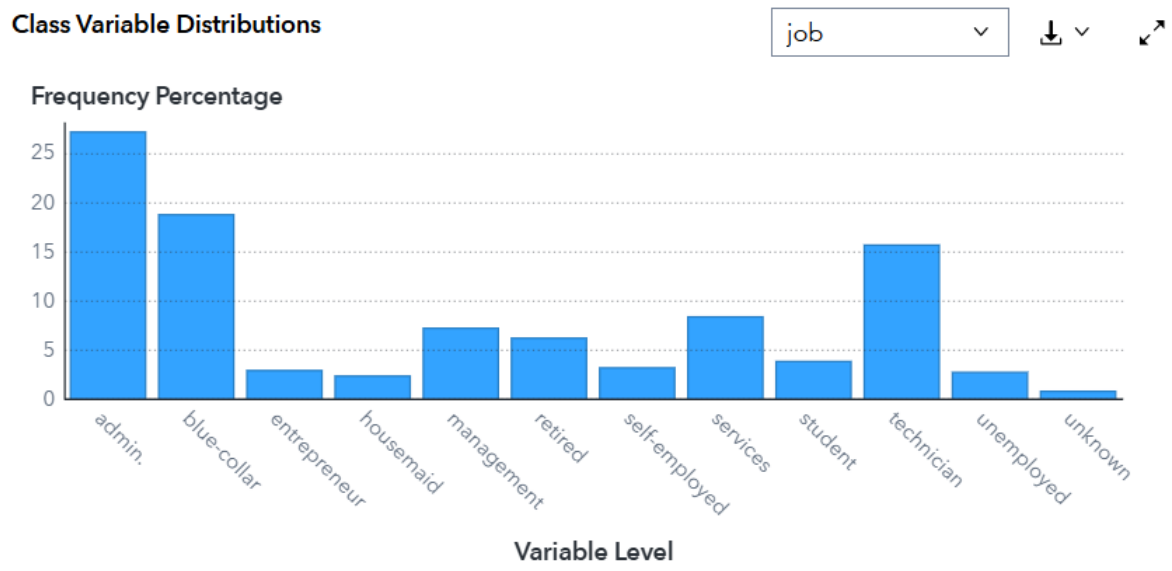

Figure 4: Class Variable Distributions

This class variable distribution plot shows the distribution of the 'job' variable and lets us see how many occurrences we have for each job in our data. Job titles such as 'admin,' 'blue-collar' and 'technician' are some of the most popular. This variable was flagged for inspection as it previously recorded the highest cardinality. You need to check this distribution early while exploring the data for balance and any strong categories which might skew the learning of a model.

Interval Variable Moments

Variable Name	Minimum	Maximum	Mean	Standard Dev...	Skewness	Kurtosis	Relative Varia...	Mean plus 2 SD	Mean minus 2...
age	17	98	40.3844	11.9435	0.9574	1.0788	0.2957	64.2713	16.4974
campaign	1	43	2.3764	2.5074	5.0322	41.3540	1.0551	7.3911	-2.6383
cons.conf.idx	-50.8000	-26.9000	-40.2336	5.3292	0.3466	-0.4358	0.1325	-29.5752	-50.8920
cons.price.idx	92.2010	94.7670	93.4814	0.6334	-0.1166	-0.9635	0.0068	94.7481	92.2146
duration	1	4,199	385.5705	358.9901	2.2196	8.4166	0.9311	1,103.5507	-332.4097
euribor3m	0.6340	5.0450	2.9728	1.8889	-0.0285	-1.9223	0.6354	6.7506	-0.8050
pdays	0	999	889.4205	311.1653	-2.4878	4.1900	0.3499	1,511.7510	267.0899

Figure 5: Interval Variable Moments

Interval Variable Moments yields some significant insights on the data as well. It is observed that “duration” variable has the highest standard deviation (Std Dev = 358.99) which have a right skewed distribution (Skewness = 2.22) showing most of the calls are shorter, however there are also some long call lengths on the higher end as an outlier. The “campaign” variable is very right-skewed (Skewness = 5.03; Kurtosis = 41.35), meaning that most clients were contacted few times during the campaign, with some cases of many contact attempts. For age the distribution is reasonably normal, with slight skewness to the right, and most being about 17-64 years (Within Mean+/-St Devs). The continuous features (economic, etc.) have relatively stable distributions in general shape, and all "pdays" feature shows a weird pattern with left-skewed distribution, and many zeros that may represent clients who were "not previously contacted", meaning most of the client who we've previously contacted are recently called. Knowledge of these patterns is essential for the problem of assessing quality a priori and will directly inform which preprocessing should be undertaken prior to training any model.

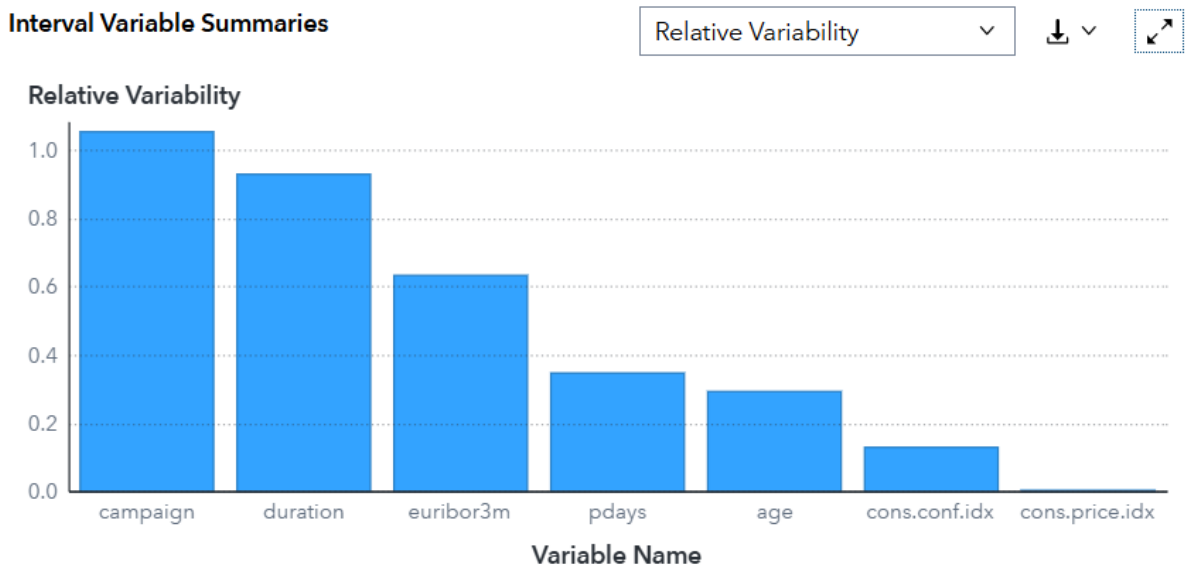

Figure 6: Interval Variable Summaries

These means and Relative Variability to Comparison a snapshot of how school's enrolment is dispersed the measure also tells you whether its mode or median. We observe that "campaign" and "duration" have the highest relative variability, indicating a large spread in the values of these features and little consistency. From a practical standpoint, this indicates that parameters such as the number of contacts per client ("campaign") and the call duration ("duration") will vary widely between individuals. Some variables, however, such as "cons. price. idx" (consumer price index) have extremely low relative variability, meaning that this economic indicator continues to show little variation across observations. Being aware of such variation aids in recognizing which factors exhibit widespread structure within the data and may need to be accounted for.

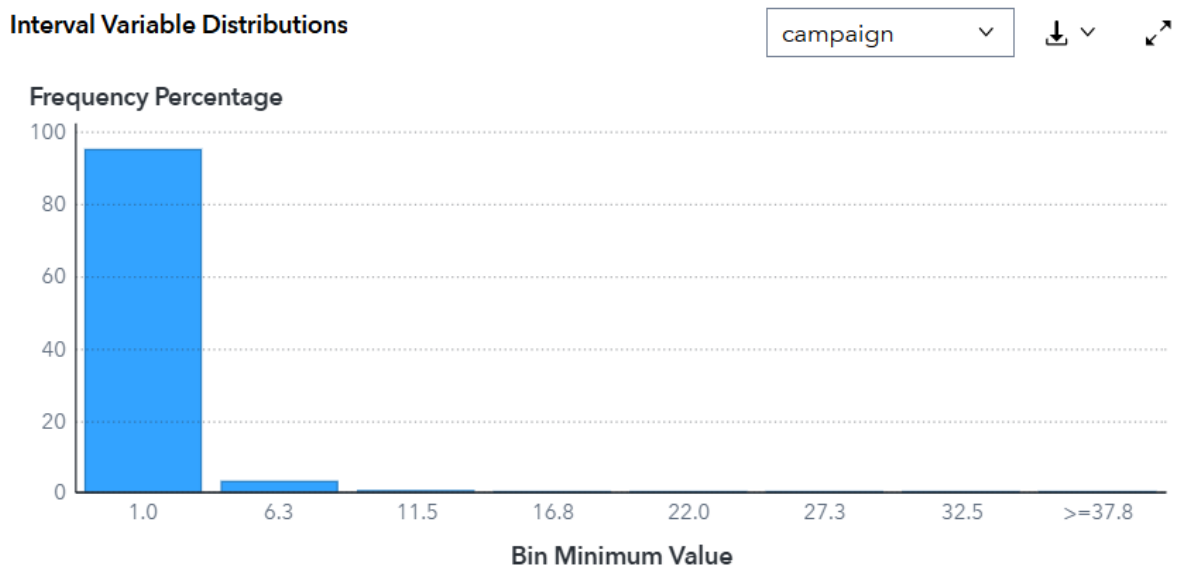

Figure 7: Interval Variable Distributions

This Interval Variable Distributions depicts the distribution of the "campaign" variable, which defines how many contacts were carried on as part of this marketing campaign. Note that the mass of distribution is highly concentrated with a vast majority of clients (the first bar) being engaged only once. The frequency of contact falls off sharply after the initial call and indicating that relatively few clients are called many times. I present this variable on its own not only for reasons listed above, but also because it had the strongest per-value variability amongst all variables identified in previous section as most of values are concentrated at the low end (1 contact), there is a long tail of higher values contributes to proliferation.

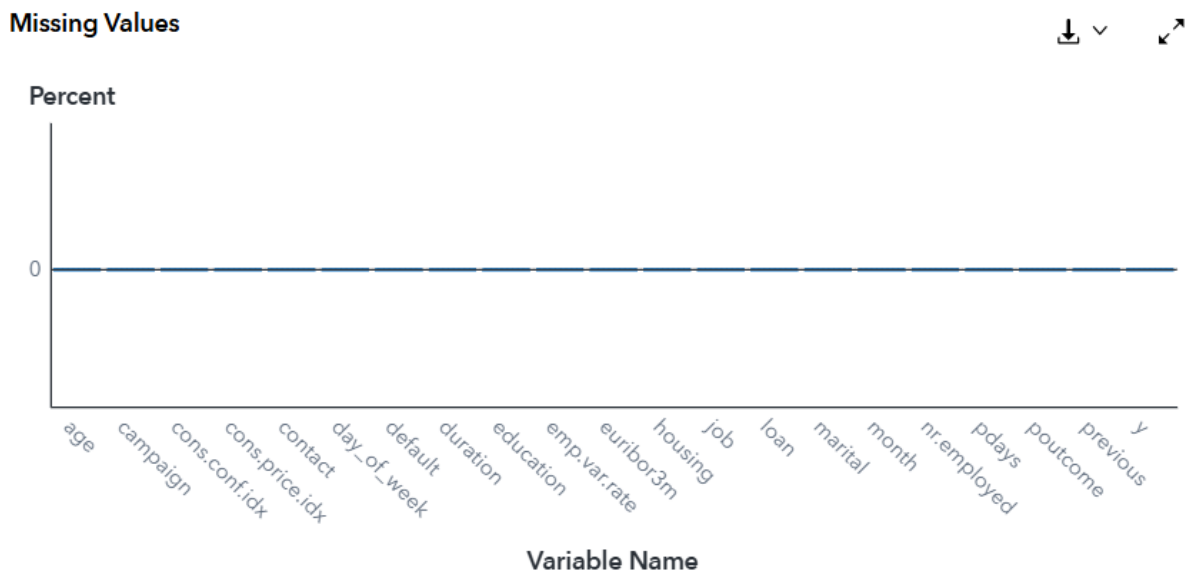

Figure 8: missing value graph

This graph above which means there does not seem to be any missing values in any variable of our data frame. Each variable is fully observed for all observations. This is best result for data quality there is no need for data cleaning or imputation of missing values before further analysis.

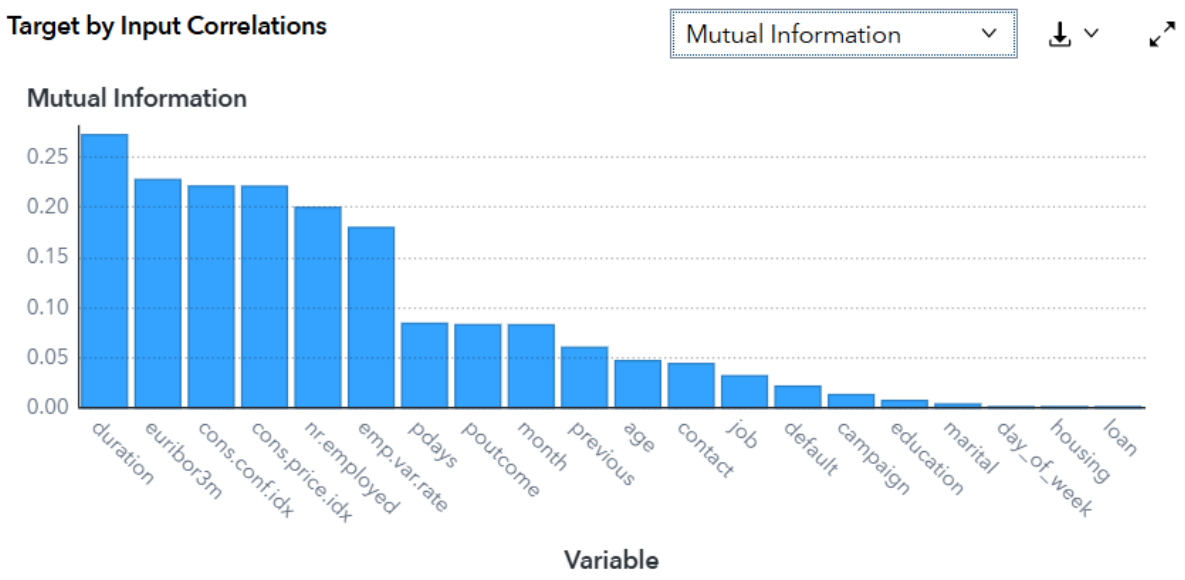

Figure 9: Target by Input Correlations Mutual Information

This plot visualizes the degree of relationship between each input feature and the target through Mutual Information. The most important result is that “duration” have a significantly higher mutual information with the target, respectively meaning it appears to be most correlated to the target outcome. The call duration basically contains more information than any other attribute to determine whether a customer subscribes or not. Other variables like "nr. employed" and "month" are moderately related, most of the other dimensions are shattered; they only give little hint in predicting the target value. This finding supports that call length would be a key indicator for predicting success of the campaign.

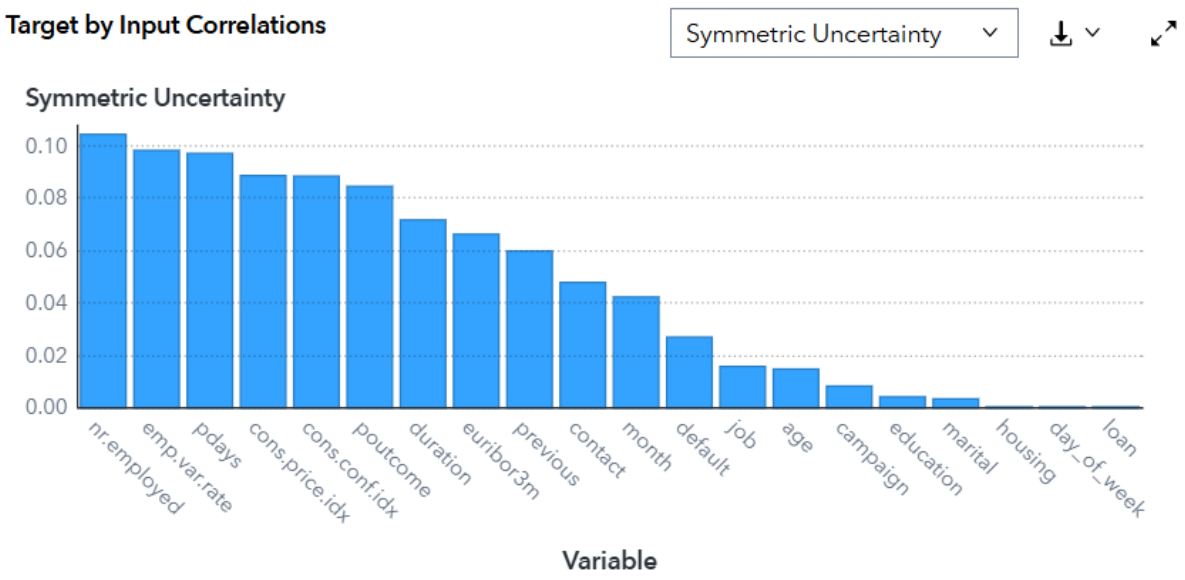

Figure 10: Target by Input Correlations Mutual Information

This Symmetric Uncertainty plot shows the strength of relationship from each predictor to target variable, and it does are that "duration" is also the highest in score indicating “the duration being on the line is best indicator if a customer will subscribe. "euribor3m" or "month", which have a weak dependency against the looking variable but most remaining factors does not usefully contribute for predict the target are useful to discover most significant variables for model.

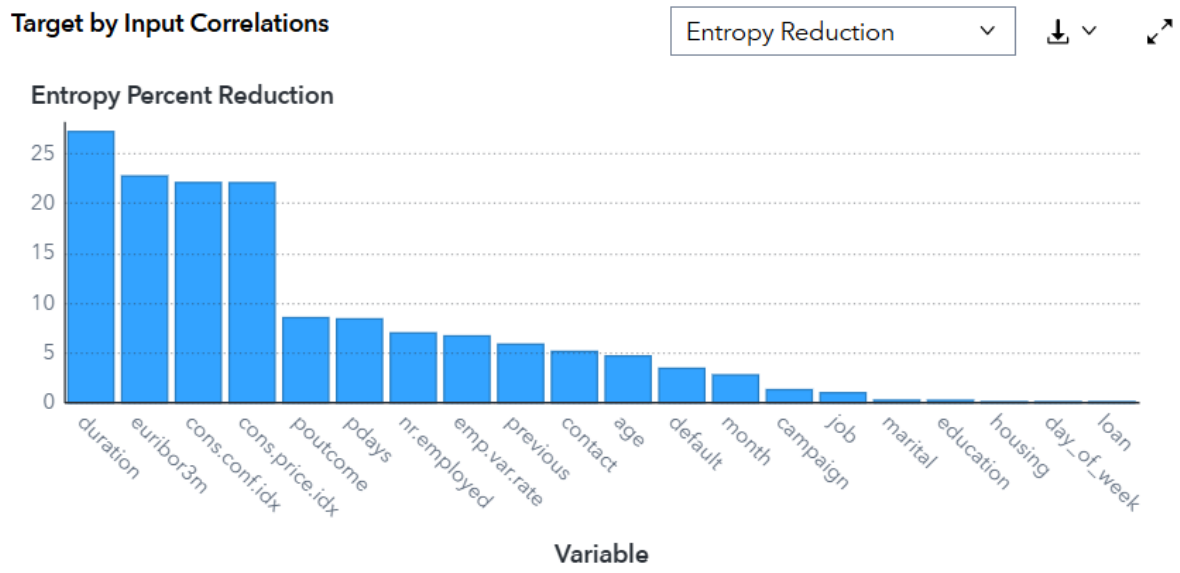

Figure 11: Target by Input Correlations Entropy Reduction

This is another target to be measured by the Input Correlation measures using entropy reduction, a measure of how much each variable reduces uncertainty about the target. "duration" has the largest reduction in entropy (about 25%), indicating it contributes the most information to predicting the target. The others feature like "poutcome" and "month" contribute moderately to the decrease of uncertainty, whereas most of the remaining features give only small reduction in uncertainty. This shows that call duration is the strongest predictor in the model.

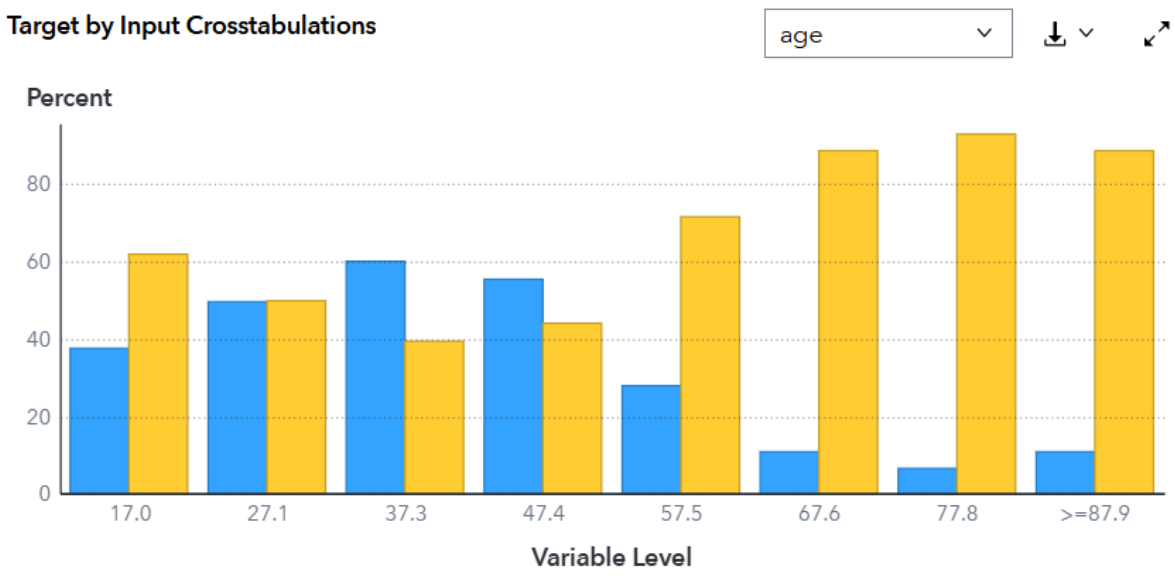

Figure 12: Target by Input Crosstabulations

For each variable, target by input crosstabulations is analysed to determine the connection with the outcome. An example of the "age" variable chart is available. It shows such an evident trend in which the older the client, the higher chance they would subscribe. This pattern is general for all cross-tabulation analysis of all the variables, with one chart per and each showing how the outcome distribution varies across levels of that variable.

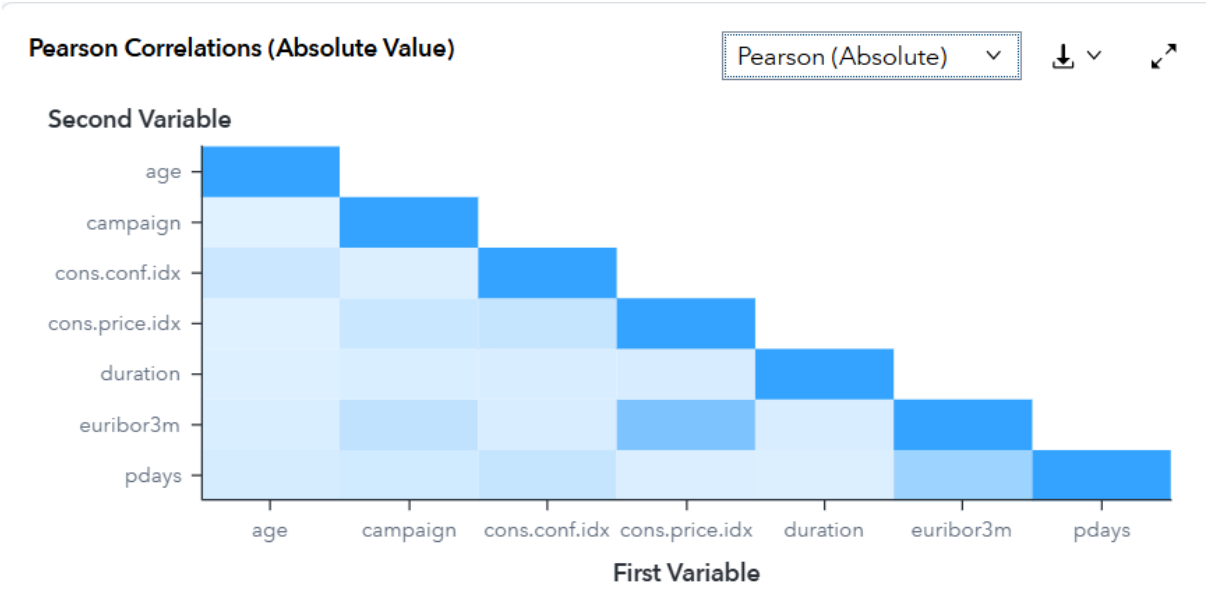

Figure 13: Pearson Correlations

This plot shows the values of Pearson correlations (in module) measured by estimating the linear relationship between numerical variables. This guideline is confirmed by evident patterns in the analysis, for example euribor3m has the highest correlation with emp. var. rate, showing a close linear association between these economic variables. The absolute value of the variable duration causes it to have negative correlations with all the other variables in dataset, explicitly indicating that it affects independently among these factors. So, there are modest correlations between economic things for instance cons. price. idx, euribor3m, and nr. used, which resulted in the evidence of commonality among these macroeconomic variables. This relation structure can be used to explore the relationships between variables and to check for possible multicollinearity present prior to any modelling.

Clustering

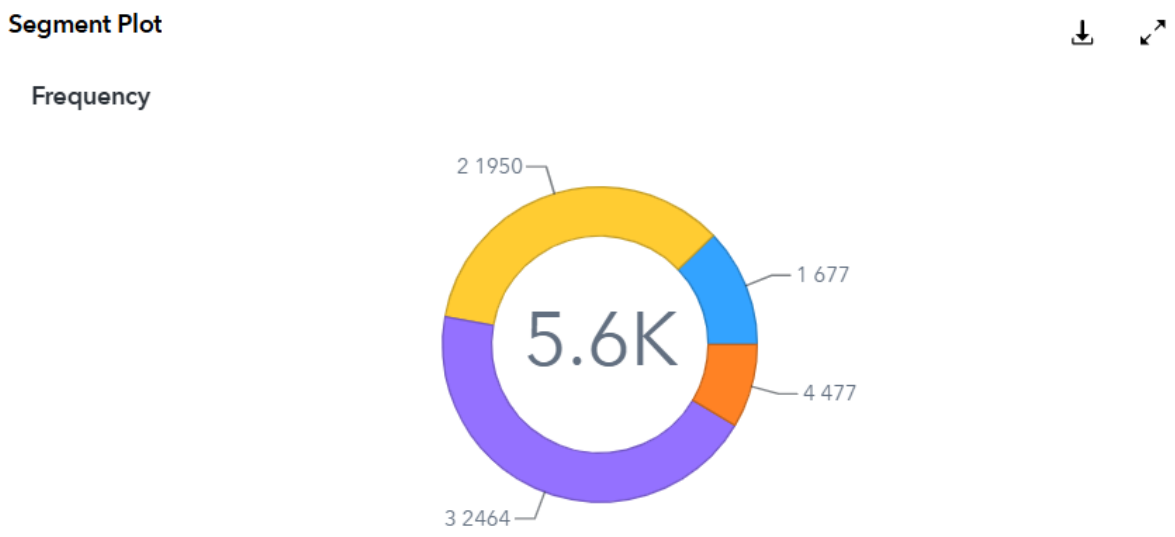

Figure 14: Segment Plot

So, on this plot from the segmentation analysis we have been able to see some clear customer groups in our dataset. Clusters are somewhat evenly distributed, meaning that the customer segments are of different sizes. The biggest family comprises about 5,600 observations, which describes the average customer profile. Two other large clusters include approximately 2,195 and 3,246 patterns respectively; small segments of around 1,677 and 4,477 patterns reveal more focused customer behaviour. This clustering shows that the composition of customers can be naturally divided into several segments, which potentially correspond to different customer behaviour and demographic aspects that might be targeted for marketing purposes.

Figure 15: ABC Statistics

Gap Statistics chart is used to find out optimal number of clusters for customer segmentation. The analysis reveals that the gap statistic reaches its maximum at 3 clusters, suggesting we should choose 3 customer segments. The value of the gap that remains high (but not increasing any more) for 4 and 5 clusters but reaching a maximum at 3 clusters is. However, a clear indication that partitioning customers into three different groups provides an optimal trade-off between segment distinctiveness and model simplicity. This best-in-class clustering will allow to identify the three major customer profiles based on substantial differences in behaviour or demographics, making targeted marketing approaches possible.

Cluster Centroids

Cluster ID	age	campaign	cons.conf.idx
1	37.3026	1.9662	-40.6040
2	39.6530	2.1470	-41.9992
3	40.2963	2.7776	-39.2089
4	47.0399	1.6080	-37.8911

cons.price.idx	duration	euribor3m	pdays
93.1303	367.6713	1.1761	542.7250
93.2765	383.9015	2.3566	989.5770
93.7160	386.1811	4.1962	991.8198
93.5596	360.8192	1.0427	313.6596

Standard: age	Standard: campaign	Standard: cons.conf.idx	Standard: cons.price.idx
-0.2457	-0.1640	-0.0620	-0.5505
-0.0485	-0.0900	-0.3237	-0.3188
0.0055	0.1682	0.1997	0.3776
0.5712	-0.3107	0.4469	0.1298

Standard: duration	Standard: euribor3m	Standard: pdays	contact
-0.0380	-0.9342	-1.0929	cellular
0.0074	-0.3094	0.3272	cellular
0.0138	0.6644	0.3343	cellular
-0.0572	-1.0048	-1.8209	cellular

day_of_week	default	education	emp.var.rate
thu	no	university.degree	-1.8000
thu	no	university.degree	-1.8000
tue	no	university.degree	1.4000
tue	no	high.school	-1.1000

housing	job	loan	marital
yes	admin.	no	single

housing	job	loan	marital
no	blue-collar	no	married
yes	admin.	no	married
no	admin.	no	married

month	nr.employed	poutcome	previous
may	5,099.1000	failure	1
may	5,099.1000	nonexistent	0
jul	5,228.1000	nonexistent	0
oct	4,963.6000	success	1

Figure 16: Cluster Centroid

The following "Four Segments" table elucidates the distinguishing features of the 4 customer segments. Cluster 1 is characterized by younger singles with administrative employment and lower economic status. Group 2 Married, moderate economic exposure customers. Cluster 3 consists of customers who were addressed in good economic situations (high euribor3m and employment variation rates). Cluster 4 Characterizes old, married customers which are with previous campaign success and were contacted on low employees' duration. Compute the human factors and economic indexes, including each cluster's age, contact time, previous campaign history so that you can perform targeted marketing strategies for different customer segments.

Segmentation

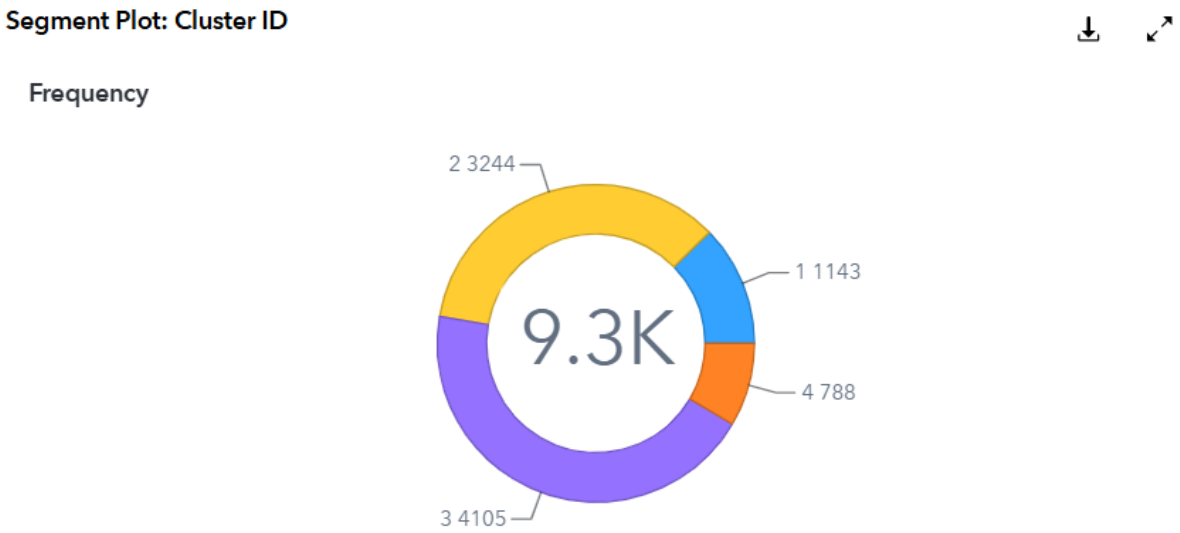

Figure 17: Segment Plot: Cluster ID

This segment profile demonstrates the variation in size of the four identified customer groups, which differs substantially by group membership. Cluster 2 is the predominant cluster with 9300 observations and thus consists of most common customer profile in the dataset. The remaining clusters have much fewer numbers of members with 2,324 customers in Cluster 1, 1,143 in Cluster3 and the smallest cluster was number 4 comprising of only 788 members. This distribution of customer base size is important to appreciate as it can lead to decisions regarding the allocation of resources including strategic importance given to larger segments and deriving marketing approaches for niche customer groups.

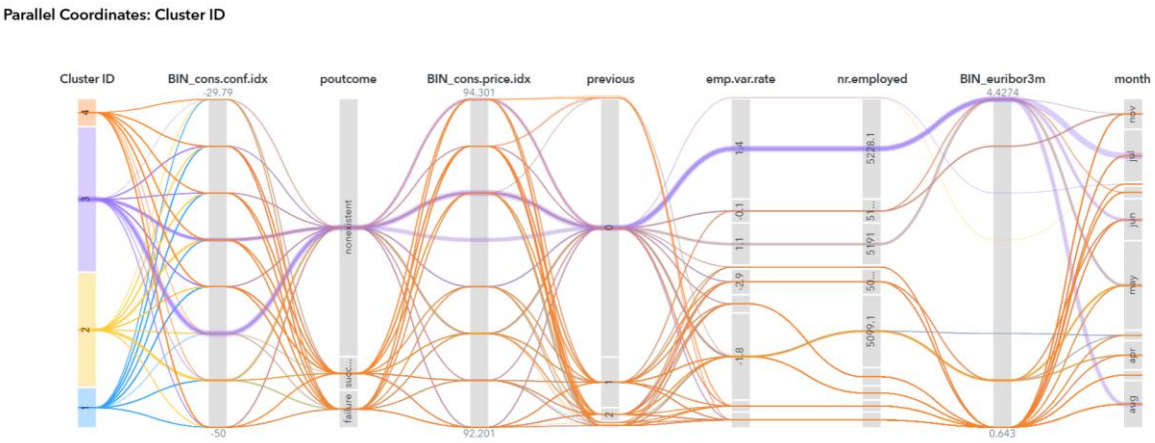

Figure 18: Parallel Coordinates

The parallel coordinates plot illustrates the primary features distinguishing the four customer segments on multiple variables at once. Each row shows a cluster's profile in terms of metrics, how they rank across e.g. economic measures (euribor3m, nr. employed), previous campaign results (poutcome) and timing of the contact (month). The fact that the coloured lines all travel along different routes show that each segment has a unique mix of properties. For instance, one cluster may exhibit a high economic profile and a positive past campaign history whereas another shows lower economic indicators and little campaign experience. This presentation will aid in interpreting which variable relationships to the segments drive differentiation of the segments.

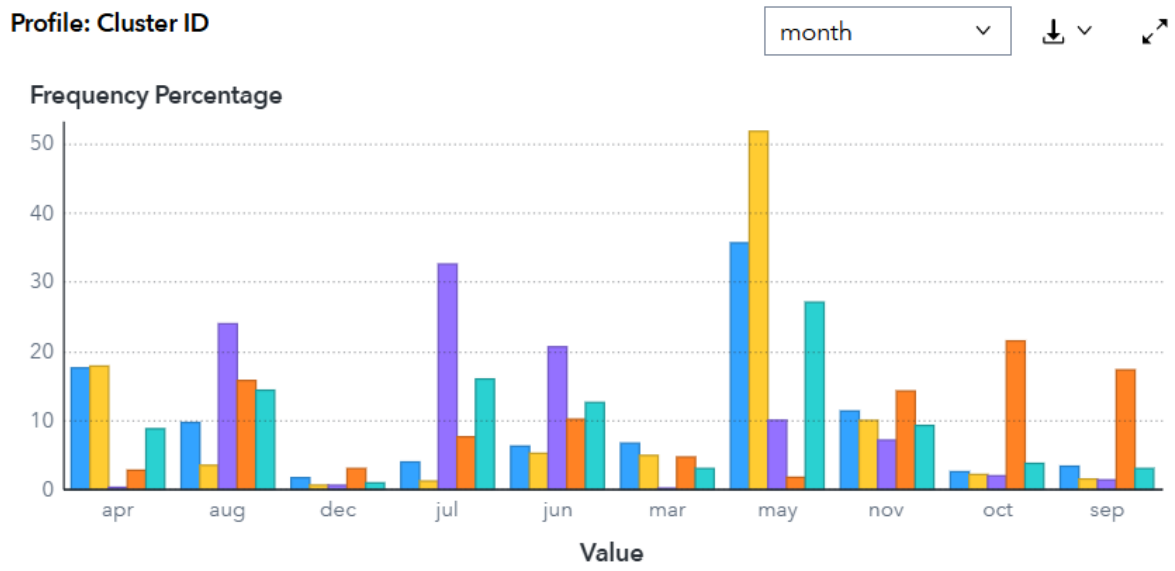

Figure 19: Segment Profile

This profile chart segment examines how the contact "month" variable is distributed among the four customers clusters that have emerged. Each coloured line represents a cluster and counts of the percentage of customers in contact are shown for each month across the year. The contact patterns of all segments are pronounced and seasonally defined, as can be seen on the plot. For example, a cluster was mainly contacted during the spring months with emphasis on May, whereas another appears to favour contacts later in the year in July and October. This suggests that the time of the marketing campaign was not uniform for all customers and may have been based on strategies or customer availability for each cluster.

The "month" is used here as an example for illustrating the segment profiling procedure. The same analysis process used to analyse the distribution of a variable across all clusters was repeated for every other variable included in the dataset to gain insight into each unique cluster feature. By profiling every variable in this way, a full portrait of each type of customer comes into view, which is vital to creating targeted and successful marketing plans based on that segment's unique profile.

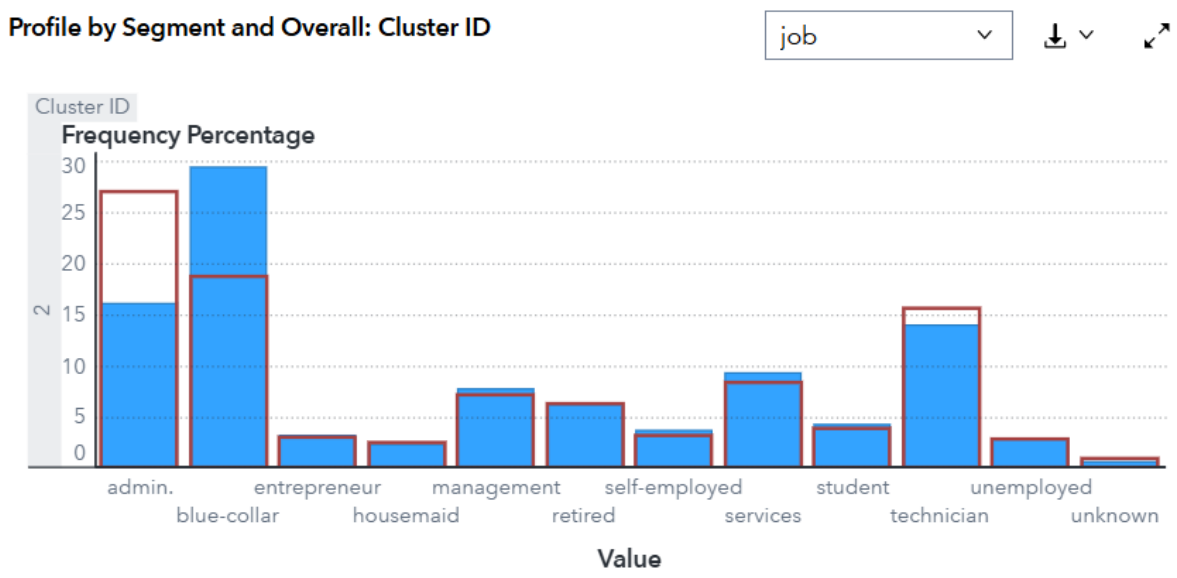

Figure 20: Profile by Segment and Overall

This chart compares how a specific attribute is spread out across the 4 customer segments and population. It shows that each cluster has its own demographic fingerprint. For example, one cluster is characterized as being "admin" workers in nature while another can be classified as being "blue-collar" in nature and a third indicates there are many "retired" present. This confirms that the clustering solution has managed to cluster customers with similar occupation together as occupation is a relevant discriminator between the clusters.

This diagnosis of the "job" variable is a model for the full profiling applied on the whole dataset. A similar comparative process is then applied to each of the other variables to develop a multifaceted profile for every segment. Constructing all the variables in this manner systematically, the analysis identifies significant defining characteristics for each cluster. This deep level of understanding is crucial in the creation of highly targeted marketing strategies, so that those messages and offers get delivered to the most suitable groups based on their entire profile.

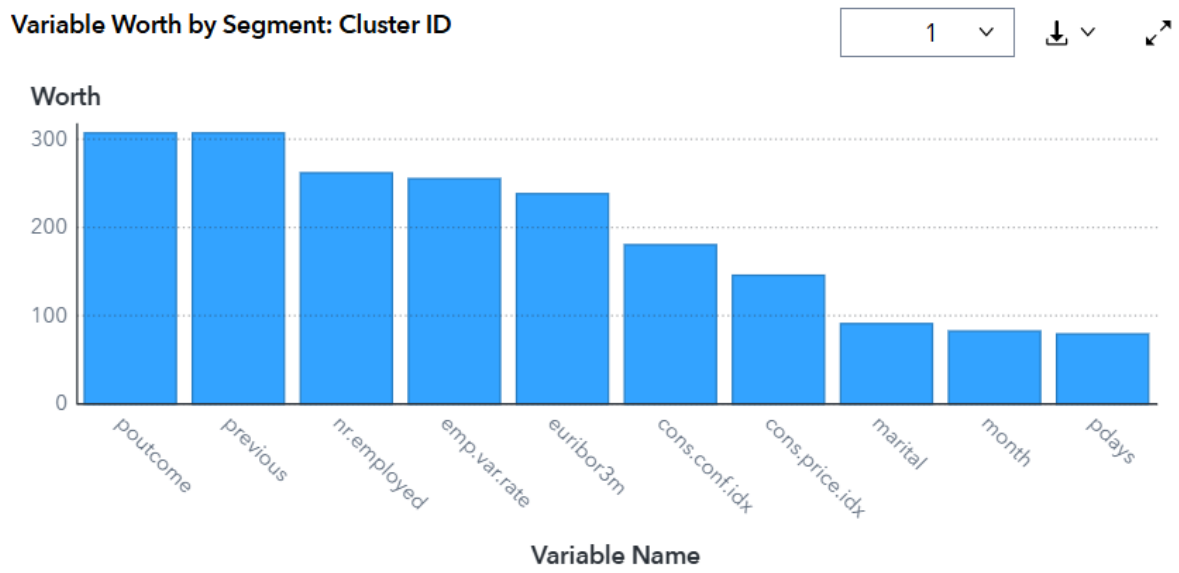

Figure 21: Variable Worth

The Variable Worth by Segment tells us that which variables are having the most effect on shaping and describing the four groups of customers. According to the analysis, poutcome (previous campaign outcome) has the highest worth score it is considered as the most important feature of segment differentiation. Other relevant variables are "previous" (number of previous contacts), "nr. used" (employment rate); as well as economic indexes "euribor3m" and "cons. conf. idx" (consumer confidence). On the other hand, “marital” type of variable displays a very low worth value and as well, this variable has not been useful to differentiate clusters. Prioritizing is based on the importance of each factor in defining the unique segment identity for targeted marketing.

3.2 Data Cleaning / Pre-processing

3.2.1 Basic Pre-Processing Setting

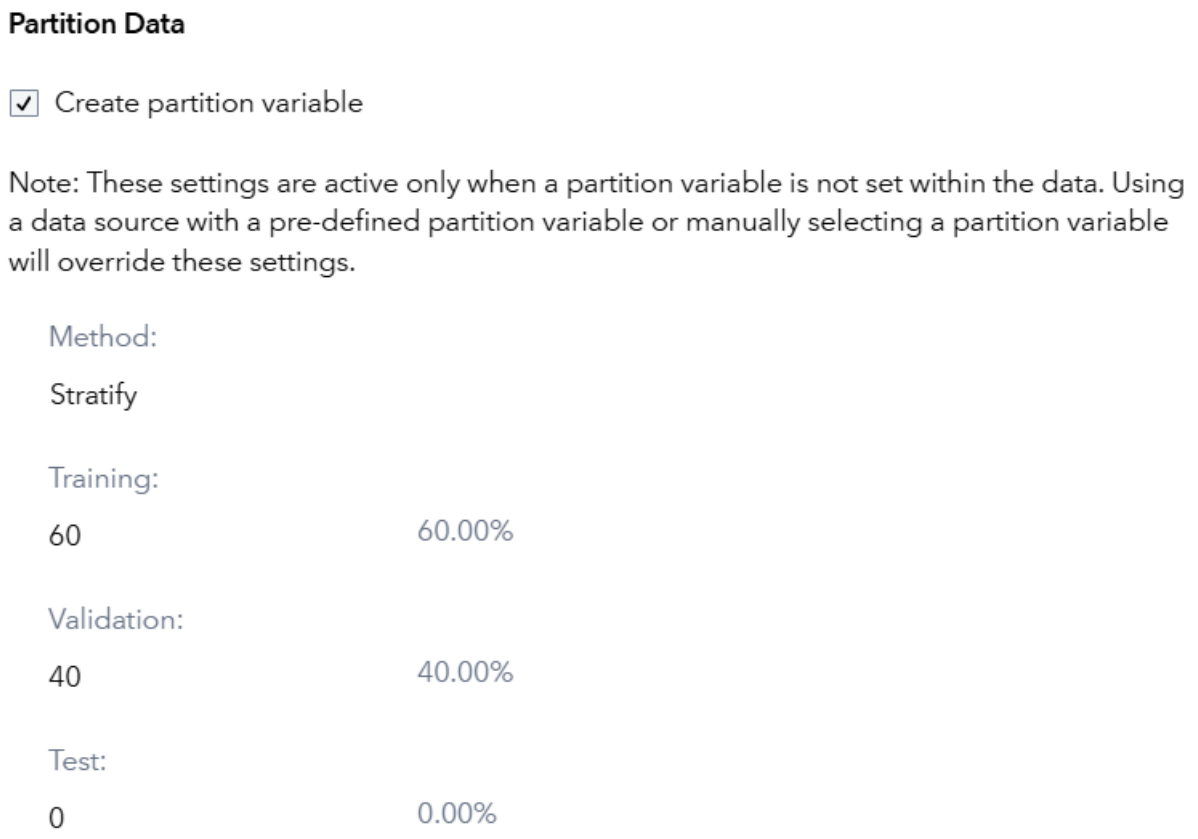

Figure 22: Partition Data

These settlings define the partitioning of data during modelling. Datasets are split according to the stratified sampling, such that target variable outcome distribution is consistent between train and test datasets. More precisely, 60% of the observations go to the training set used to construct the predictive model. The 40% of the data is assigned to the validation set for performance evaluation and parameters' optimization. No data is allocated specifically to a test set in this setup, so the validation set is then effectively being used as the holdout sample to estimate how well our final model will generalize when it's deployed

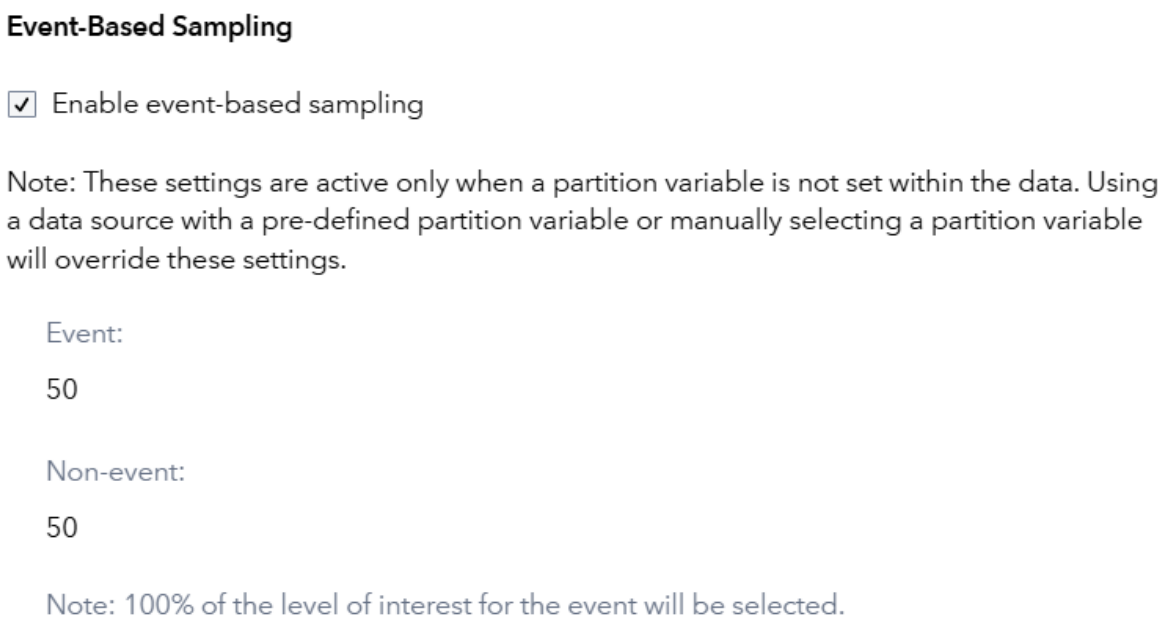

Figure 23: Event-Based Sampling

This setting allows event-based sampling to deal with class-imbalance in the dataset. Here, the sampling will ensure we have a balanced partition i.e. by randomly selecting events and non-events with an equal 50-50 percent distribution. This allows the model to have adequate example of the rare target outcome (the "event") during training, preventing it from becoming too biased towards predicting for majority class. Through artificially balancing the classes, the sampling procedure assists the model to learn the patterns of both outcomes resulting in a better predictability value for the present event.

Figure 24: Model Comparison

This setting specifies that Accuracy is the primary metric used to evaluate and compare the performance of different classification models. By selecting accuracy as the class selection statistic, the model comparison process will rank all trained models based on their overall correctness and the model with the highest accuracy score will be chosen as the best performing model for the given classification task.

3.2.2 Decision Tree, Neural Network, Logistic Regression, & Forest Pre-Processing Setting

Filtering

Output

Excluded Class Values

Obs	Variable	Label	Role	Level	Train Count	Train Percent	Filter Method
1	education		INPUT	illiterate	5	0.0897988506	MINPCT
2	emp.var.rate		INPUT	-0.2	1	0.0179597701	MINPCT
3	marital		INPUT	unknown	12	0.2155172414	MINPCT
4	nr.employed		INPUT	5176.3	1	0.0179597701	MINPCT
5	previous		INPUT	6	3	0.0538793103	MINPCT
6	previous		INPUT	5	9	0.161637931	MINPCT
7	previous		INPUT	4	30	0.5387931034	MINPCT

Figure 25: Filtering

The Filtering node is executed to identify and exclude rare or infrequent categories from categorical input variables. This is a crucial preprocessing step, as it prevents the model from basing predictions on categories with insufficient data, which can lead to overfitting and poor generalization.

Justification:

A review of the "Excluded Class Values" report from the Filtering step confirms its automatic application using the MINPCT (Minimum Percentage) method. The node successfully identified and removed rare categories, such as "illiterate" in the education variable (with only 5 occurrences) and low-frequency values in variables like previous and marital. This process enhances the model's stability and reliability by ensuring it learns from robust, well-represented patterns in the data, thereby improving its predictive accuracy on new, unseen data.

Imputation

Output

Input Variable Statistics

Obs	Input Variable	Measurement Level	Number of Missing Values	Percentage Missing	Imputable	Minimum	Maximum	Mean	Midrange	Standard Deviation	Skewness	Kurtosis	Label
1	duration	INTERVAL	0	0	0	1	4199	381.2577227	2100.00	357.12262197	2.3350266072	9.7955643318	
2	emp.var.rate	NOMINAL	0	0	0	.	.	.	.	.	.	.	
3	euribor3m	INTERVAL	0	0	0	0.634	5.045	2.9410335848	2.84	1.8892486198	0.0066644537	-1.924612524	
4	GRP_cons .price.idx	ORDINAL	0	0	0	.	.	.	.	.	.	.	Grouped cons.price.idx
5	GRP_duration	ORDINAL	0	0	0	.	.	.	.	.	.	.	Grouped duration
6	GRP_euribor3m	ORDINAL	0	0	0	.	.	.	.	.	.	.	Grouped euribor3m
7	GRP_month	ORDINAL	0	0	0	.	.	.	.	.	.	.	Grouped month
8	GRP_nr .employed	ORDINAL	0	0	0	.	.	.	.	.	.	.	Grouped nr.employed
9	GRP_poutcome	ORDINAL	0	0	0	.	.	.	.	.	.	.	Grouped poutcome
10	pdays	INTERVAL	0	0	0	0	999	886.62176724	499.50	314.65145005	-2.44320282	3.9708144496	
11	WOE_duration	INTERVAL	0	0	0	-1.834246743	1.955004879	0.0154620801	0.06	1.3676550084	0.0875631159	-1.150589325	Weightof Evidence: duration
12	WOE_job	INTERVAL	0	0	0	-1.15780116	0.5990793227	-0.007641601	-0.28	0.476757845	-0.952239725	0.9100236373	Weightof Evidence: job
13	WOE_month	INTERVAL	0	0	0	-3.014308659	0.6034184106	-0.064499986	-1.21	0.8266554063	-2.071905579	3.8882309035	Weightof Evidence: month
14	WOE_nr.employed	INTERVAL	0	0	0	-2.801026658	1.3452538112	-0.100462909	-0.73	1.2669897497	-0.730577846	-0.885285768	Weightof Evidence: nr.employed

Figure 26: Imputation

The Imputation node is executed first to identify and remediate any missing (null) values. This is a mandatory step, as most machine learning algorithms cannot process incomplete records.

Justification:

A review of the "Input Variable Statistics" report from the Imputation step confirms the high quality of the source data. All input variables processed show 0 missing values and a 0% missing percentage. While no data modification was required, this step serves as a critical validation checkpoint, formally verifying the dataset's integrity and completeness before transformation.

Anomaly Detection

Output

The SAS System

The SVDD Procedure

Number of Observations Read	5224
Number of Observations Used	5224

Model Information	
Optimization Method	Active Set
Kernel Type	RBF
RBF Kernel Bandwidth	0.90561854
Bandwidth Selection Method	ModifiedMean
Bandwidth Relative Scale	1.45594543
Expected Outlier Fraction	0.01
Optimization Tolerance	0.001
Standardization	True
Number of Interval Variables	7
Number of Nominal Variables	0

Training Results	
Number of Support Vectors	297
Number of Support Vectors on Boundary	297
Number of Dropped Observations	0
Threshold R Square Value	0.99400
Constant (C_r) Value	0.00727
Run Time (seconds)	0.43078
Bandwidth Calculation Time (seconds)	0.00000596

Optimization Summary	
Number of Iterations	1
Objective Value	0.007272287
Infeasibility	0.0009806617
Optimization Status	Optimal
Degenerate	No

Status Information	
Instance	Status
Model1	Success

Output CAS Tables			
CAS Library	Name	Number of Rows	Number of Columns
Analytics_Project_0913608a-22aa-448d-873f-8024dcbcbeb7	_AKRFFQQKO1EQTVG7Y57XOWFRH_AST	1	2

Figure 27: Anomaly Detection

The Anomaly Detection node is executed to identify unusual patterns or outliers in the data by modelling the boundary of "normal" observations. This is a crucial analytical step for fraud detection, quality control, or data cleansing, as it helps flag records that deviate significantly from the majority.

Justification:

Review of the procedure output for SVDD (Support Vector Data Description) indicates that training was successful on 5,224 observations. Optimal model is built using 297 support vectors for data boundary and an RBF kernel. With a high threshold R-square value of 0.994 and an optimal optimization status, the model can effectively capture differences between normal patterns and anomalies. This methodology adds to data quality and risk assessment, by implementing an efficient way to detect outliers which may indicate a need for further investigational work.

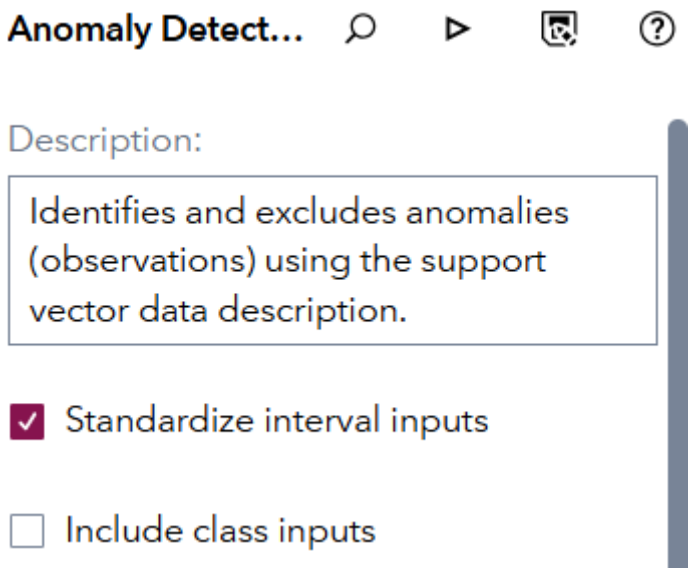

Figure 28: Anomaly Detection Tune

The configuration for the Anomaly Detection node involves two key settings. The option to standardize interval inputs is essential to tick because it rescales all numerical variables to a common range, preventing variables with larger natural scales (like income or duration) from dominating the anomaly detection process and ensuring all inputs contribute equally to identifying outliers.

3.2.3 Bayesian Network & Gradient Boosting Pre-Processing Setting

Interactive Grouping

The SAS System

Results Summary (Training)

Obs	Display Variable	Gini Statistic	Information Value	Group Variable	Weight of Evidence	Level for Interactive	Calculated Role	New Role	Pre-Defined Grouping	Level	Minimum	Maximum	Label	Type	Information Value Ordering
1	duration	60.136	1.473	GRP_duration	WOE_duration	INTERVAL	INPUT	DEFAULT		INTERVAL	1	4199		N	1
2	nr.employed	54.924	1.272	GRP_nr.employed	WOE_nr.employed	NOMINAL	INPUT	DEFAULT		NOMINAL	4963.6	5228.1		N	2
3	euribor3m	51.074	1.114	GRP_euribor3m	WOE_euribor3m	INTERVAL	INPUT	DEFAULT		INTERVAL	0.634	5.045		N	3
4	emp.var.rate	52.736	1.098	GRP_emp.var.rate	WOE_emp.var.rate	NOMINAL	INPUT	DEFAULT		NOMINAL	-3.4	1.4		N	4
5	pdays	20.043	0.607	GRP_pdays	WOE_pdays	INTERVAL	INPUT	DEFAULT		INTERVAL	0	999		N	5
6	poutcome	23.470	0.594	GRP_poutcome	WOE_poutcome	NOMINAL	INPUT	DEFAULT		NOMINAL	.	.		C	6
7	month	30.968	0.514	GRP_month	WOE_month	NOMINAL	INPUT	DEFAULT		NOMINAL	.	.		C	7
8	cons.price.idx	26.613	0.399	GRP_cons.price.idx	WOE_cons.price.idx	INTERVAL	INPUT	DEFAULT		INTERVAL	92.201	94.767		N	8
9	previous	22.620	0.393	GRP_previous	WOE_previous	NOMINAL	INPUT	DEFAULT		NOMINAL	0	6		N	9
10	contact	21.336	0.238	GRP_contact	WOE_contact	BINARY	INPUT	DEFAULT		BINARY	.	.		C	10
11	job	23.333	0.212	GRP_job	WOE_job	NOMINAL	INPUT	DEFAULT		NOMINAL	.	.		C	11
12	cons.conf.idx	21.835	0.169	GRP_cons.conf.idx	WOE_cons.conf.idx	INTERVAL	INPUT	DEFAULT		INTERVAL	-50.8	-26.9		N	12
13	default	12.572	0.123	GRP_default	WOE_default	NOMINAL	INPUT	DEFAULT		NOMINAL	.	.		C	13
14	age	15.083	0.080	GRP_age	WOE_age	INTERVAL	REJECTED	DEFAULT		INTERVAL	17	98		N	14
15	education	12.293	0.050	GRP_education	WOE_education	NOMINAL	REJECTED	DEFAULT		NOMINAL	.	.		C	15
16	campaign	9.390	0.033	GRP_campaign	WOE_campaign	INTERVAL	REJECTED	DEFAULT		INTERVAL	1	34		N	16
17	marital	6.938	0.022	GRP_marital	WOE_marital	NOMINAL	REJECTED	DEFAULT		NOMINAL	.	.		C	17
18	housing	2.829	0.004	GRP_housing	WOE_housing	NOMINAL	REJECTED	DEFAULT		NOMINAL	.	.		C	18
19	day_of_week	2.846	0.003	GRP_day_of_week	WOE_day_of_week	NOMINAL	REJECTED	DEFAULT		NOMINAL	.	.		C	19
20	loan	1.106	0.003	GRP_loan	WOE_loan	NOMINAL	REJECTED	DEFAULT		NOMINAL	.	.		C	20

Figure 29: Interactive Grouping

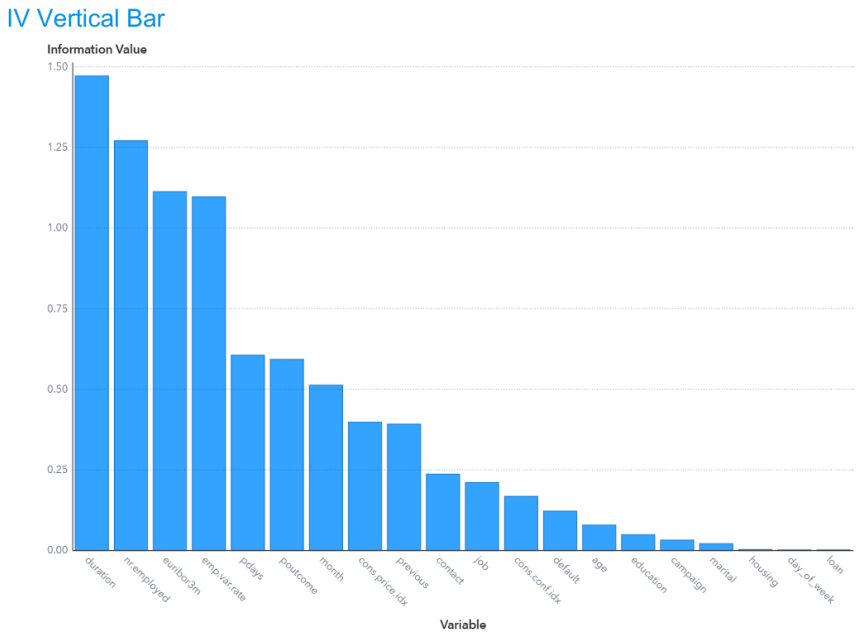

Figure 30: IV Vertical Bar

This is an important step in interactive grouping to enhance the predictive power of the raw variables. It transforms the variables for non-linear relationships and does an initial screening to exclude non-informative features.

Justification:

Ranking Predictive Power: The "IV Vertical Bar" chart and the "Results Summary (Training)" table objectively rank all variables by their Information Value (IV). This analysis uncovers the most influential drivers for this project, such as duration with IV = 1.473, nr.employed with IV = 1.272, and euribor3m with IV = 1.114.

Intelligent Transformation: This node creates new high-value features. By binning continuous variables, the node creates GRP_ variables. It also generates WOE_ variables that replace categories with a numerical score reflecting their predictive power. This is the best standard practice to capture complex nonlinear patterns.

Efficient Noise Reduction: The process is further justified by its successful screening of the non-predictive variables. It correctly identified and "REJECTED" the features that have a low value of information, like age - IV = 0.080, education - IV = 0.050, marital - IV = 0.022, and loan - IV = 0.003. This removes statistical noise and reduces the complexity, hence avoiding poor performance by non-informative data in the final model.

Variable Selection

Output

Fast Variable Selection

The VARREDUCE Procedure

Number of Observations Read	5568
Number of Observations Used	5568

Selection Summary

Iteration	Parameter	Proportion of Variance Explained	SSE	MSE	AIC	AICC	BIC
1	WOE_duration	0.290003	0.709997	0.00012754	-0.340699	1.659303	-0.340946
2	WOE_nr.employed	0.511373	0.488627	0.00008779	-0.713641	1.286361	-0.713058
3	duration	0.530048	0.469952	0.00008445	-0.751892	1.248111	-0.750478
4	WOE_month	0.539162	0.460838	0.00008282	-0.770758	1.229247	-0.768514
5	pdays	0.545269	0.454731	0.00008174	-0.783381	1.216626	-0.780305
6	emp.var.rate -2.9	0.551695	0.448305	0.00008060	-0.796894	1.203115	-0.792988
7	GRP_duration 2	0.556300	0.443700	0.00007979	-0.806501	1.193510	-0.801765
8	WOE_job	0.558954	0.441046	0.00007932	-0.811782	1.188232	-0.806214
9	GRP_nr.employed 3	0.561080	0.438920	0.00007896	-0.815895	1.184121	-0.809497
10	GRP_cons .price.idx 3	0.562598	0.437402	0.00007870	-0.818640	1.181379	-0.811412
11	GRP_month 6	0.565781	0.434219	0.00007814	-0.825225	1.174797	-0.817166
12	GRP_euribor3m 2	0.569478	0.430522	0.00007749	-0.833060	1.166966	-0.824170
13	emp.var.rate 1.4	0.572507	0.427493	0.00007696	-0.839402	1.160628	-0.829681
14	emp.var.rate -1.7	0.574224	0.425776	0.00007666	-0.842707	1.157327	-0.832157
15	GRP_cons .price.idx 4	0.575777	0.424223	0.00007640	-0.845642	1.154397	-0.834260
16	GRP_poutcome 3	0.577039	0.422961	0.00007618	-0.847904	1.152139	-0.835692
17	emp.var.rate -3.4	0.578146	0.421854	0.00007600	-0.849807	1.150241	-0.836764
18	euribor3m	0.580270	0.419730	0.00007563	-0.854134	1.145919	-0.840261
19	GRP_cons .price.idx 2	0.581289	0.418711	0.00007546	-0.855848	1.144210	-0.841145
20	GRP_month 11	0.581892	0.418108	0.00007536	-0.856571	1.143493	-0.841036

Selected Effects

Number	Selected Variable	Variable Type
1	WOE_duration	INTERVAL
2	WOE_nr.employed	INTERVAL
3	duration	INTERVAL
4	WOE_month	INTERVAL
5	pdays	INTERVAL
6	emp.var.rate	CLASS
7	GRP_duration	CLASS
8	WOE_job	INTERVAL
9	GRP_nr.employed	CLASS
10	GRP_cons .price.idx	CLASS
11	GRP_month	CLASS
12	GRP_euribor3m	CLASS
13	GRP_poutcome	CLASS
14	euribor3m	INTERVAL

Figure 31: Variable Selection

Variable Selection is the final stage of pre-processing, wherein the feature set from the pool of expanded original, GRP_, and WOE_ variables is optimized. It selects the most parsimonious and predictive combination of features, eliminating redundancy.

Justification:

High Model Efficacy: The "Selection Summary" table confirms that this procedure was successful. In this case, the "Fast Selection" process finished with a final model in which 20 iterations selected accounted for 58.2% of the target variance explained (Proportion of Variance Explained = 0.581892).

Optimal Feature Mix: This step is necessary to resolve multicollinearity and to find the most efficient model. The "Selected Effects" table shows that 14 unique variables are chosen in the final set. The algorithm chose a sophisticated mix of WOE_ variables, such as WOE_duration, GRP_ variables, such as GRP_month, and original variables, such as pdays. It proves that the process successfully reduced the feature set to a small, powerful, statistically sound group.

3.3 Model Construction and Optimization

3.3.1 Bayesian Network Model (OOI DUN TZI)

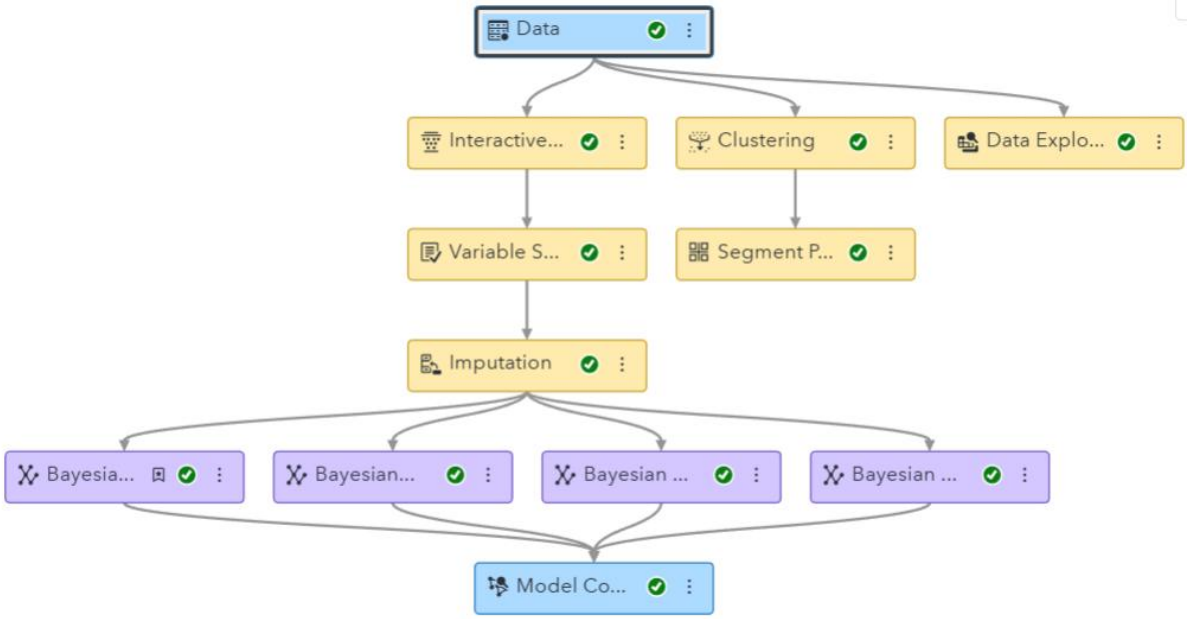

Figure 32: Bayesian Network Model Pipeline

Bayesian Network model is the models that are going to evaluate in this pipeline, and this model operates by modelling the association among variables in a directed graph with each node illustrating a variable and each relationship a probabilistic dependency. Furthermore, the model is used during training to study the conditional probabilities of each relationship using the data, and this implies that the model establishes the sensitivity of the probability of a given variable, which is dependent on the value of variables related to the given variable. Upon learning these conditional probability tables, the model is then able to make an inference using them. In case of new data, the Bayesian Network takes the learned probabilities and the evidence it has received to compute the probabilities of the various outcomes. This is because probabilistic reasoning enables the model to make predictions whenever some information is missing or uncertain hence it is effective in dealing with complex and interdependent data.

Moreover, to have a short brief of the workflow the data that has been pre-processing which is prepared and good to analyse will be fed into four parallel Bayesian Network node, and each node represents a different model configuration. Nevertheless, there will be a model comparison node to selects the best performing model among them, to clear about the model configuration, it will have 60 percent allocated for training and 40 percent reserved for validation, and the training portion will be used to allow the Bayesian Network to learn the

relationships among the variables, which also model the potential probability distribution. When the training process is completed, the model is applied to the validation portion that consists of data that the model has not encountered before, and then the predictions produced by the model are then going to be compared to the actual outcomes in the validation data. Additionally, to know how closely the predicted results match the true results is to have performance measures such as the accuracy, ROC, F1 Score, and Fit Statistics.

The pre-processing pipeline is essential for two key reasons

Data Compatibility:

In Bayesian Networks there will be required discrete variable instead of continuous values, means that the Interactive Grouping step is important by converting those continuous variables such as "euribor3m" into new "GRP_" class variables. Therefore, this will be ensured that the grouped data has strong and meaningful relationship with the target variable, and this is why making it suitable for Bayesian Network modelling.

Model Feasibility:

Too many variables will be computationally expensive and overly complex when building Bayesian Network, and this will be able to solve by using Variable Selection. In this pre-processing, it has node address by reducing the dataset to only 14 most predictive features that has been shown in 3.2.3 Variable Selection Output Table. Therefore, this simplification makes it possible to construct a stable and interpretable network structure.

3.3.2 Gradient Boosting Model (OOI DUN TZI)

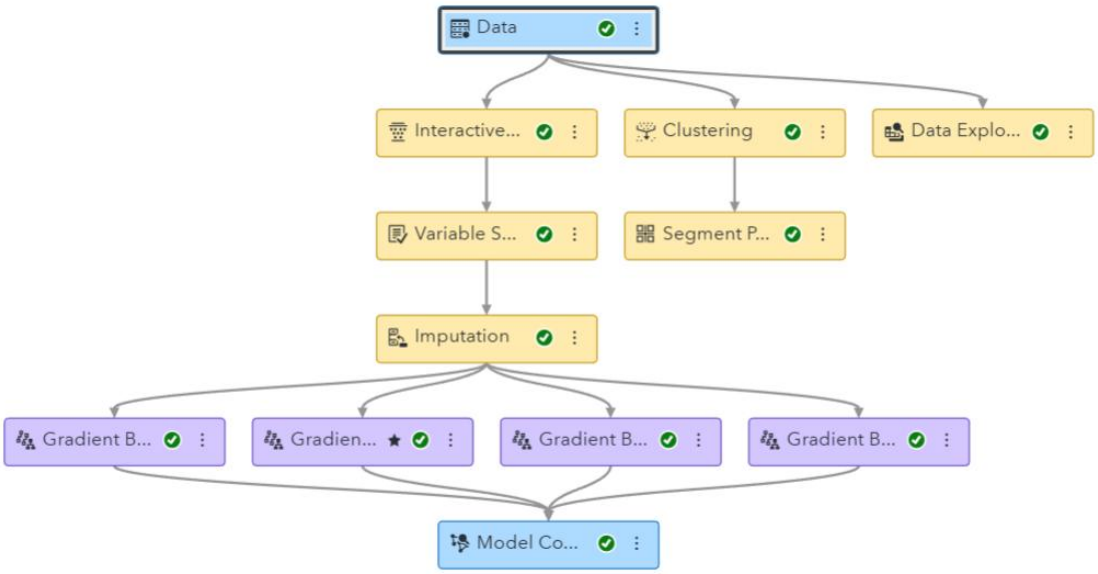

Figure 33: Gradient Boosting Model Pipeline

Gradient Boosting will be the model that is used in this pipeline, which is a highly effective ensemble learning technique. Moreover, to understand how Gradient Boosting works it will start by sequentially building a series of decision trees, where each tree will make an initial prediction. Furthermore, the model will calculate the difference between the predicted values and the actual values that are known as the residual errors. In further details, a new tree is trained specifically to predict these residuals, and this will effectively be learning the patterns of the previous model that did not capture. By this kind of iteratively continues process, each additional tree refining the model accuracy by focusing on remaining errors and then the final prediction will be obtained by combining the output of all the trees by typically through a weighted sum. In this stage, those error correcting approach enables Gradient Boosting to achieve strong predictive performance and especially in cases where relationships in the data are complex and non linear.

Similar to the 3.3.1 pipeline, the workflow sends the optimized data to four parallel Gradient Boosting nodes, each with different tuning settings. A Model Comparison node then identifies the best model from the group.

The pre-processing pipeline enhances the performance of Gradient Boosting through the following strategies

Creating Stronger Features (Interactive Grouping)

1. **WOE Transformation:**
   WOE_ the Weight of Evidence variables will be created in this stage, which will represent the predictive relationship between each of the variables and the target. Therefore, these values are particularly useful for decision trees, because there will be having a clear and monotonic signal for splitting the data.
2. **Supervised Discretization:**
   Those continuous variables are also grouped into GRP_ bins, and this will be able to stabilize the tree splitting process and capture complex patterns more effectively and improve model robustness.

Optimizing Model Performance (Variable Selection)

1. **Noise Reduction:**
   In this step it will ablet o remove those variables that provide nonmeaningful predictive signal to prevent the Gradient Boosting model from learning irrelevant patterns and help to minimize the overfitting chance.
2. **Redundancy Reduction:**
   This will be the place that remove those highly correlated or redundant variables to ensure that the model remains stable, which able the variable analyze those important results that are meaningful and interpretable.

3.3.3 Neural Network Model (YAP BOON SIONG)

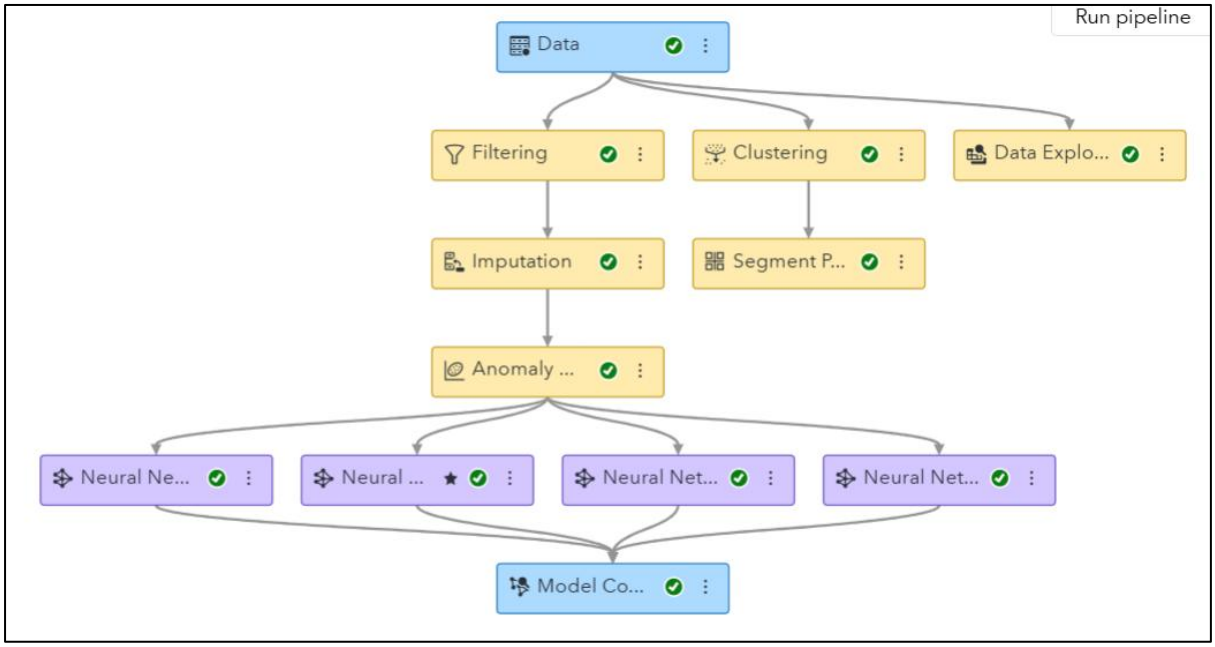

Figure 34: Neural Network Model Pipeline

This pipeline is used to test a neural network model that imitates the human brain architecture and uses layers of connected artificial neurons to identify the complex nonlinear relationships in datasets. The neural networks are especially useful at recognizing fine detail patterns that might be missed by other statistical or rule-based approaches.

The processed data in this workflow is directed to four parallel neural network nodes that share the same input data but have different hidden-layer layouts, activation functions and learning rates. Another model-comparison node is then used to calculate the model configuration with the highest predictive accuracy and least error rate.

Preprocessing phases are highly important in improving model learning and generalization:

1. **Data Normalization (standardization node)** is being carried out is due to the sensitivity of the neural networks to the scale of input; continuous data including age, balance, and duration are being standardized so that both the input feature plays a fair role in the optimization algorithm convergence as well as so that the optimization algorithm converges well.
2. **Variable Selection (feature reduction node)** to reduce overfitting as well as simplify the computational complexity the only variables selected those that are most salient discovered in the prior selection phase, concerning predictive power of the variable

collection are then used, ultimately allowing the model to focus on variables with the highest predictive value, thus enhancing both accuracy and interpretability.

3. **Network optimization** is an iterative technique that hypersets hyperparameters (the number of hidden layers, the count of neurons at each layer and the learning rate) and evaluates the model effectiveness by utilizing a validation dataset to minimize mean-squared error (MSE) and maximize classification accuracy.

This optimization procedure ensures that the neural network reaches a middle ground between predictive model and generalization making it applicable in capturing the nonlinear customer subscription behavior dependencies.

3.3.4 Forest Model (YAP BOON SIONG)

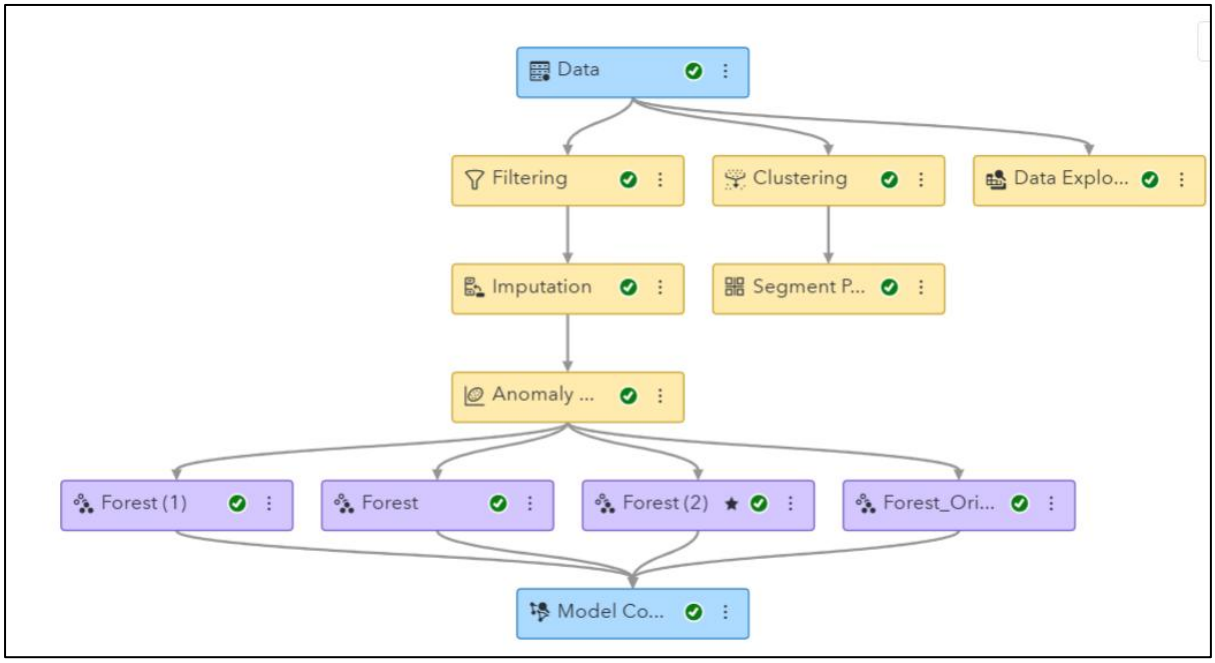

Figure 35: Forest Model Pipeline

The forest model which is an ensemble technique that builds many decision trees and combines their predictions to provide better accuracy and stability are also tested in this pipeline. The forest model is implemented in SAS Viya where the forest model is a collection of decision trees which are trained on random subsets of data and variables, reducing prediction bias and improving prediction ability.

The workflow that is going to be used is a four parallel forest node with each node having a specific number of trees, depth, and leaf size. A model-comparison node is next and this node determines the best configuration in terms of misclassification rate and mean square error.

Preprocessing and optimization schemes are developed in order to enhance the stability and interpretability of the models:

1. **Feature Engineering (interactive grouping and Weight of Evidence transformation)** is used to continuous variables are discretized into bins of groups (GRP variables) and categorical variables are transformed into WOE form. This transformation aligns the distributions of the variables with the target in the process increasing the ability of the model to identify significant splits across the trees.

2. **Variable Selection (Noise and Redundancy Reduction)** eliminates the irrelevant or highly correlated variables which guarantees every tree provides distinct predictive content that consequently results in greater efficiency and reduces computational cost.
3. **Model Optimization (hyperparameter tuning)** include the number of trees, maximum depth and minimum leaf size. The number of trees increases the number of computations, although accuracy usually increases with the number of trees. The last model makes a compromise with a configuration configuration that gives high precision with a minimum amount of overfitting, as seen through a smaller mean square error on an average basis.

Overall, the forest model provides a highly effective and decipherable ensemble of frameworks that complements the neural network model by additionally showing superior predictive capabilities in addition to offering higher insights into decision courses.

3.3.5 Decision Tree (CHEAH JUN HENG)

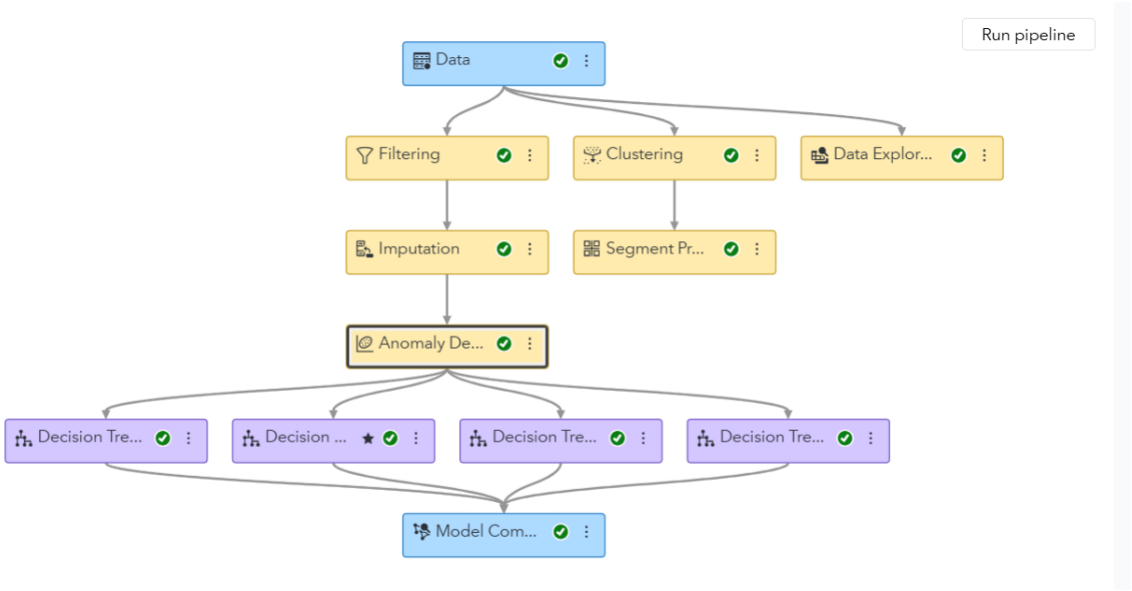

Figure 36: Decision Tree Pipeline

Pipeline Overview

The analysis workflow is applied in sequence to guarantee robust and accurate model results. It starts with Data Preprocessing (Filtering, Imputation and Anomaly Detection) to clean and prepare it. This is succeeded by Exploratory Data Analysis and Clustering, to study data patterns and cluster the customer base. Pre-processed data is input to several Decision Tree models whose performances are evaluated by the Model Comparison node to choose which predictive model will be deployed.

This Decision Tree is a predictive modelling tool that uses a flowchart as a decision-making device. It asks some questions about the input data ("Is the duration of this call more than 5 minutes?"). Each answer triggers a new branch and question until finally reaching a prediction (e.g., "client will subscribe"). Its main strengths are simplicity, interpretability and capacity to manage both numeric and categorical data.

Justification for Preprocessing in Relation to the Decision Tree Model

The three preprocessing steps are critical for building an effective and reliable Decision Tree.

1. Filtering: Decision Trees can easily be overfit by creating specific branches for rare categories. By removing infrequent categories (e.g., the 'illiterate' education level), filtering prevents the tree from learning noise and overly complex rules that do not generalize well to new data, leading to a more robust model.

2. Imputation: While no missing values were found in this dataset, the imputation step is a vital safeguard. Decision Trees cannot handle missing values directly, and their presence would have caused errors during training. This step ensures the model will always receive a complete dataset.

3. Anomaly Detection: Outliers can disproportionately influence the splitting decisions at the root of a Decision Tree, skewing the entire model structure. By identifying and excluding these anomalous points, we ensure the tree learns the central patterns of the data, resulting in more stable and accurate predictions for most cases.

3.3.6 Logistic Regression (CHEAH JUN HENG)

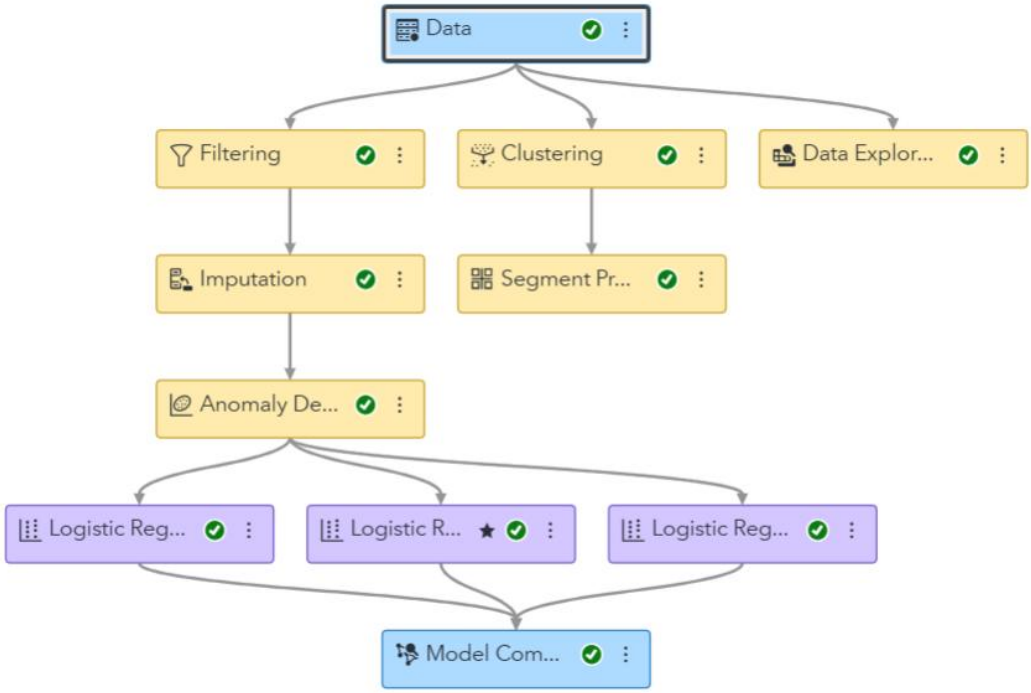

Figure 37: Logistic Regression Pipeline

Pipeline Overview

The analysis is guided by a rigorous methodology to produce well-founded model results. Data Preprocessing (Filtering, Imputation, Anomaly Detection) The cleaning and preparation of the data starts here. The transformed data is then entered into several Logistic Regression models, which are compared against each the other in the Model Comparison node to determine which predictive model should be deployed.

This is a Logistic Regression model, where predicted value is binary ('yes' for customer subscribed and 'no' for not subscribed). It's different from a Decision Tree, where it predicts the subscription probability by combining all input features linearly. For this campaign we calculate the impact resulting from variables such as call duration, employment rate (nr. position), and month jointly affect the probability of customer acquisition. Its great advantage is that it makes a clear classification but also measures the exact impact and significance of each variable on outcome, rendering a very interpretable for business strategy model.

Justification for Preprocessing in Relation to the Logistic Regression Model

The three preprocessing steps are critical for building an effective and reliable Logistic Regression model.

1. Filtering: Logistic Regression requires a well-represented dataset for stable coefficient estimation. By removing infrequent categories, filtering prevents the model from calculating unreliable and volatile estimates for rare groups, ensuring that the inferred relationships between variables and the outcome are robust and generalizable.

2. Imputation: Logistic Regression cannot process observations with missing values. While no missing values were found in this dataset, the imputation step is a mandatory safeguard. It guarantees data completeness, preventing the algorithm from failing and ensuring the entire dataset is used for training.

3. Anomaly Detection: Outliers can severely distort the estimated coefficients in a Logistic Regression model, pulling the "line of best fit" in misleading directions and reducing predictive accuracy. By identifying and excluding these anomalous points, the model captures the true underlying relationships between the input variables and the subscription probability, leading to more reliable predictions.

Summary

Finally, doing the same preprocessing guarantees a fair comparison when evaluating the model since both algorithms were fed with a high-quality dataset and that model's performance could be assessed based on their ability to predict information rather than how well they prepared data. Decision Tree and Logistic Regression models were developed in both tasks to predict customer subscription. The same preprocessing steps (Filtering, Imputation and Anomaly Detection) were composed for both modelling methods as they deal with basic data quality concerns that are not dependant on the model. Filtering preserves stability by eliminating low prevalence categories that cause overfitting in Decision Trees and unstable coefficients in Logistic Regression. Imputation ensures the presence of a full dataset needed in two methods' mathematical bases. anomaly detection gets rid of serious outliers that can mess up spots the data is being split in a decision tree or screw with coefficient estimates in logistic regression.

3.4 Model Validation

3.4.1 Bayesian Network Model (OOI DUN TZI)

3.4.1.1 Modify & Origin Tunning Explanation

Table 1: Bayesian Network Origin & Modify Tunning

Parameter Category	Parameter	Bayesian Network Origin Tuning	Bayesian Network Modify Tuning
Input Options	Missing class inputs	Exclude	Exclude
	Missing interval inputs	Exclude	Exclude
	Number of bins	20	37
Variable Selection Options	Prescreen variables	Checked	Checked
	Variable selection	Unchecked	Unchecked
	Independence test statistics	G-Square	G-Square
	Significance level	0.2	0.2
Network Structure Options	Network structure	Naive (Checked), Tree-Augmented (Checked), Parent-Child (Checked), Markov blanket (Unchecked)	Naive (Checked), Tree-Augmented (Checked), Parent-Child (Checked), Markov blanket (Unchecked)
	Maximum parents	4	4
	Choose best number of parents	Checked	Checked
	Parenting method	Set of parents	Set of parents

In this table it shows that the Bayesian Network Modify Tuning mainly differences are the number of bins, which is increasing from 20 to 37. Therefore, this enables us to improve how the continuous variables are represented and helps the model capture those finer data patterns. Additionally, in these kinds of changes it will lead to higher accuracy because it enhances the estimation of conditional probabilities between the variables. However, other parameters are

tuned differently between Bayesian Network Origin Tuning before, and the accuracy result remains the same as the way tuned in the table above, this will be suggesting the model has likely reached its optimal performance. Which means that the current setup already provides the best balance between accuracy, interpretability, and efficiency. As a result, further adjustments no longer bring noticeable improvements.

3.4.1.2 Modify Model & Origin Model Result Comparison

Model Comparison

Champion	Name	Algorithm Name	Accuracy	Average Squared Error
true	Bayesian Network (4)	Bayesian Network	0.8521	0.1166
false	Bayesian Network_Origin	Bayesian Network	0.7427	0.1672
false	Bayesian Network (5)	Bayesian Network	0.7716	0.1596
false	Bayesian Network (6)	Bayesian Network	0.8257	0.1311

Figure 38: Model Comparison Bayesian Network

The “Model Comparison” table above has shown the champion model which is the “Bayesian Network (4)” by indicated “true” value in its corresponding row. As a result, the champion model shows a clear improvement over “Bayesian Network_Origin” in a way of champion model achieves higher accuracy of 0.8521 compared to 0.7427 for the original model. In addition, the champion model records a lower Average Square Error 0.1166 than the original model 0.1672, this means that its predictions are more precise and have less overall error.

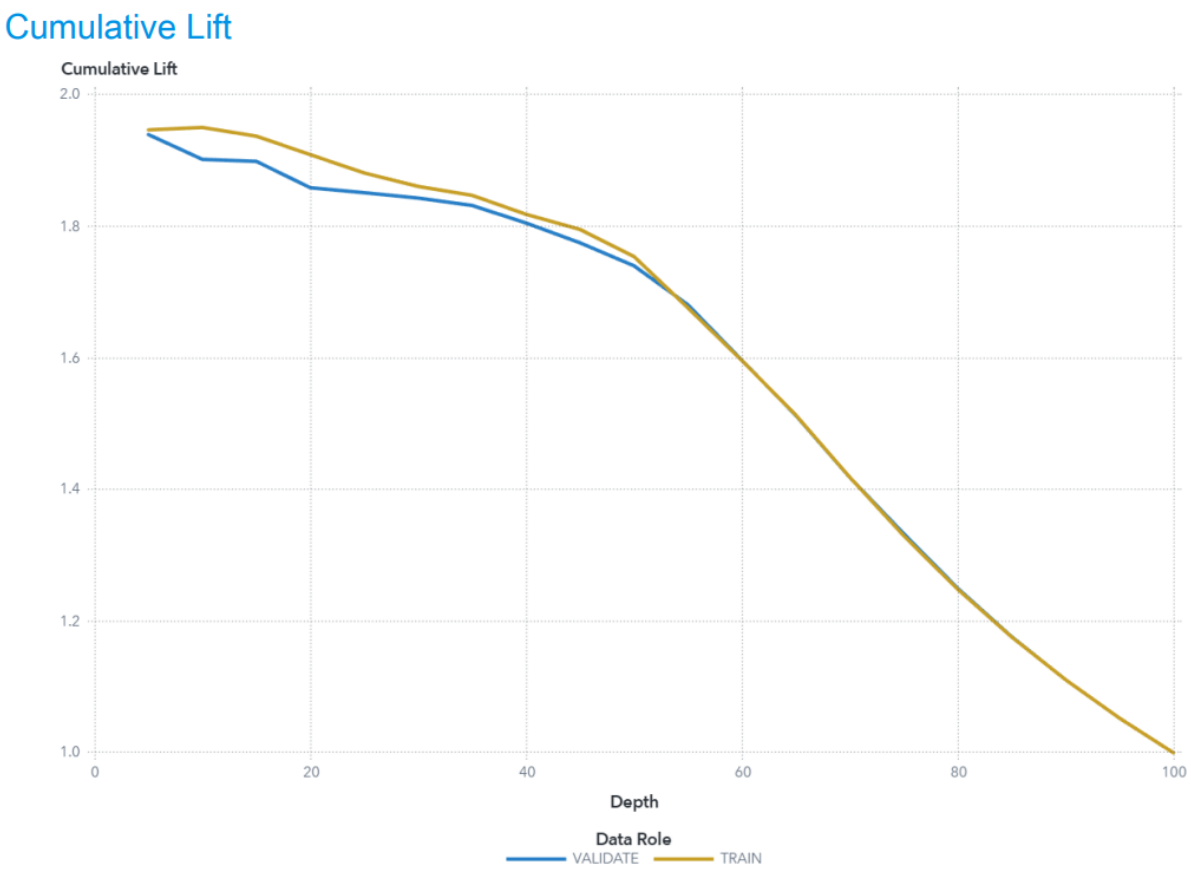

Figure 39: Cumulative Lift Bayesian Network

The cumulative lift graph helps to show how much the model performs when finding positive outcomes compared to random guessing. In the validation partition, the cumulative lift is 1.9 at the 10 percent depth, this means that the top 10 percent of the records contain about 1.9 times more positive events than would be expected by random selection. In other hand, the training partition cumulative lift is 1.95 at the same depth, which showing that the top 10 percent of ranked observation include 1.95 time more events than random.

In this case, both values are greater than one and this indicates that using this model is much better than making a random choice when identifying responders, and the data is sorted by predicted probability of "yes" divided into 20 groups of equal size. Then it will compare against what would be expected by chance and based on this measure it shows how effective the model is at spotting true positives compared to a random selection.

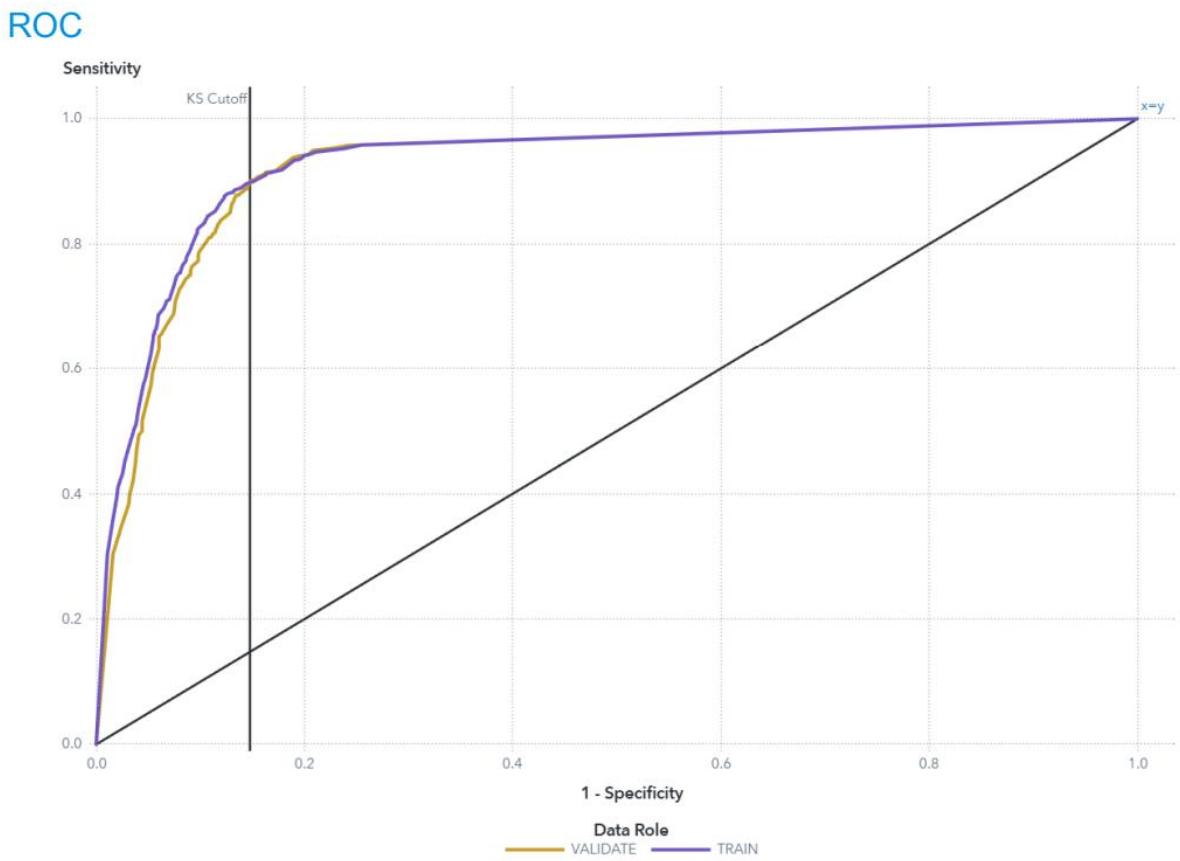

Figure 40: ROC Bayesian Network

The ROC graph demonstrates the relationship between the true positive rate and false positive rate for the different cutoff values, and this may help to evaluate how well the model can be identify between positive and negative outcomes.

Based on this model, the KS cutoff line for the validation partition is drawn at a threshold of 0.18, where the false positive rate is 0.148 and the true positive rate is 0.901, and this means that at this point about 90.1 percent of actual positive cases are correctly identified while 14.8 percent of negative cases are wrongly predicted as positive.

The ROC curve that rapidly approaches the upper left corner and shows that higher accuracy, then a diagonal line will be represented by random predictions. Since the ROC curve here leans strongly towards the upper left corner, the model shows good classification performance and strong discrimination ability.

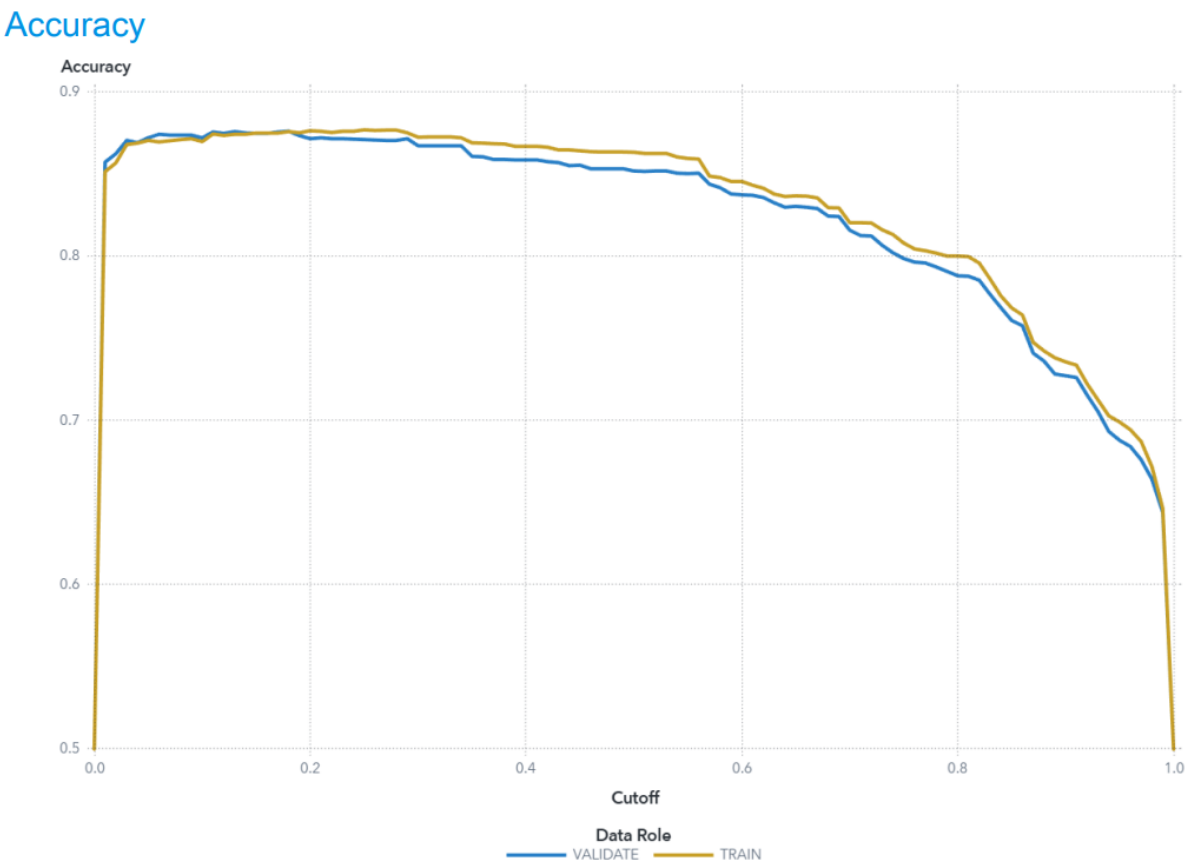

Figure 41: Accuracy Bayesian Network

The accuracy graph shows the percentage of the total cases that the model correctly classifies as either positive or negative, and this model has achieved the accuracy for the training partition at a cutoff of 0.5 is 0.864, and for the validation partition is 0.852.

This shows that the model correctly predicted about 86.4 percent of training data and 85.2 percent of validation data, and that accuracy is measured by checking how many predictions match the actual outcomes across different cutoff points. Moreover, the small difference between training and validation accuracy will show that the model maintains consistent and dependent performance on new unseen data.

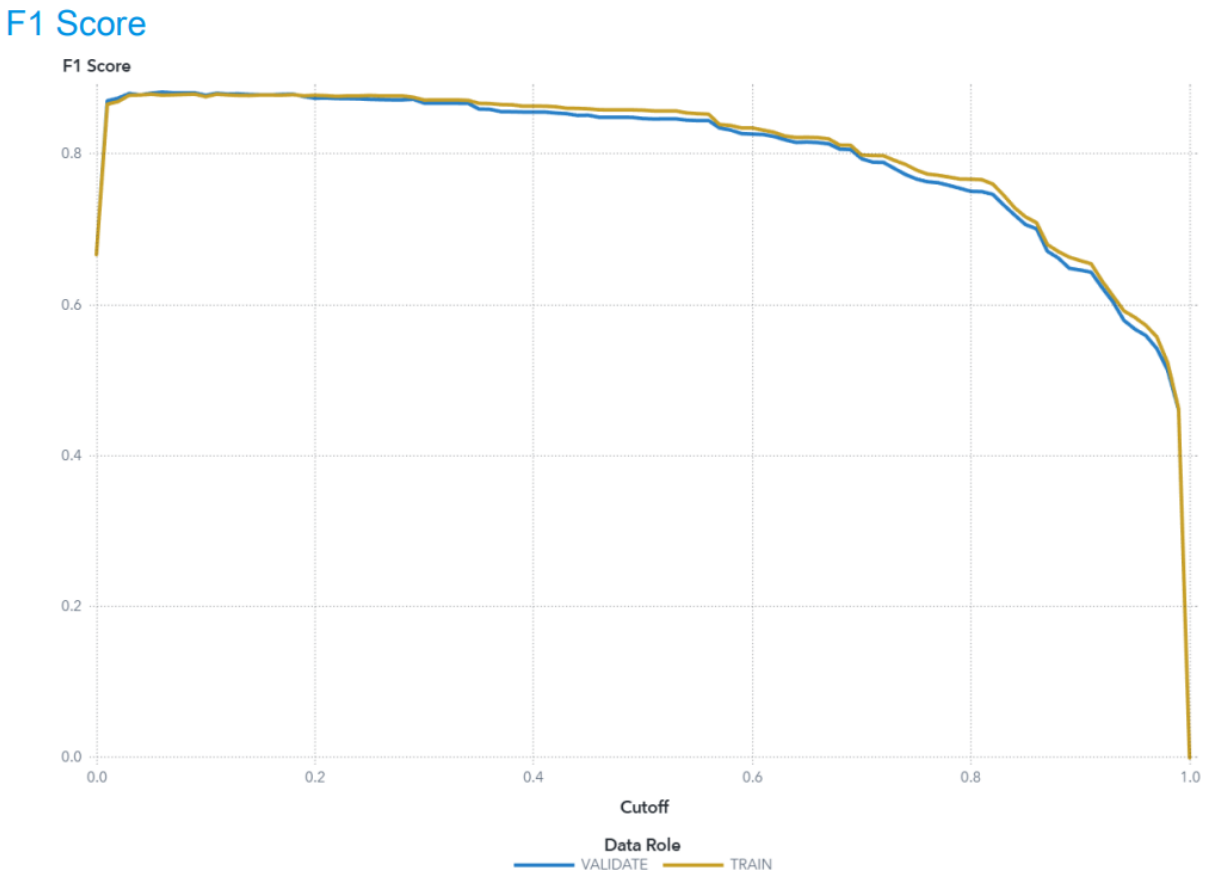

Figure 42: F1 Score Bayesian Network

F1 graph is used to indicate the precision and recall of the model. Precision is the number of the successful cases of the model that are actually positive, and recall is the number of actual positive cases that the model actually detects.

In this model, the F1 score with training partition is 0.858 and the F1 score with the validation partition is 0.847, both at the cutoff of 0.5. These scores reflect that the model is robust and steady and that there is only slight decrease in the validation scores. The large F1 scores indicate that the model can be successfully used to detect true positives and the false positives are rather small.

Fit Statistics

Target Name	Data Role	Partition Indicator	Formatted Partition	Number of Observations
y	TRAIN	1	1	5,568
y	VALIDATE	0	0	3,712

Average Squared Error	Divisor for ASE	Root Average Squared Error	Misclassification Rate	Multi-Class Log Loss
0.1115	5,568	0.3339	0.1365	0.4403
0.1166	3,712	0.3415	0.1479	0.4653

KS (Youden)	Area Under ROC	Gini Coefficient	Gamma	Tau
0.7543	0.9277	0.8555	0.8883	0.4278
0.7527	0.9216	0.8433	0.8782	0.4217

KS Cutoff	KS at User-Specified Cutoff	Misclassification Rate at KS Cutoff (Event)	Misclassification Rate (Event)
0.2500	0.7270	0.1228	0.1365
0.1800	0.7042	0.1237	0.1479

Figure 43: Fit Statistics Bayesian Network

The Fit Statistic Table gives a perfect report of the model in both the training and validation dataset. It emphasizes the quality, consistency, and precision of the model according to a variety of important indicators.

The training and validation values of the Average Squared Error are 0.1115 and 0.1166, respectively indicating that there are not too big errors of prediction. The Misclassification Rate of training and validation is 0.1365 and 0.1479 respectively, which implies that there was a minimal fraction of the records that were wrongly classified. The Log Losses of training and validation are 0.4403 and 0.4653 respectively, and prove that the predicted values are in close contact with the actual ones.

Training and validation Area Under the ROC Curve (AUC) values are 0.9277 and 0.9216 respectively, which shows there is a good level of discrimination. The values of Gini Coefficient of the training and validation 0.8555 and 0.8433 respectively confirm that the model is very well performing.

All in all, these validation metrics are quite high despite them being slightly less than the training results, which is not surprising. This trend indicates that the model is generalizable and works reliably on unobserved data and does not have a high overfitting issue.

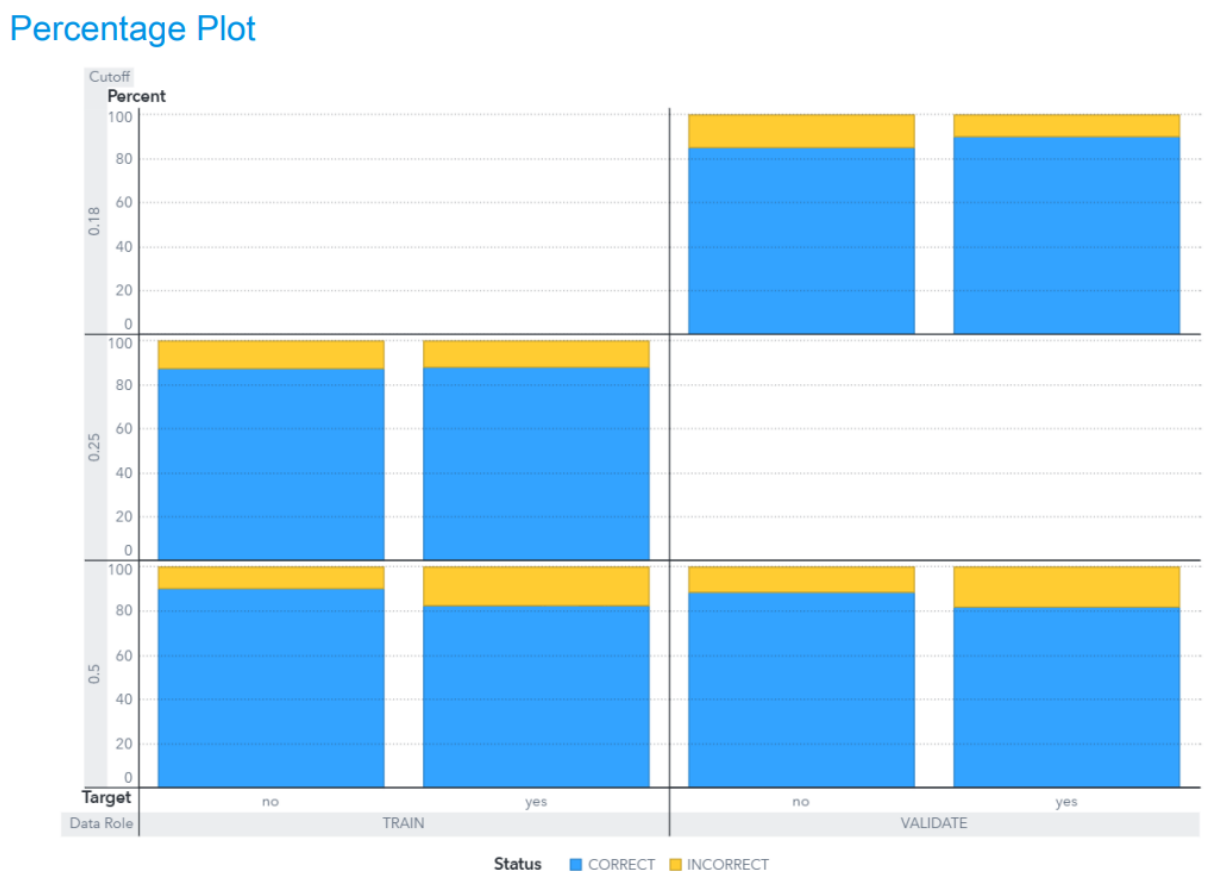

Figure 44: Percentage Plot Bayesian Network

The percentage plot provides a graphical overview of the model in terms of its ability to classify the events by the various cutoff points. It presents the confusion table in percentages, indicating the proportion of cases that the prediction was correct and those that it was incorrect.

The model is compared to two cutoff values, which are the default cutoff of 0.5 and the KS cutoffs of 0.25 in for the training partition and the validation partition is 0.18. The part labeled "CORRECT" represents the fraction of true positives at the level of event represented by the bar labeled the "yes". Lastly, this plot allows to quickly see how accurately the model identifies actual positive cases and how that performance changes at different thresholds.

Summary

The Bayesian Network model has good and stable predictive performance, and this mean that it is a champion model as compared to the original Bayesian Network_Origin. It has a higher accuracy of 0.8521 than the original model which has an accuracy of 0.7427, and its Average Squared Error is lower at 0.1166 than the original model of 0.1672, indicating that the champion model can be used to make more accurate and reliable predictions.

The confirmed cumulative lift results indicate that the model finds positive outcomes nearly twice as well as the random selection and the values of lift are 1.95 of the training data and 1.9 of the validation data. The values of ROC and AUC, 0.9277 and 0.9216 respectively in training and validation respectively indicate that the model is highly discriminative over positive and negative cases.

The values of accuracy 0.864, training, and 0.852, validation suggest that the model is consistently effective on both datasets. The values of F1, 0.858, training, and 0.847, training, and validation, respectively. The validation measures are also near the training scores implying that the level of overfitting is mild.

All in all, Bayesian Network model has high predictive reliability, it generalizes better with unknown data and it is evidently better than its counterpart. It is arguable that it is a stable, accurate and effective forecasting model.

3.4.2 Gradient Boosting Model (OOI DUN TZI)

3.4.2.1 Modify & Origin Tunning Explanation

Table 2: Gradient Boosting Modify & Origin Tuning

Parameter Category	Parameter	Gradient Boosting Origin Tuning	Gradient Boosting Modify Tuning
Basic Options	Number of trees	100	500
	Learning rate	0.1	0.05
	Subsample rate	0.5	0.8
	L1 regularization	0	0.3
	L2 regularization	1	3
	Interval target distribution	Normal	Normal
	Seed	12,345	12,345
Tree-splitting Options	Maximum number of branches	2	4
	Maximum depth	4	6
	Minimum leaf size	5	30
	Missing values	Use in search	Use in search
	Number of interval bins	50	128
	Interval bin method	Quantile	Quantile
Early Stopping Options	Maximum class levels	128	12
	Class target metric	Misclassification rate	Log loss
	Early stopping method	Stagnation	Stagnation
	Stagnation	5	8
	Tolerance	0	0.0001

The initial Gradient Boosting model was a 100 tree model with 0.1 learning rate, which can learn faster (and less accurately). The adjusted setting set the trees to 500 and the learning rate to 0.05 to make the learning slow and more accurate. L1 regularization (0.3 -0.3) and L2 (1-3) elevated overfitting. The tree-structure was increased in terms of the number of branches and depth by 2 and 4 respectively and the smallest splits of the leaf size were increased by 5 to 30 to avoid over-specific splits. Both models were based on a Normal target distribution, the same seed, and Quantile binning, only the number of interval bins was raised to 50, 128. The initial model employed the misclassification rate in early stopping after 5 stagnation round, whereas the new one employed the log loss with 8 round and a tolerance of 0.0001. All in all, the modified tuning is more accurate and generalized whereas the original is focused on speed and simplicity.

3.4.2.2 Modify Model & Origin Model Result Comparison

Model Comparison

Champion	Name	Algorithm Name	Accuracy	Average Squared Error
true	Gradient Boosting (1)	Gradient Boosting	0.8071	0.1395
false	Gradient Boosting_Origin	Gradient Boosting	0.7769	0.1549
false	Gradient Boosting (2)	Gradient Boosting	0.7982	0.1418
false	Gradient Boosting (3)	Gradient Boosting	0.7934	0.1440

Figure 45: Model Comparison Gradient Boosting

According to the table of the Model Comparison, the champion model is Gradient Boosting (1) because the value of true in the row of this model is true as can be seen in the table. The improvement of this champion model over the Gradient Boosting_Origin is evident. As a matter of fact, the champion model has a better accuracy of 0.8071 than the original model of 0.7769. Moreover, its Average Squared Error of 0.1395 is a lower number than the original model which is 0.1549, which means that it makes more accurate predictions, and that the total error is smaller.

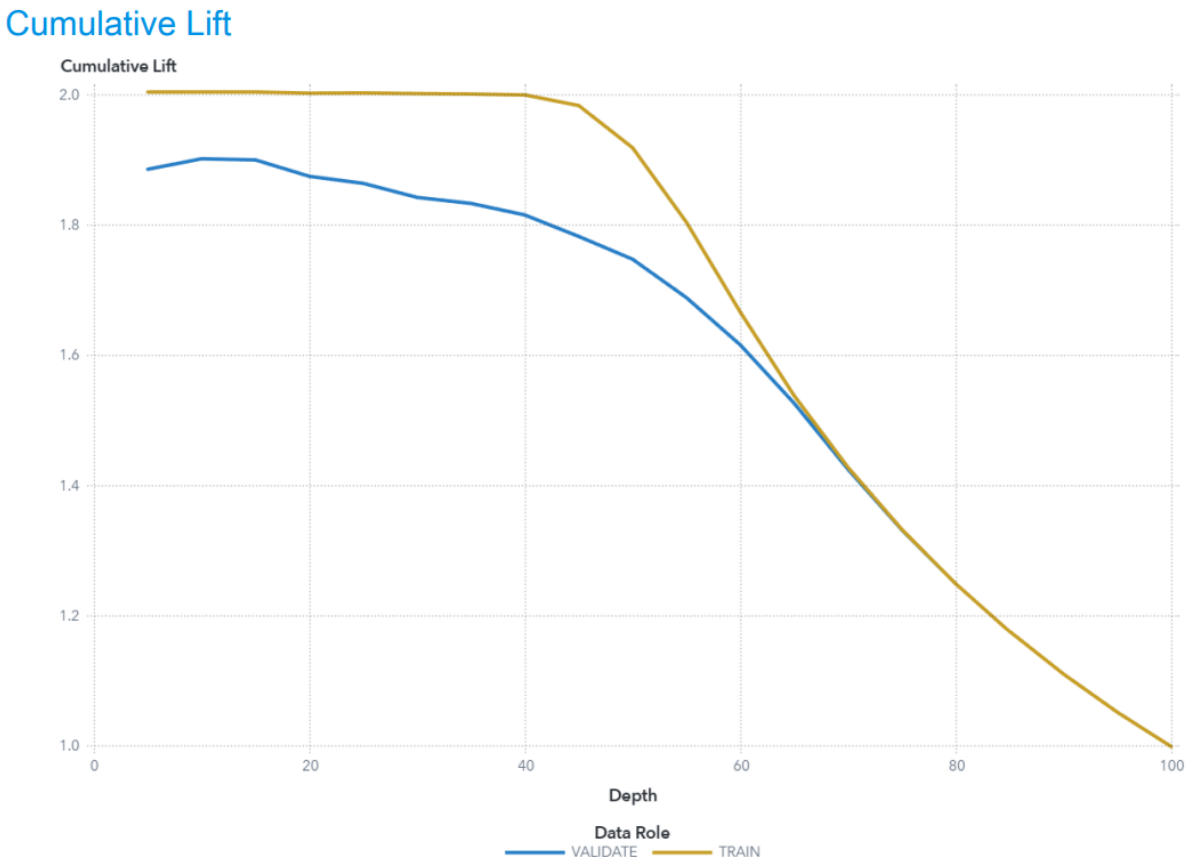

Figure 46: Cumulative Lift Gradient Boosting

The Cumulative Lift Graph is used to consider the effectiveness of the predictive model in distinguishing between positive cases, for example the respondents who answered “yes” and random selection. It is the measure of improvement of the model relative to random guessing as a baseline.

In this analysis, the cumulative lift of the validation partition is 1.9 at the 10 percentiles whereas the cumulative lift of the training one stands at 2.0. This implies that in the top 10 percent of observations of predicted probability, there are about 1.9 times (to validate) and 2.0 times (to train) more positive responses than would be the case with only chance.

Cumulative lift is computed by first order ranking the data (by the predicted likelihood of the event of interest, (P_yyes). The data is sorted into equal quantiles and the cumulative lift of each quantile is calculated as the ratio of the number of positive responses achieved up until that quantile is reached of the total quantile to the expected number of positive responses under random selection.

The cumulative lift value of more than one suggests that the model is better than random selection to detect prospective positive responses. The increase in cumulative lift curve will mean that the model will rank the responders that are most likely to occur in a better way.

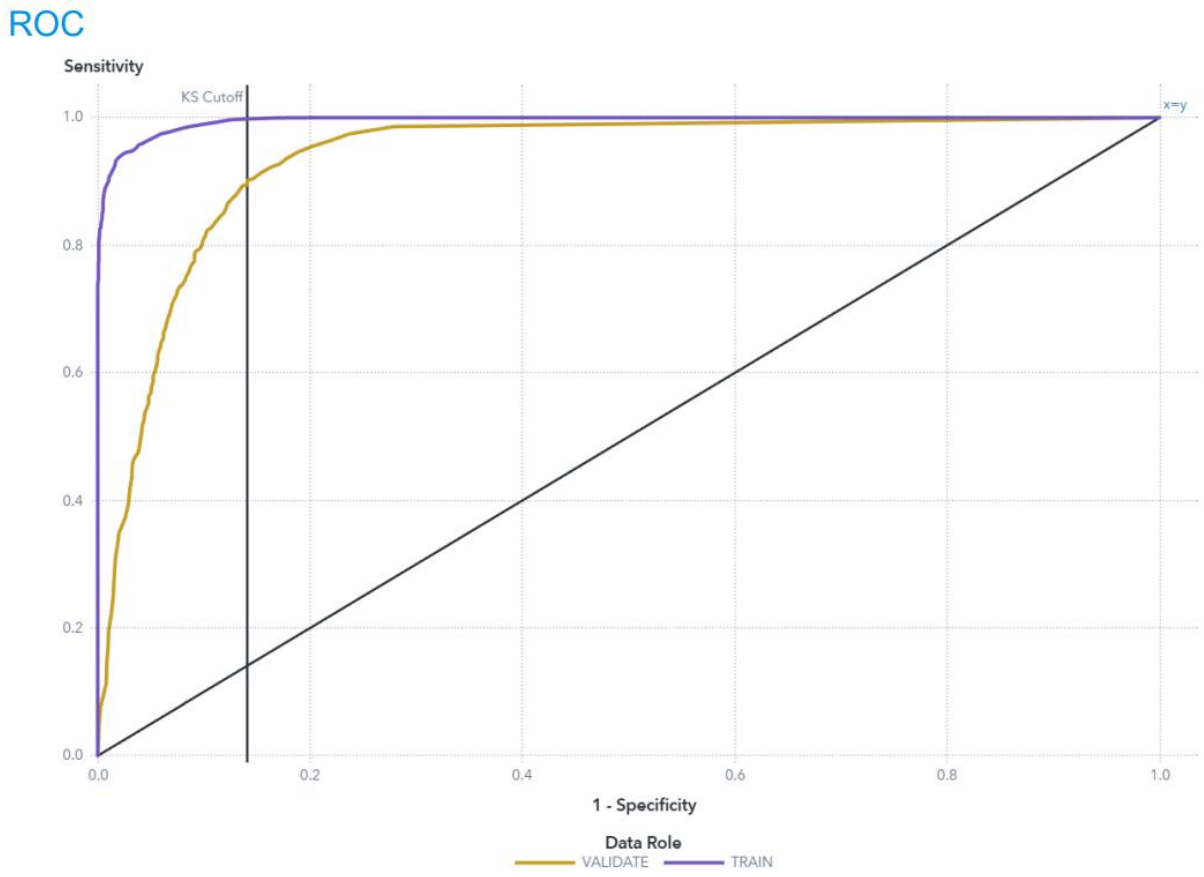

Figure 47: ROC Gradient Boosting

A graph (ROC Graph) shows the trade off between the sensitivity of the model (true positive rate) and 1-specificity (false positive rate) for a series of cutoff thresholds. It gives a complete picture of classification performance since it reveals a variation of the proportion of the correctly identified positives as the decision threshold varies.

In this model, the KS (Kolmogorov-Smirnov) cutoff is 0.12, 1-specificity =, 0.141 and sensitivity = 0.901. It means that at this level, it is detected that about 90.1 percent of real positive cases are recognized and 14.1 percent of the negative cases are false-identified as positive.

A ROC curve that tends to reach the upper-left of the plot means that the classification accuracy will be high. A random classifier with sensitivity =1-specificity is a diagonal line in the graph. The hysteresis of the ROC curve (AUC) measures the total discriminating capability of the

model. The AUC of the validation partition of 0.9356 in this instance indicates that there is an excellent capability of differentiating positive and negative cases.

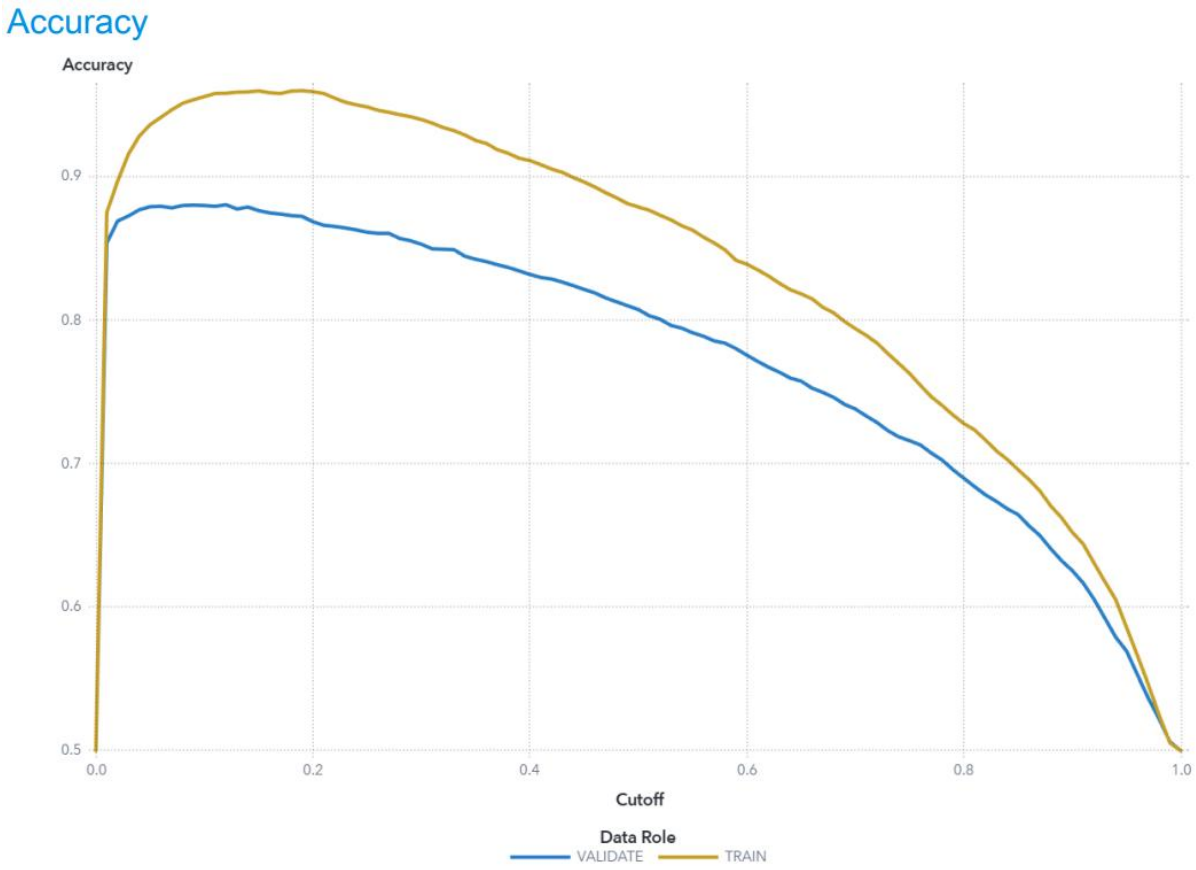

Figure 48: Accuracy Gradient Boosting

Accuracy Graph will show the percentage of all the correct or incorrect predictions, either positive or negative. It demonstrates the accuracy variation with the change of decision threshold.

Here the model was found to be accurate according to the training part of the population of 0.879 and the validation part of the population of 0.807 at the default cutoff point of 0.5. This implies that the model was able to classify the training data and validation data with an average of 87.9 percent and 80.7 percent respectively.

Even though there is a minor decrease in accuracy on transition to validation data, the difference is not very high. This indicates that the model is opportunistic in generalizing to unobservable data and it has reasonable predictive characteristics.

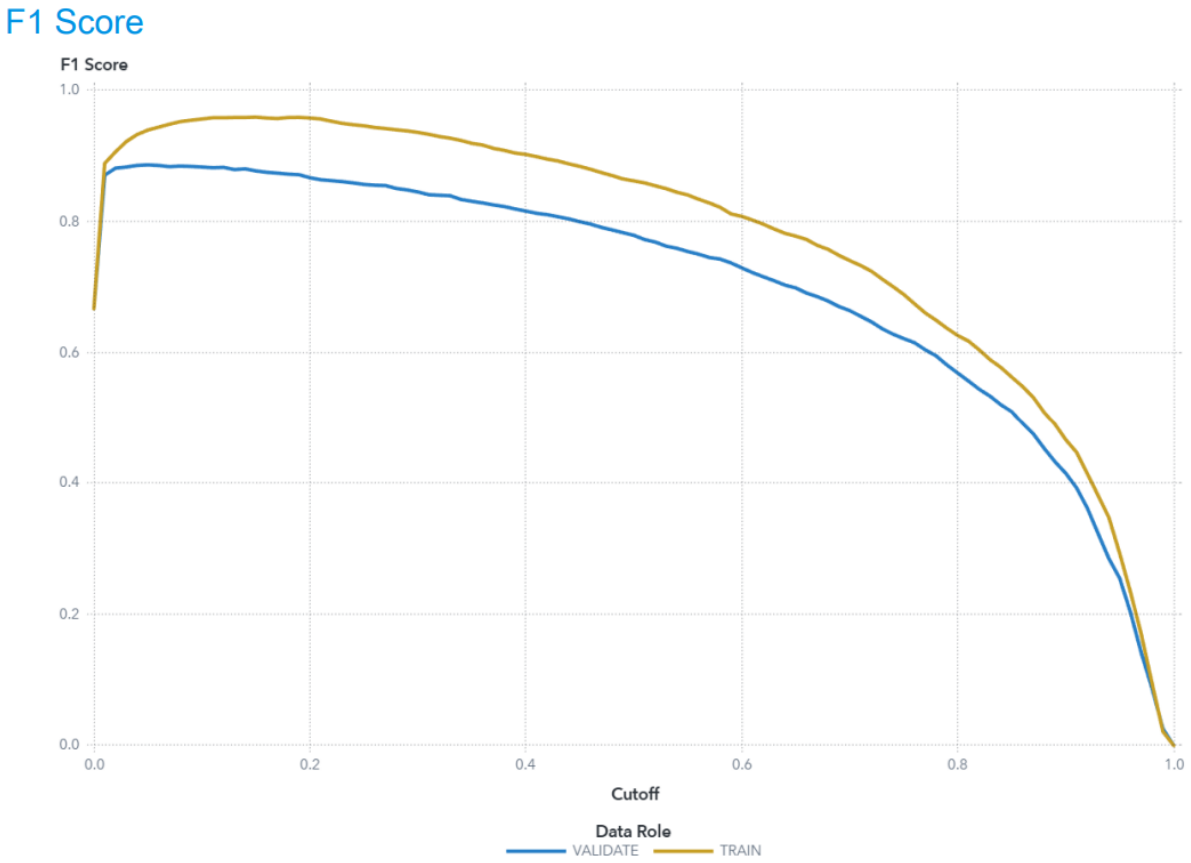

Figure 49: F1 Score Gradient Boosting

The F1 Score Graph gives a value that containing a mixture of both precision and recall. Precision is the ratio of the number of predicted positive cases to the actual positive cases, and in other hand the recall is the ratio of the actual positive cases to the predicted positive cases. As a result, to have a general impression of the model performance, the F1 score balances these two.

In this model, the F1 score of the training partition is 0.862 and that of the validation partition is 0.779 both at a cutoff value of 0.5. The larger the F1 score, the more desirable the balance between a positive result being identified correctly and low false positive.

The validation partition has a slightly low F1 score, which is anticipated, which means that the model is slightly less effective with unseen data. However, the outcome is good and indicates sound classification abilities.

Fit Statistics

Target Name	Data Role	Partition Indicator	Formatted Partition	Number of Observations
y	TRAIN	1	1	5,568
y	VALIDATE	0	0	3,712

Average Squared Error	Divisor for ASE	Root Average Squared Error	Misclassification Rate	Multi-Class Log Loss
0.0834	5,568	0.2888	0.1214	0.2571
0.1395	3,712	0.3735	0.1929	0.4521

KS (Youden)	Area Under ROC	Gini Coefficient	Gamma	Tau
0.9192	0.9951	0.9902	0.9908	0.4952
0.7602	0.9356	0.8713	0.8825	0.4358

KS Cutoff	KS at User-Specified Cutoff	Misclassification Rate at KS Cutoff (Event)	Misclassification Rate (Event)
0.1900	0.7572	0.0404	0.1214
0.1200	0.6142	0.1199	0.1929

Figure 50: Fit Statistics Gradient Boosting

The Fit Statistic Table is used to clearly show the performance of the model both on the training and the validation dataset. It also shows some of the crucial indicators used to gauge the accuracy, predictive capacity and reliability of the model.

The Average Squared Error (ASE) of training data is 0.0834 and that of validation data is 0.1395, indicating that the model does not give large errors in making predictions. The Misclassification Rate of training and validation is 0.1214 and 0.1929 respectively, and it implies that very few predictions were not classified correctly. The training and validation values of Log Loss of 0.2571 and 0.4521 indicate that the actual values are very near to the predicted values.

The Action Under the ROC Curve (AUC) of 0.9951 and 0.9356 of training and validation respectively indicate that the model possesses a great capacity to differentiate between positive and negative cases. The Gini Coefficient of 0.9902 on training and 0.8713 on validation also testifies to the good performance of the model.

In general, the validation measures are high, although a little bit lower than the training ones. This insignificant disparity is natural and indicates that the model is highly predictive and generalizes to new, unknown data.

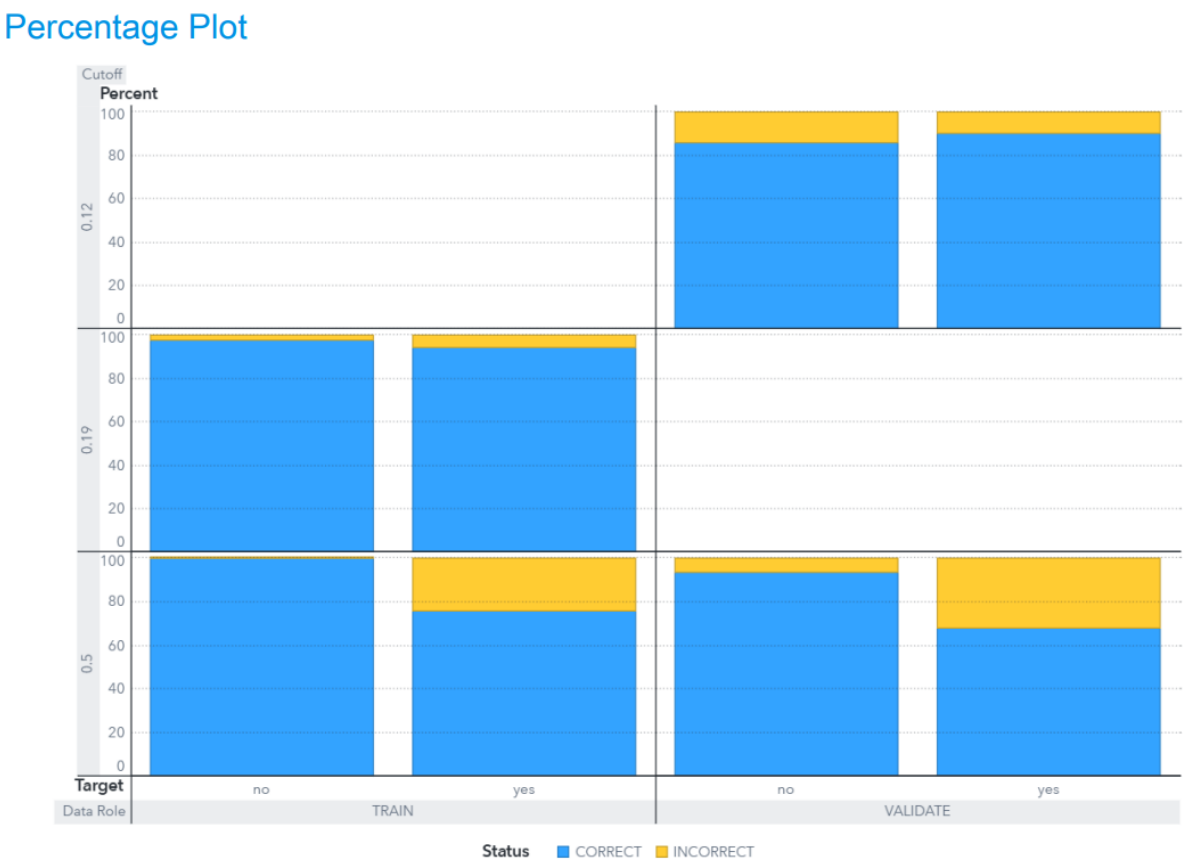

Figure 51: Percentage Plot Gradient Boosting

Percentage Plot picks the confusion matrix at percentages of the cutoff value selected. It indicates the percentage of both correct and incorrectly observed data of the training and the validation partitions.

In the plot, the bars are the distribution of the predicted classes and the section labelled as CORRECT is the percentage of observations which the model correctly predicted. This report contains cutoff values of the default (0.5) and the cutoff values of the KS (0.19 during the training and 0.12 during validation).

The graph gives a good snapshot of the model effectiveness in distinguishing between positive and negative classes in particular thresholds as well as it can determine the best cutoff point of the model in classification.

Summary

Gradient Boosting model exhibits good and consistent predictive power which is higher than the initial Gradient Boosting_Origin model. It has a greater accuracy of 0.8071 than 0.7769 and a lesser Average Squared Error of 0.1395 than 0.1549 meaning that its estimates are more precise and consistent.

The results of cumulative lift indicate that the model detects positive cases nearly twice as well as random selection with lift values of 2.0 and 1.9 on the training and validation data respectively. The value of 0.9951 and 0.9356 of ROC curve and AUC during training and validation, respectively, validate great discrimination of positive and negative classes.

The model has a training and validation accuracy of 0.879 and F1 score of 0.862 and 0.779, respectively. These findings demonstrate that the performance is steady between the two datasets and there is a minor decrease in the validation split. The small difference indicates a slightly high degree of overfitting, which is typical of well-performing predictive models.

All in all, Gradient Boosting model is highly generalist in terms of unknown data, highly predictive and far much better than the initial one. It can be deemed as a consistent, precise and efficient paradigm of sound forecasting.

3.4.3 Neural Network Model (YAP BOON SIONG)

3.4.3.1 Modify & Origin Tunning Explanation

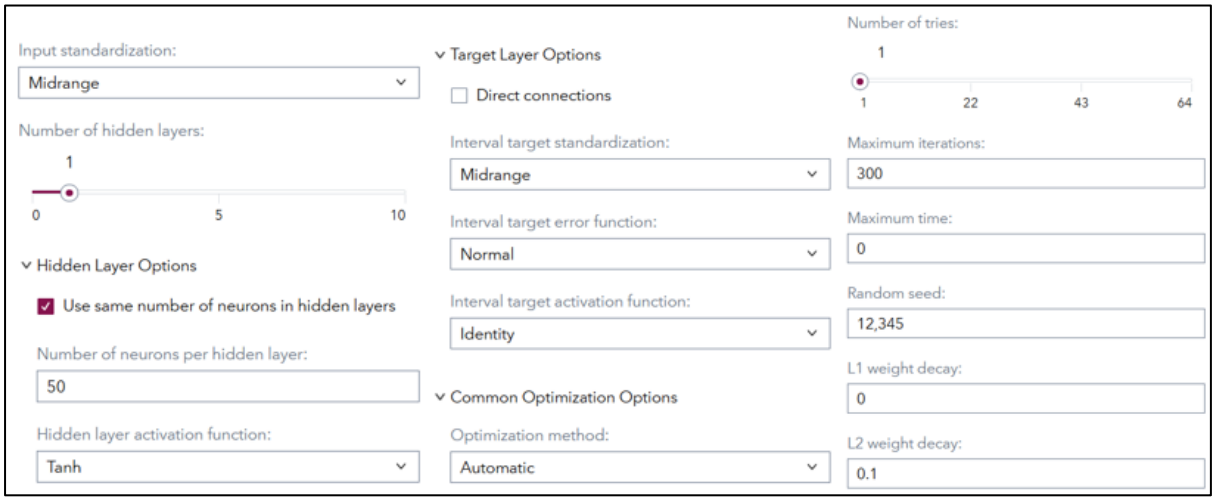

Figure 52: Neural Network Model Origin Tuning

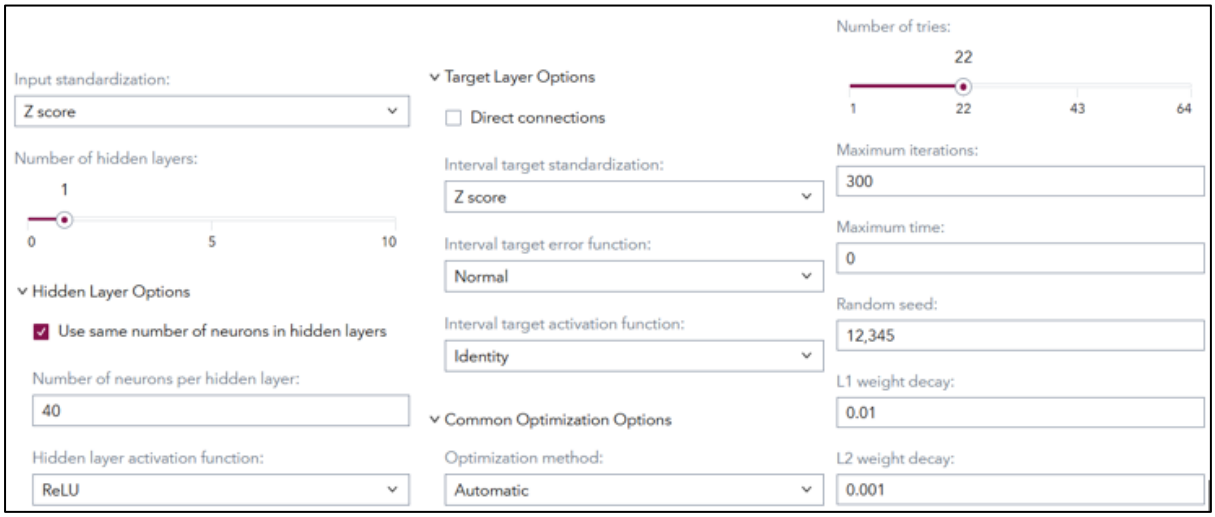

Figure 53: Neural Network Model Modified Tuning

Table 3: Neural Network Modify & Origin Tuning

Parameter Category	**Parameter**	**Neural Network Origin Tuning**	**Neural Network Modified Tuning**
Basic Configuration	Input standardization	Midrange	Z score
	Include missing inputs	No	No
Input Standardization	Method	Midrange	Z-score
Network Structure	Number of hidden layers	1	1
	Use same number of neurons in hidden layers	Yes	Yes
	Number of neurons per hidden layer	50	40
	Hidden layer activation function	Tanh	ReLU
	Direct connections	Enabled	Enabled
Target Configuration	Interval target standardization	Midrange	Z-score
	Interval target error function	Normal	Normal
	Interval target activation function	Identity	Identity
Optimization Settings	Optimization method	Automatic	Automatic
	Number of tries	1	22
	Maximum iterations	300	300

	Maximum time	0	0
	Random seed	12,345	12,345
Regularization	L1 weight decay	0	0.01
	L2 weight decay	0.1	0.001
Advanced Settings	Number of LBFGS corrections	6	6

The initial neural-network model was defined having one hidden layer with fifty neurons and hyperbolic tangent activation function. Mid-range standardisation was used in data preprocessing, and regularisation was little (L1 = 0, L2 = 0.1). The amount of simplicity and quick training cycles was put ahead in this configuration, but a slight vulnerability to overfitting still persisted.

The tunned model was modified to increase the levels of generalisation and stability through the implementation of Z -score normalisation and rectified linear unit (ReLU) activation. This convergence increases faster and enhances performance indicators. Also, the initialisation number was lifted to 22 rather than 1 which has more extensively covered the parameter space. The regularisation parameters were changed to L1 = 0.01 and L2 = 0.001, thus, trading off model flexibility and sparsity. Lastly, the neuron count was reduced to 40 to control model complexity.

Overall, the modified model focuses more on robustness and predictive accuracy and the baseline model on computational efficiency and parsimony.

3.4.3.2 Modify Model & Origin Model Result Comparison

Model Comparison

Champion	Name	Algorithm Name	Accuracy
true	Neural Network (1)	Neural Network	0.8240
false	Neural Network_Origin	Neural Network	0.7454
false	Neural Network (5)	Neural Network	0.8230
false	Neural Network (6)	Neural Network	0.8230

Figure 54: Model Comparison Table Neural Network

The table above compares the performance of four different neural-network configurations in terms of accuracy. The model that has the largest accuracy is referred to as “Neural Network (1)” because it attained the highest accuracy of 0.8240 (82.40%). The other competitors, the first and the second successions, are slightly worse in performance, models are “Neural Network (5) and (6)” with accuracy 0.8230 and the original model has 0.7454. Such results are effective corroborations of the assertion that “Neural Network (1)” is the best configuration to be used among the tested ones in the task given.

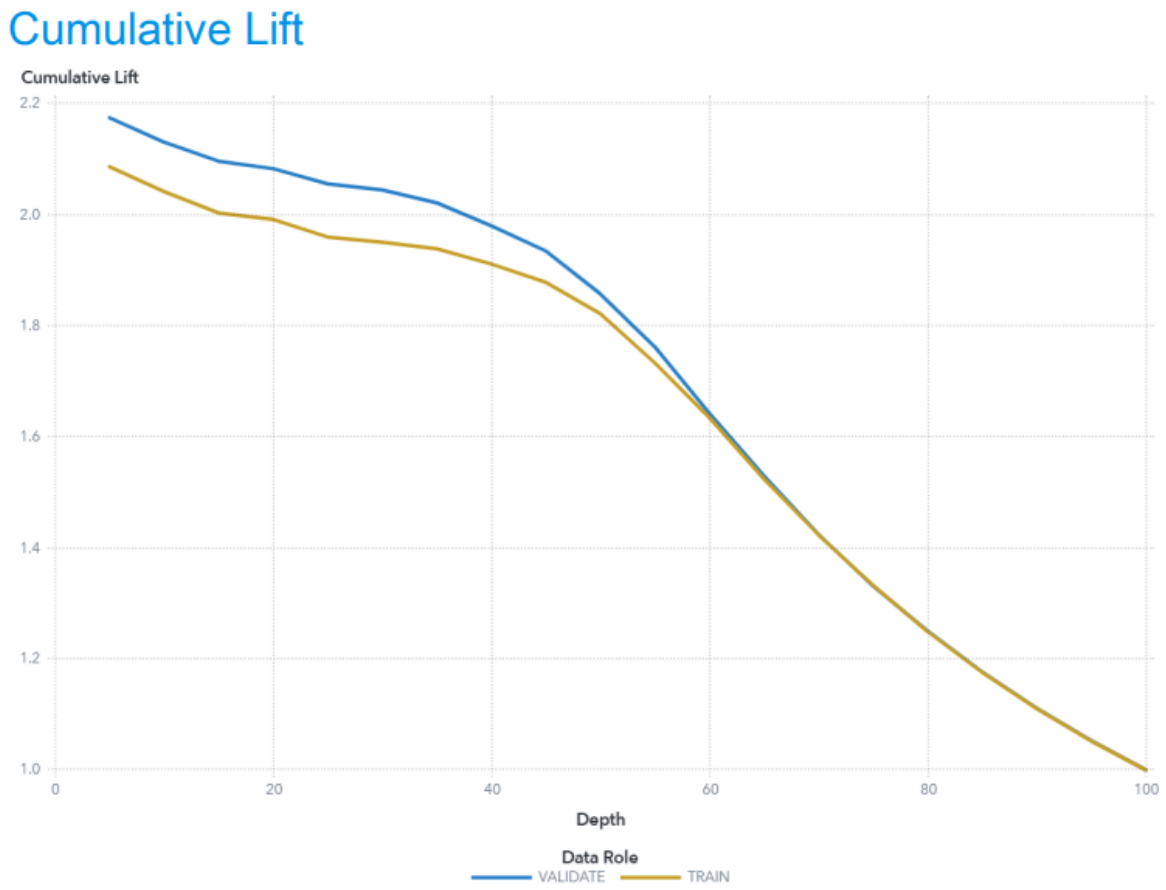

Figure 55: Cumulative Lift Graph Neural Network

The VALIDATE partition gave the cumulative lift of 2.13 at the 10% quantile which implies that the model is capable of identifying subscribers by approximately 2.13 times the likelihood of responding as compared to just picking them by random number. Similarly, the lift derived with the TRAIN partition was 2.04, which demonstrated uniform performance between the two subsets. Since the 2 lift values are higher than unity, then it is obvious that the model clearly more effectively distinguishing potential responders than there is a chance. Cumulative lift is by ranking observations by predicted probability (P_yyes) and dividing the observations into quantiles. Cumulative lift above two indicates a good amount of targeting and will increase marketing accuracy.

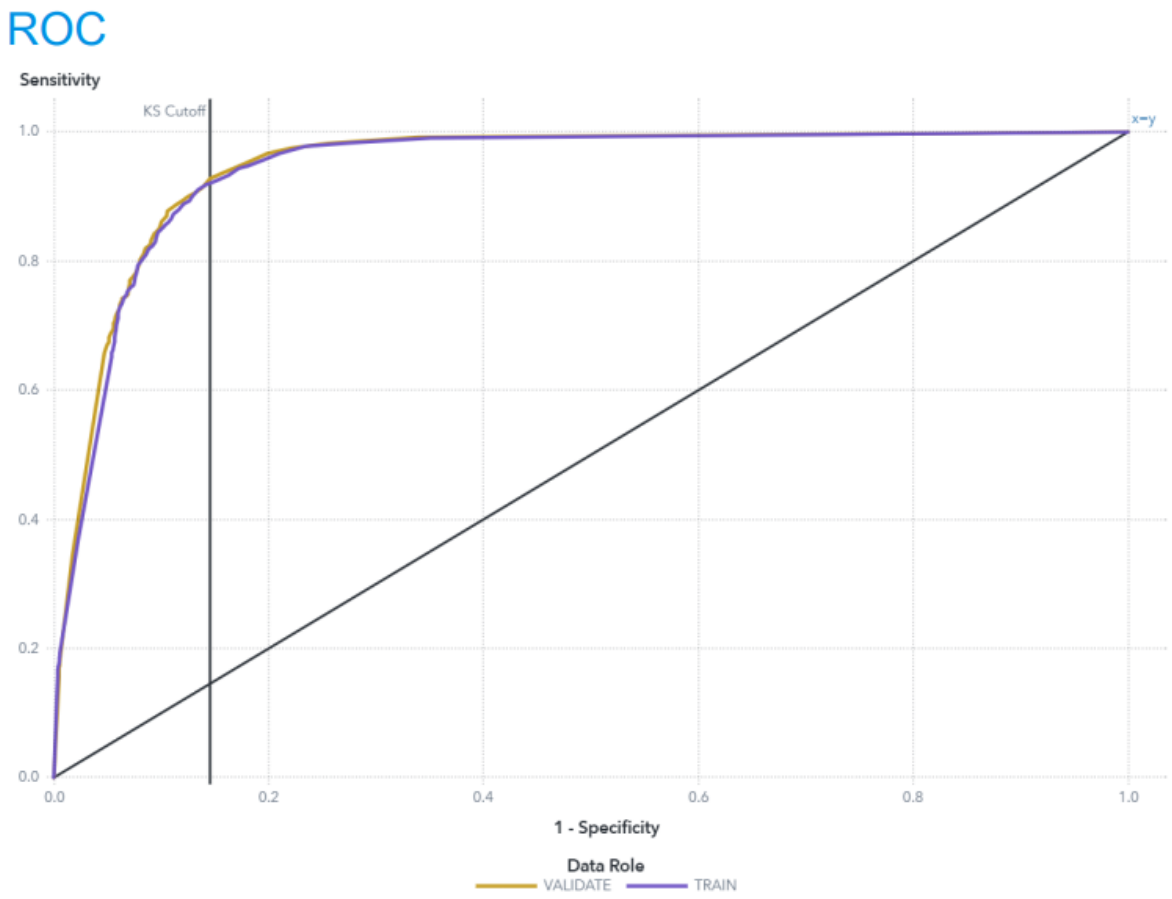

Figure 56: ROC Graph Neural Network

The ROC curve plots a sensitivity (true-positive rate) versus implementation of 1-specificity (false-positive rate) along a range of decision thresholds thus evaluating the ability to classify. The maximum point at which the difference between sensitivity and 1 -specificity is the greatest is the KS cutoff reference line- at a threshold of 0.08, 1 -specificity = 0.146 and sensitivity = 0.928. Thresholds are considered between 0 and 1 with a 0.01 step between them, where the value described by predicted probability (P_yyes) as “yes” or “no”, in this case yes and no represent the observed values of users of the eBay eLearning site. The output is a confusion matrix with each threshold with true positives (TP) and false positives (FP), true negatives (TN), and false negatives (FN).

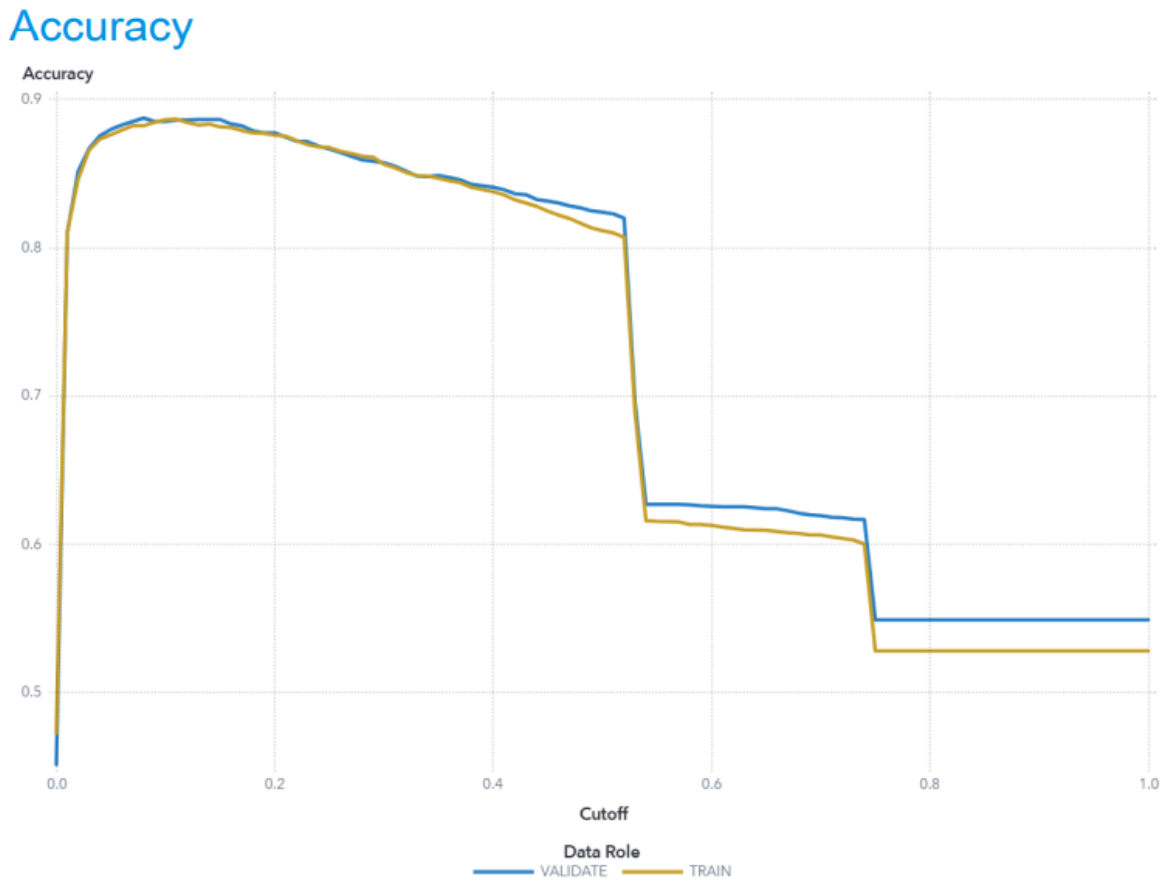

Figure 57: Accuracy Graph Neural Network

In this model, cut-off at 0.5 is 0.812 in TRAIN partition and 0.824 in VALIDATE partition with accuracy which means an equal measure of reliability. Accuracy measures the percentage proportion of the correctly classified events and non-events, both cut-off values, at cut-offs of 0 to 1 in 0.01 steps. The predictions are made at every threshold by comparing the predicted event probability (P_yyes) with the cut-off, whereby when the probability equals or exceeds the cut-off, the observation is denoted as an event and when the probability equals or falls short of the cut-off, the observation is defined as a non-event. There are correct classifications when the predicted and actual outcomes coincide that is to say, when all of them are events (TP) or all of them are non-events (TN) and when there is theism they are referred to as misclassification.

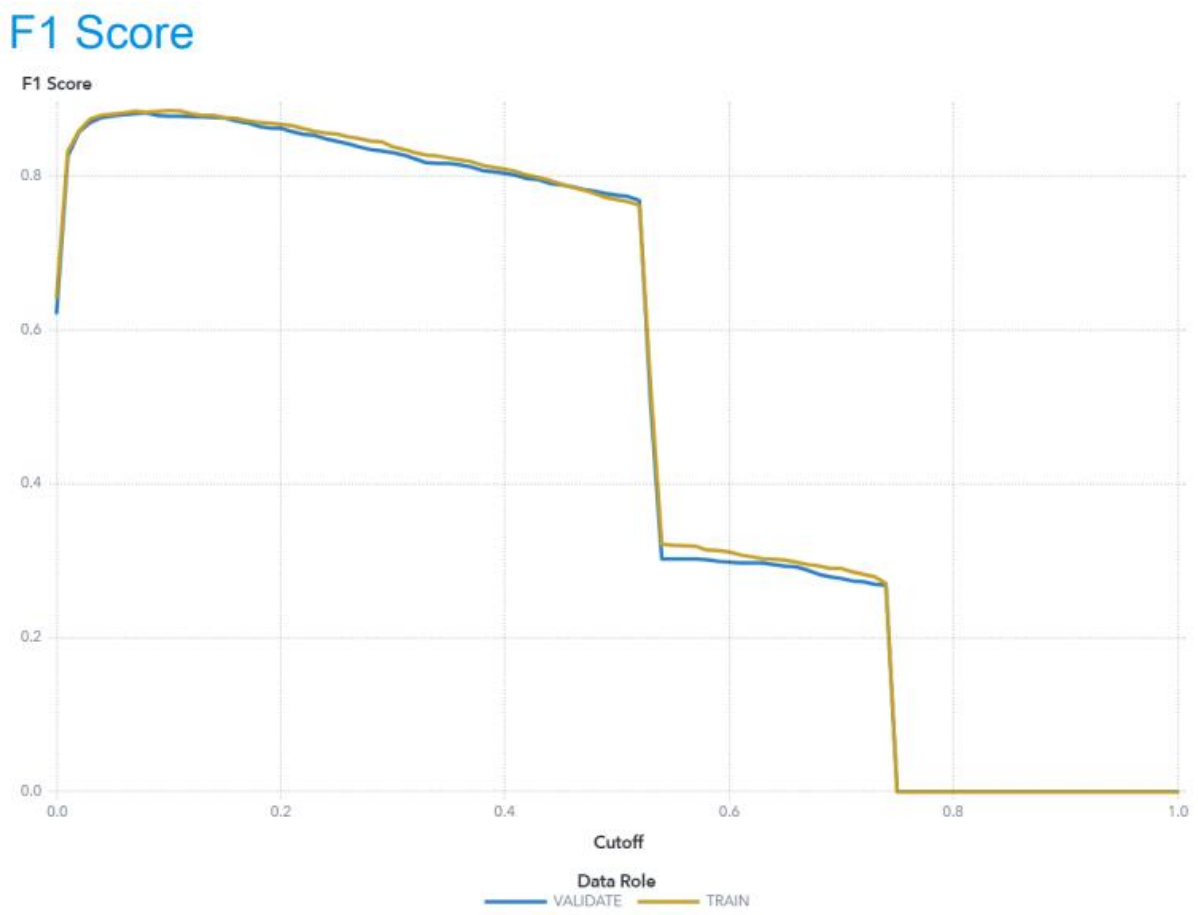

Figure 58: F1 Score Graph Neural Network

TRAIN partition and VALIDATE partition have F1 score of 0.769 and 0.775 respectively at threshold of 0.5 which depicts a sufficient compromise between precision and recall. The F1 score is a harmonic mean between precision and recall, providing one measure, which covers an accuracy of the positive prediction of the correct number of the model and its ability to detect the true occurrence. Again predictions are demarcated by whether the predicted probability (P_yyes) meets or surpasses the threshold. Then they are evaluated subsequently by the use of the confusion matrix to provide assumptions of preciseness, recall, and later the F1 score over the whole range of thresholds.

Fit Statistics

Target Name	Data Role	Partition Indicator	Formatted Partition
y	TRAIN	1	1
y	VALIDATE	0	0

Number of Observations	Average Squared Error	Divisor for ASE	Root Average Squared Error
4,750	0.1649	4,750	0.4061
3,017	0.1607	3,017	0.4009

Misclassification Rate	Multi-Class Log Loss	KS (Youden)	Area Under ROC
0.1884	0.4968	0.7765	0.9430
0.1760	0.4857	0.7825	0.9466

Gini Coefficient	Gamma	Tau	KS Cutoff
0.8860	0.9048	0.4417	0.1000
0.8931	0.9112	0.4424	0.0800

KS at User-Specified Cutoff	Misclassification Rate at KS Cutoff (Event)	Misclassification Rate (Event)
0.6075	0.1135	0.1884
0.6206	0.1124	0.1760

Figure 59: Fit Statistic Neural Network

This table provides extensive fit statistics of training and validation data, indicating good performance and generalisation. As the Average Squared Error (ASE) is small and stable (0.16), it is an indication that there are slight erroneous predictions. Crucially, the Misclassification Rate on the validation data (0.1760) is a bit smaller than the training rate (0.1884) which indicates that the model is slightly better on the unseen data.

This model has a high discriminatory power indicated by high and balanced Area under Receiver operating Characteristic Curve (AUC) values (0.9430 and 0.9466 during training and validation). The Gini coefficient is also equally powerful and steady.

Overall, the consistently high scores in both of the partitions, especially the high AUC and lower error during validation, support the conclusion that the model is highly robust, resists overfitting, and has the dependability of the predictive importance on new data.

Figure 60: Percentage Plot Neural Network

Event Classification report visually provides the confusion matrix of the different cut-off value of each partition. The plot contains the default cut-offs (0.5) and also the KS threshold values of the current partitions 0.1 (TRAIN) and 0.08 (VALIDATE). In this dataset, the segment of the bar corresponding to the event level of y, "yes", the segment of the bar colored as "CORRECT" corresponds to true positives.

3.4.4 Forest Model (YAP BOON SIONG)

3.4.4.1 Modify & Origin Tunning Explanation

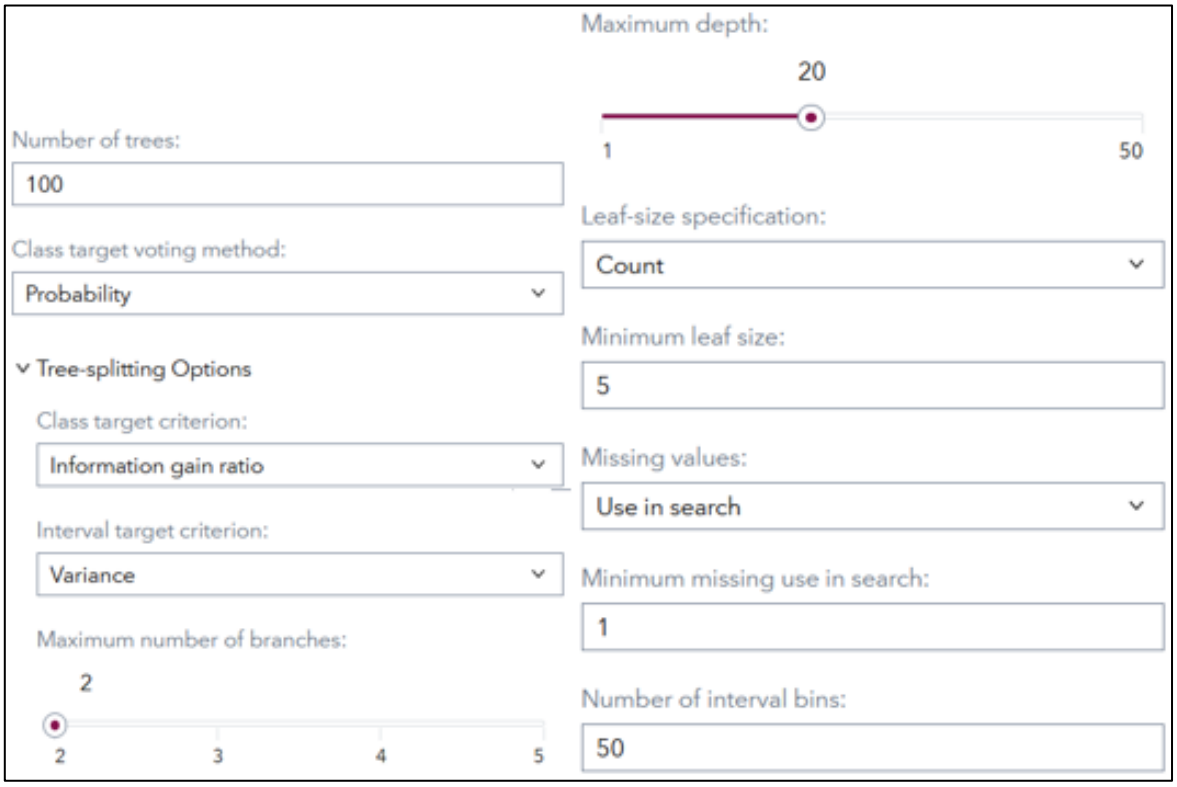

Figure 61: Forest Model Original Tuning

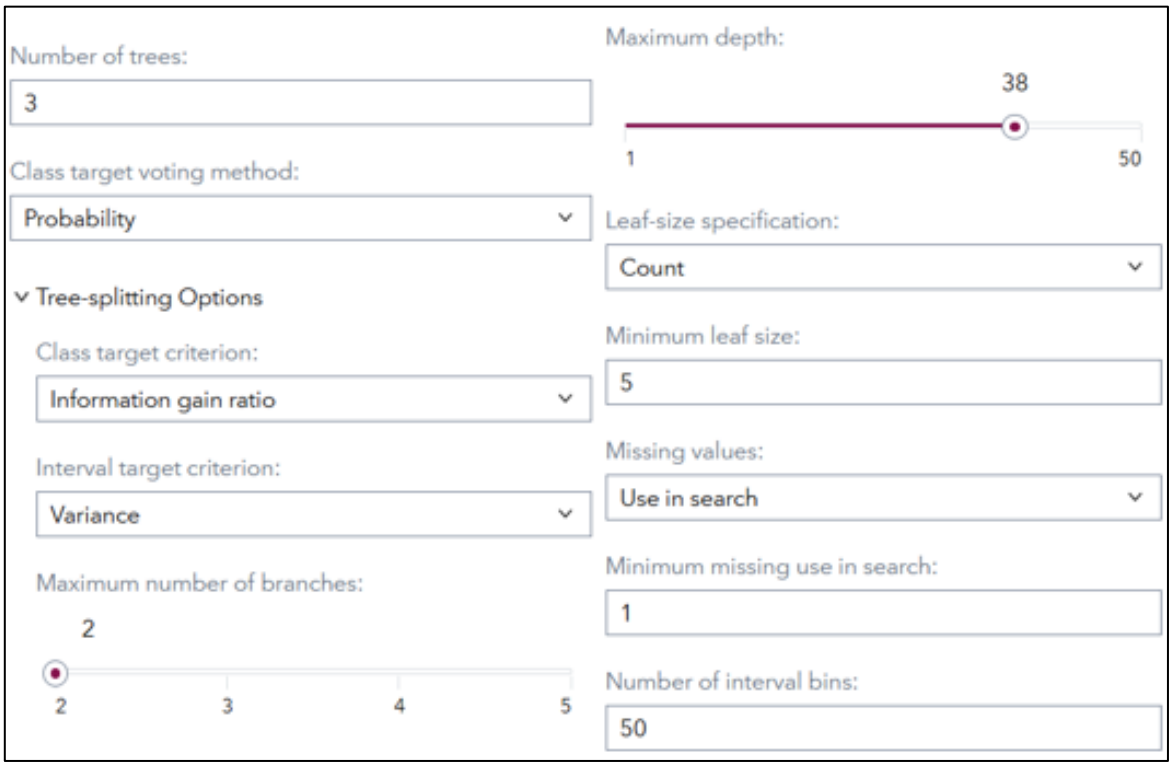

Figure 62: Forest Model Modified Tuning

Table 4: Forest Modify & Origin Tuning

Parameter Category	**Parameter**	**Forest Model Origin Tuning**	**Forest Model Modified Tuning**
Basic Options	Number of trees	100	3
	Class target voting method	Probability	Probability
	Class target criterion	Information gain ratio	Information gain ratio
	Interval target criterion	Variance	Variance
Tree Splitting Options	Maximum number of branches	2	2
	Maximum depth	20	38
	Leaf-size specification	Count	Count
	Minimum leaf size	5	5
Missing Value Handling	Missing values	Use in search	Use in search
	Minimum missing use in search	1	1
Binning Options	Number of interval bins	50	50
	Interval bin method	Quantile	Quantile
Sampling Options	In-bag sample proportion	0.6	0.6
	Inputs to consider per split	Default	Default

The original forest model used consisted of 100 decision trees, limited to the maximum depth of 20 and the minimum size of a leaf containing 5 observations. This design gave importance to a wide coverage of the feature space and a constant inductive process. Using Information Gain Ratio and variance as split criteria, along with an in-bag sample proportion of 0.6, the model provided a cutoff between the performance of the predictive model and their interpretation.

The modified tuning model step entailed the decrease of the number of trees in the ensemble to 3, which increased the efficiency of the computations made and alleviated redundancy. The rest of the parameters like maximum depth slightly changed to 38 and branching parameters were kept constant to ensure structure consistency.

Both models used Quantile binning (50 bins) and search-based handling for missing values. Overall, the original tuning emphasized ensemble strength and stability, while the modified tuning focused on faster computation and capturing more detailed patterns.

3.4.4.2 Modify Model & Origin Model Result Comparison

Model Comparison

Champion	Name	Algorithm Name	Accuracy
true	Forest (2)	Forest	0.7564
false	Forest (1)	Forest	0.7199
false	Forest	Forest	0.7501
false	Forest_Origin	Forest	0.7418

Figure 63: Model Comparison Table Forest

This table provides a Model Comparison of four variations of the Forest algorithm, evaluated by their Accuracy. The arrangement referred to as “Forest (2)” was the best model with an accuracy of 0.7564 (75.64%). All other variants such as the original model of “Forest_Origin” with an accuracy (0.7418), “Forest (1)” with an accuracy (0.7199) and “Forest” with an accuracy (0.7501) were less predictive. In such a way, “Forest (2)” is the best ensemble among considered entries which provides the best predictive performance in the given task.

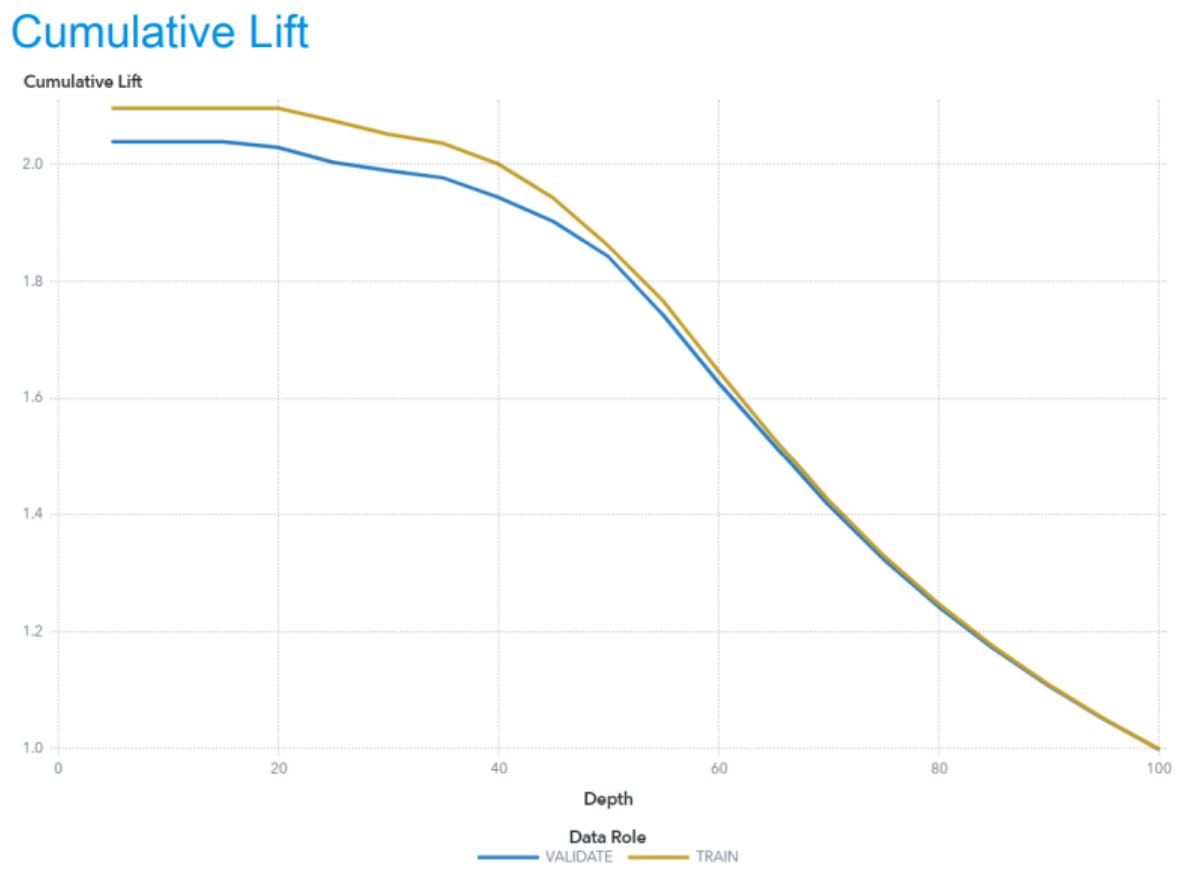

Figure 64: Cumulative Lift Graph Forest

The VALIDATE partition has reached a cumulative lift of 2.04 in its current evaluation and has a lift of 2.1 at the 10% in the TRAIN partition. These values show that at any given highest decile of the predicted events, twice the true occurrences are picked in comparison with a random selection threshold. Both the numbers of lifts are more than 1. Hence, it indicates that the model is more proficient in identifying likely responders rather than by chance. Cumulative lift is computed by ranking observations in order of the likelihood of returning the positive response (P_yyes) and finding the observed cumulative proportion of events in each quantile and comparing it to the prediction with random sampling.

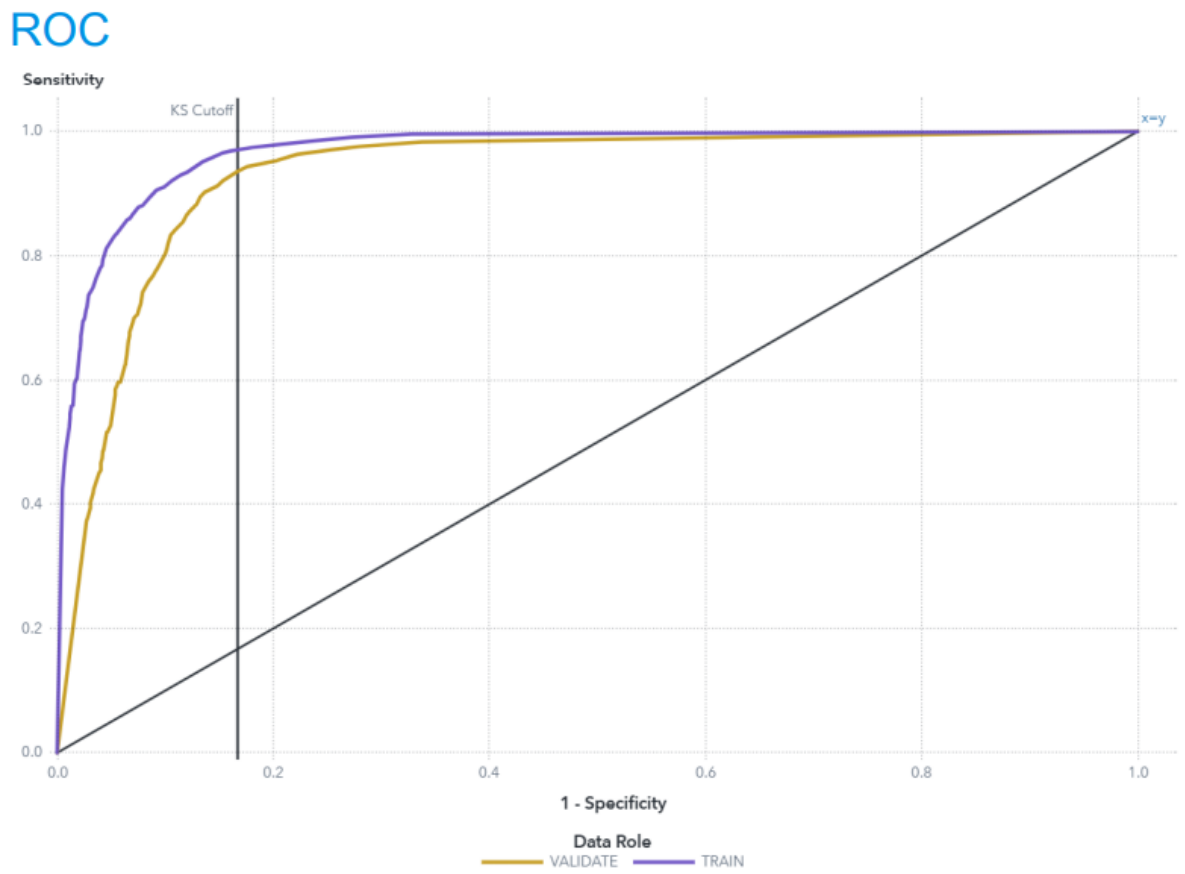

Figure 65: ROC Graph Forest

The receiver operating characteristic (ROC) curve is presented as the cutoff of sensitivity (true positive rate) and 1-specificity (false positive rate) at different probability levels, hence giving a complete evaluation of performance in classification. In this analysis, the KolmogorovSmirnov (KS) reference line crossed the ROC curve with a cut off value of 0.07 with sensitivity of 0.935 and 1-specificity of 0.167. It is the maximization of the distance between the positive and negative partition of VALIDATE indicating optimum balance between event detection and restriction of false events. The ROC curve is constructed by calculating predictions based on whether the predicted probability of the event (P_yyes) exceeds each cutoff value, ROC curve was developed by then obtaining confusion matrices to true positives (TP), false positives (FP), false negatives (FN) and true negative (TN). In most cases, the more plot under the ROC (AUC) is the higher the discriminative power.

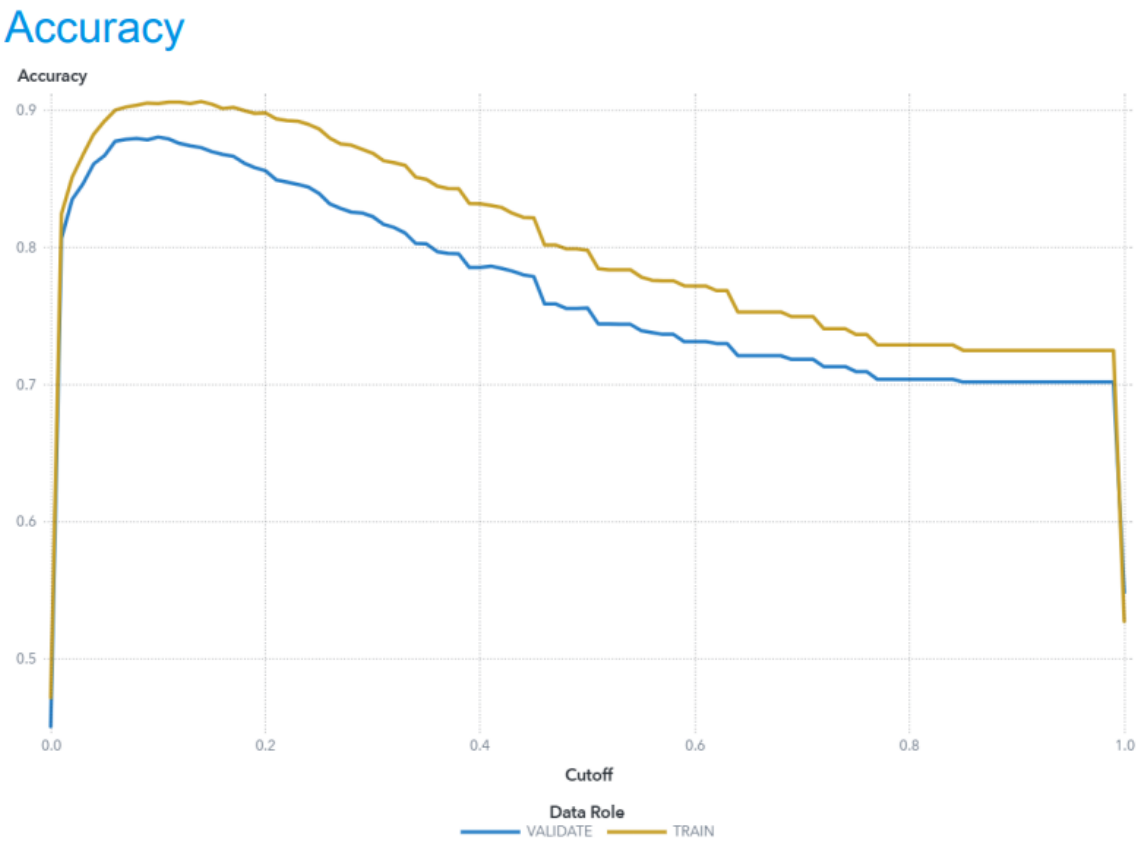

Figure 66: Accuracy Forest

On the TRAIN partition, the model reached a cutoff of 0.5 with accuracy of 0.798, and Validate partition had an accuracy of 0.756. Accuracy is the ratio of correctly since it summarizes both incidences and false incidences. It is calculated by comparing the predicted class, which is assigned based on whether (P_yyes) passes or not passes the cutoff point. These values show that this model works reasonably well in separating event types, and the validated accuracy has a small decrease compared to the training performance which is indicative of a small generalisation variance across data sets.

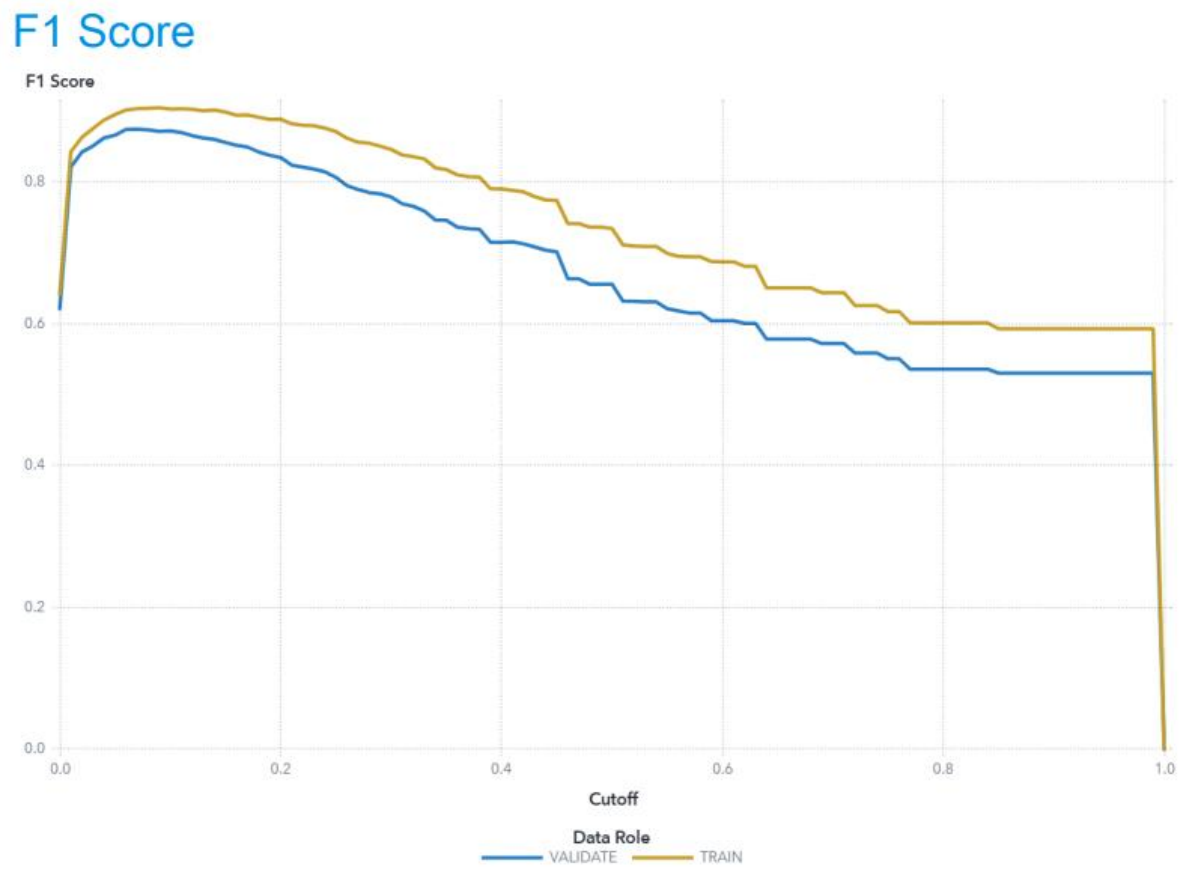

Figure 67: F1 Score Forest

The F1 score of the TRAIN partition can be achieved to 0.735 at a decision threshold of 0.5 whereas the score of the VALIDATE partition can be achieved to 0.656. The F1 measure is the harmonic mean or prediction and recall, such that it balances the trade-off between success in identifying events and false positive (precision). The measure is then based on the confusion matrix and measures the overall balance between useful retrieval and avoidance of the unhelpful. The reduced F1 in the validation set is a sign that there is a slight drop in its performance with the unseen data which is an indication of the possible sensitivity to generalisation. In general, such a model has adequate predictive validity and has a similar performance on both training and validation subsets.

Fit Statistics

Target Name	Data Role	Partition Indicator	Formatted Partition
y	TRAIN	1	1
y	VALIDATE	0	0

Number of Observations	Average Squared Error	Divisor for ASE	Root Average Squared Error
4,750	0.1283	4,750	0.3582
3,017	0.1628	3,017	0.4034

Misclassification Rate	Multi-Class Log Loss	KS (Youden)	Area Under ROC
0.2015	0.4444	0.8165	0.9682
0.2436	0.8586	0.7687	0.9314

Gini Coefficient	Gamma	Tau	KS Cutoff
0.9365	0.9434	0.4669	0.0900
0.8629	0.8846	0.4275	0.0700

KS at User-Specified Cutoff	Misclassification Rate at KS Cutoff (Event)	Misclassification Rate (Event)
0.5752	0.0941	0.2015
0.4699	0.1206	0.2436

Figure 68: Fir Statistic Forest

This is reflected in a bold statistical analysis of model fit, with an area under AUC curve of 0.9682 and a low ASE at 0.1283 which means that it works well on the training data.

However, important metrics start to worsen on the validation sample: the AUC achieves a lower value of 0.9314, the misclassification rate increases to 0.2436, and the log loss increases to 0.8586.

This large performance disparity represents a result of overfitting in that the model picks up specifics within the training data that do not carry forward to novel evidence. Even though absolute accuracy is an acceptable level, the discrepancy implies that further parameter tuning is required to increase the generalisation and reliability of novel data.

Figure 69: Percentage Plot Forest

The event classification report summarizes visually the confusion matrix over a range of decision thresholds of each partition of data. Cutoffs to be used are the default 0.5 and experimentally derived KS cutoffs of 0.09 (TRAIN) and 0.07 (VALIDATE). In the visual representation, the bar that related to the event label yes includes a section coloured "CORRECT", indicating true positives.

3.4.5 Decision Tree (CHEAH JUN HENG)

3.4.5.1 Origin and Tuned Explanation

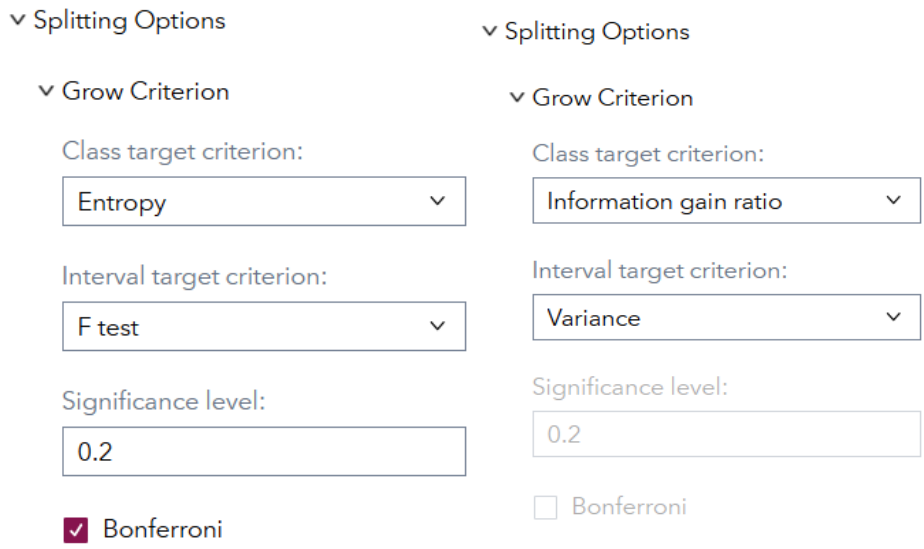

Figure 70: Grow Criterion Original (Left) and Tuned (Right)

The initializing value of Decision Tree's grow criterion was adjusted to improve model accuracy through more informative splitting. Early implementation of the model used Information Gain Ratio for classification to penalize features which contain many "categories" that might lead to overfitting. But in the case of this dataset, Entropy is discovered to give more accurate model. Entropy solely punishes node impurity and is advantageous if features with larger number of categories are truly more informative for prediction (i.e. trees can fully exploit large predictive power). Similarly, for numeric target the criterion was switched from Variance to F Test. The F Test is stronger statistically, as it looks to see if the change in means between groups are from two different points on (so a split that not only will be numerically different but also statistically significant). This mixture of the Entropy and F test with a significance level as 0.2 tries to create a dominant, yet accurate tree by choosing pure and statistically significant splits.

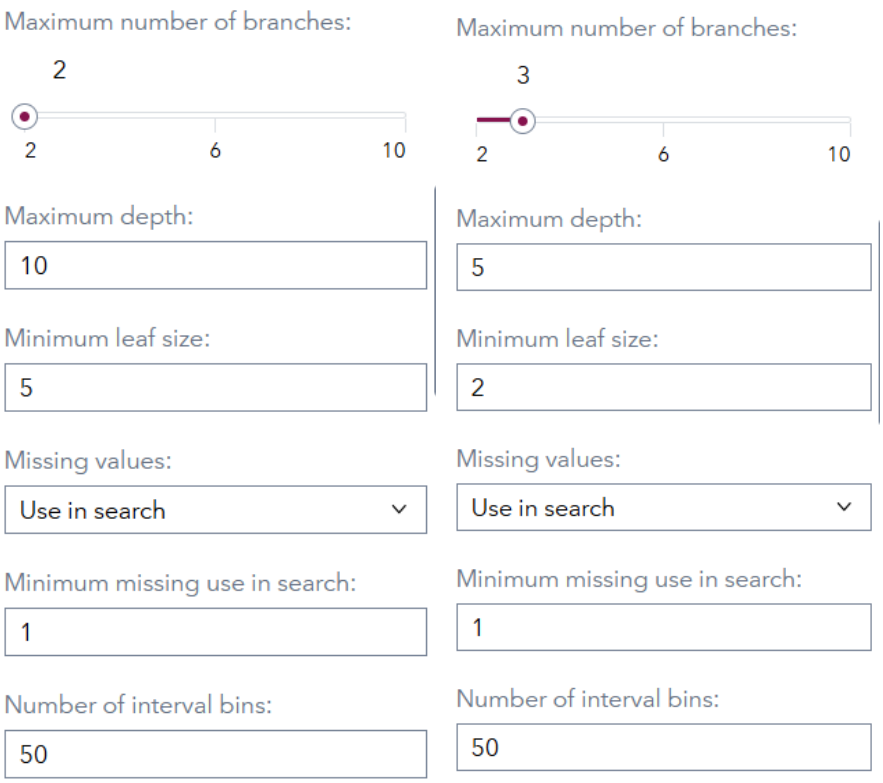

Figure 71: Branches Origin (Left) Tuned (Right)

Tuning the structural constraints of the Decision Tree was essential to trade-off model simplicity and generalisation. The settings that weren't modified (left) created an extremely complex tree with max depth of 10 and min samples leaf of 5. Although this can represent complex patterns it tends to overfit, that is, the model does not generalize and provides poor performance on new data.

The parameters were tuned to build a stouter model (right). The maximum depth was decreased from 10 to 5, which also limits by direct the number of sequential questions that the tree can ask and thus helps to prevent it from learning too much of very specific rules. Moreover, the minimum leaf size was decreased from 5 to 2 in order that tree could be able to build more pure and specialized nodes. Most importantly, the maximum number of branches is set as 3 to allow multi-way splits (that can divide data more effectively than binary ones) by which it constructs a simpler and more accurate tree. Together, these changes result in a simpler and more general model that is less likely to overfit and lose generality on held-out data.

3.4.5.2 Modify Model & Origin Model Result Comparison

Model Comparison

Champion	Name	Algorithm Name	Accuracy
true	Decision Tree (1)	Decision Tree	0.8237
false	Decision Tree_Origin	Decision Tree	0.5582
false	Decision Tree (2)	Decision Tree	0.7242
false	Decision Tree (4)	Decision Tree	0.8064

Figure 72: Decision Tree Model Comparison

The findings display a hierarchy in accuracy of the models, the tuned Decision Tree (1)'s classification has significantly higher 0.8237 (82.37%). This score is much higher than that of Decision Tree_Origin model (0.5582, 55.82%), and thus, the default parameters of decision tree were very suboptimal for this dataset. The other tuned versions, Decision Tree (2) and (4), obtained accuracies of 0.7242 and 0.8064, respectively, further reinforcing that the setting used for the champion model presumably a different criterion to grow or prune trees proved to be the more suitable in terms of capturing systematics from the data without overfitting.

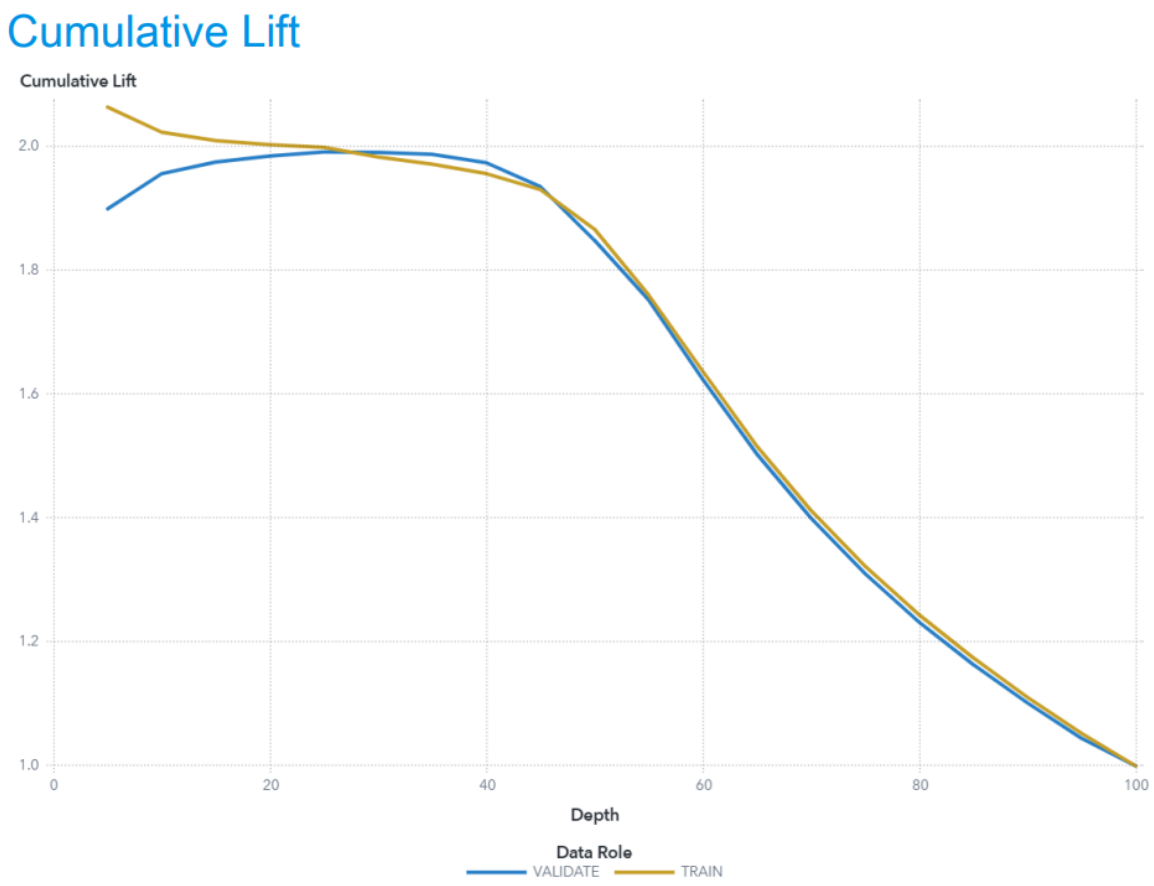

Figure 73: Decision Tree Cumulative Lift

The Cumulative Lift chart compares how well the various Decision Tree models can saliency in your most likely customers. If the value of lift for a given depth is higher, better is our model. From the chart, Decision Tree (1) VALIDATE (the blue line) leads at every stage and has highest cumulative lift, therefore it's confirmed as our winning model. It is also much higher than the green line, which was our Validation lift of our Decision Tree_Origin VALIDATE model and had a low lift score meaning hyperparameter tuning worked wonderfully in this case. The close train-validate performance in the champion model also implies that it does not overfit and generalizes well to new data.

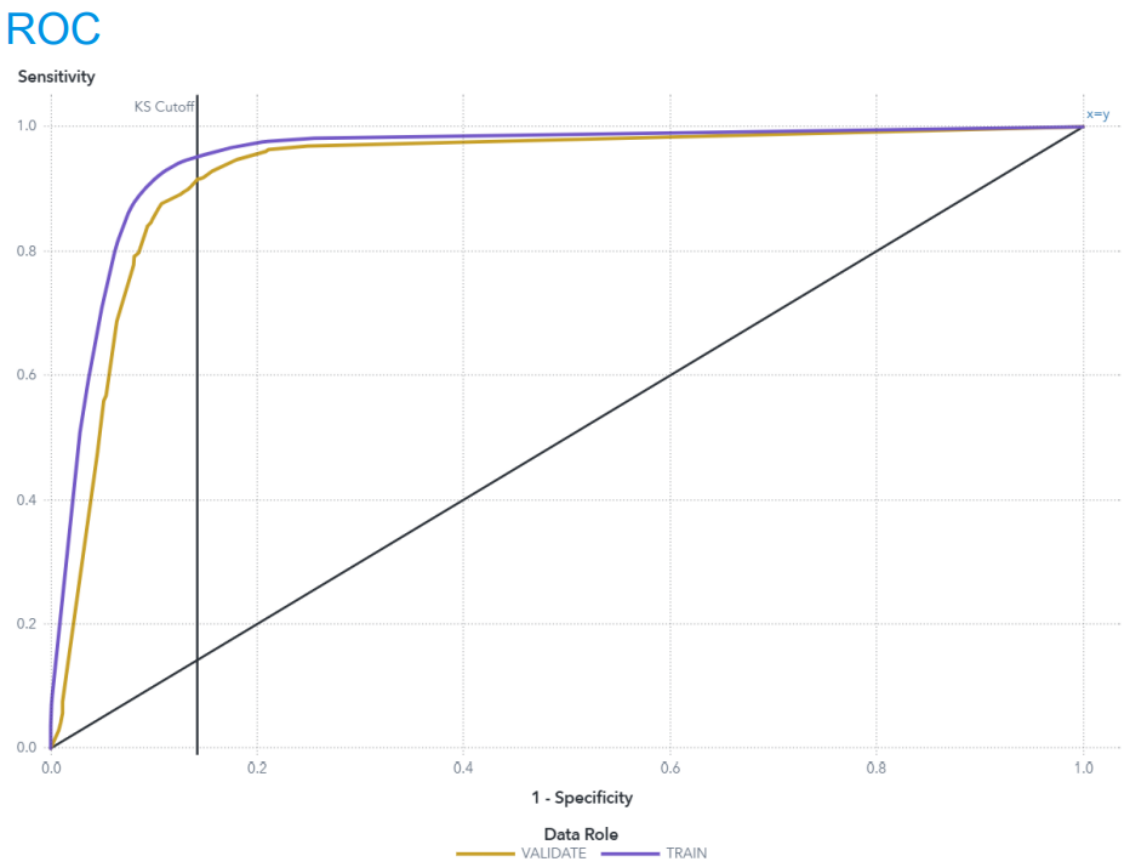

Figure 74: Decision Tree ROC

ROC curve The ROC (Receiver Operating Characteristic) curve is used to evaluate the classification of models at various thresholds. The main insight is that Decision Tree (1) - VALIDATE has the quickest spike towards the top-left corner (blue line), where performance is optimal – high True Positive Rate + low False Positive Rate. This certainly cements its position as a winning model. It generalizes well on the validation set close to its training performance and does not overfit. On the other hand, all the other models, including the original, present lower curves indicating their worse discriminative power for classes. The Decision Tree (1) has the highest Area Under the Curve (AUC), indicating best overall predictive performance.

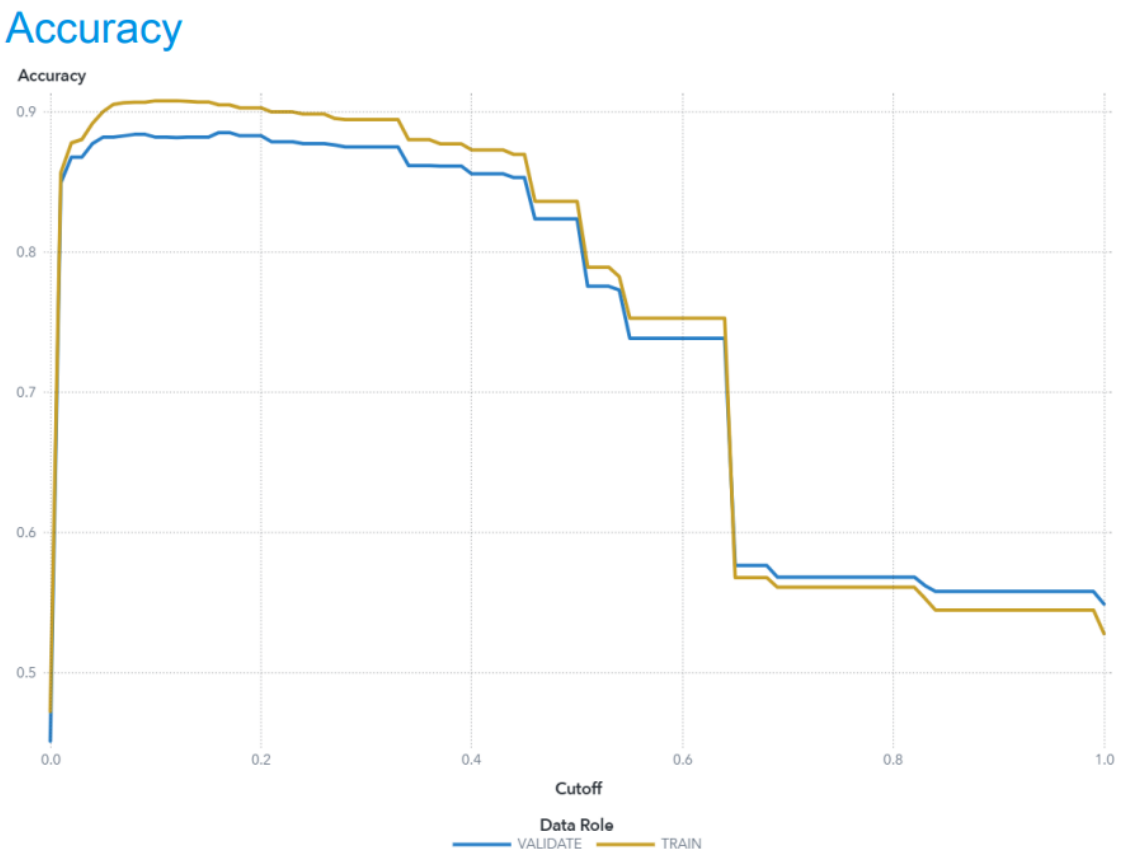

Figure 75: Decision Tree Accuracy

The Accuracy versus Cutoff curve shows the performance consistency of the model with different decision thresholds, where Decision Tree (1) has significantly highest accuracy from the validation set. This best performing model continues to have the highest ROC-AUC score for this threshold and most others, except the left and right tail ends, once again proving its stability and practice value. The huge performance improvement of the tuned champion over the original model clearly demonstrates that hyperparameter optimization has taken place as compared to Random Search and Nelder-Mead, showing a good correlation between training and validation in Decision Tree (1) which generalizes well without overfitting.

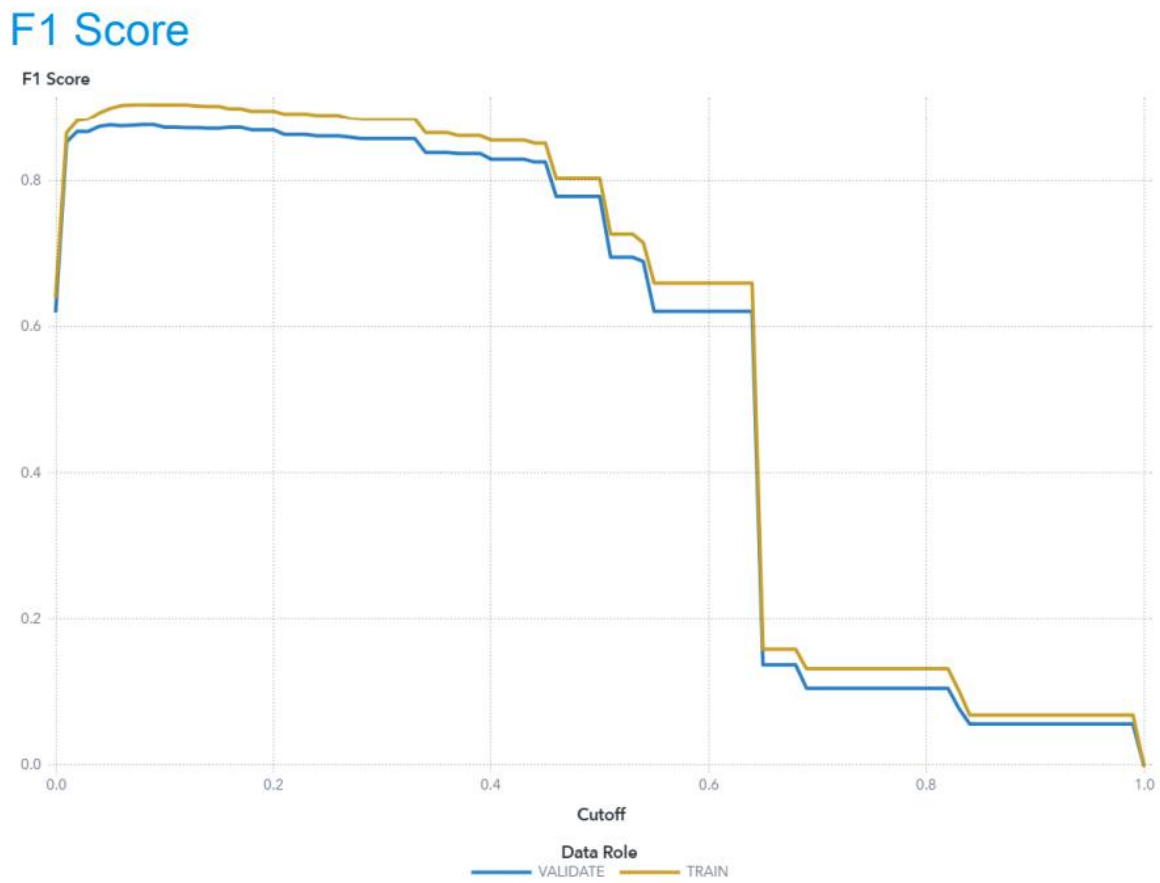

Figure 76: Decision Tree F1 Score

The F1 Score graph checks the power of these models by focusing on balancing precision and recall, which is very important when dealing with imbalanced datasets. The findings indicate that Decision Tree (1) -VALIDATE's peak F1 score remains consistently higher to any other model across the different probability cut-offs, thus establishing itself as the champion model. This means that it provides the best balance between getting right the subscribers (recall) and keeping its positive predictions "clean" (precision). The strong overlap between training and validation performance reveals stable generalization to unseen data. By comparison, the predefined Decision Tree model yielded a much lower and instable F1 score particularly on the validation set, pointing out that hyperparameter optimization has made an important contribution creating a fair and reliable classifier.

Fit Statistics

Statistics Label	TRAIN_Decision_Tree (1)	VALIDATE_Decision_Tree (1)	TRAIN_Decision_Tree_Origin
Area Under ROC	0.9508	0.9270	0.8992
Average Squared Error	0.1381	0.1501	0.1902
Divisor for ASE	4,750	3,017	4,750
Formatted Partition	1	0	1
Gamma	0.9292	0.8915	0.9069
Gini Coefficient	0.9016	0.8540	0.7985
KS (Youden)	0.8184	0.7737	0.7582
KS at User-Specified Cutoff	0.6590	0.6230	0.0232
KS Cutoff	0.1000	0.0800	0.0200
Misclassification Rate	0.1638	0.1763	0.4613
Misclassification Rate (Event)	0.1638	0.1763	0.4613
Misclassification Rate at KS Cutoff (Event)	0.0920	0.1160	0.1227
Multi-Class Log Loss	0.4379	0.6350	0.5928
Number of Observations	4,750	3,017	4,750
Partition Indicator	1	0	1
Root Average Squared Error	0.3716	0.3874	0.4361
Target Name	y	y	y
Tau	0.4495	0.4231	0.3981

Figure 77: Decision Tree Fit Statistics

The Fit Statistics table provides a comprehensive comparison of model performance, confirming Decision Tree (1) as the superior model. On the validation set, which simulates unseen data, it achieves an Area Under ROC of 0.9270 and a Gini Coefficient of 0.8540, both significantly higher than the original model's scores of 0.8992 and 0.7985, indicating much better classification power. Most critically, its Misclassification Rate of 0.1763 (82.37% accuracy) vastly outperforms the original model's rate of 0.4613 (53.87% accuracy). The model also shows a low Average Squared Error of 0.1501, reflecting well-calibrated probability estimates. The consistency between its training and validation metrics, such as the ROC values (0.9508 vs. 0.9270), demonstrates effective generalization without overfitting, validating the tuning process.

3.4.6 Logistic Regression (CHEAH JUN HENG)

3.4.6.1 Origin and Tuned Explanation

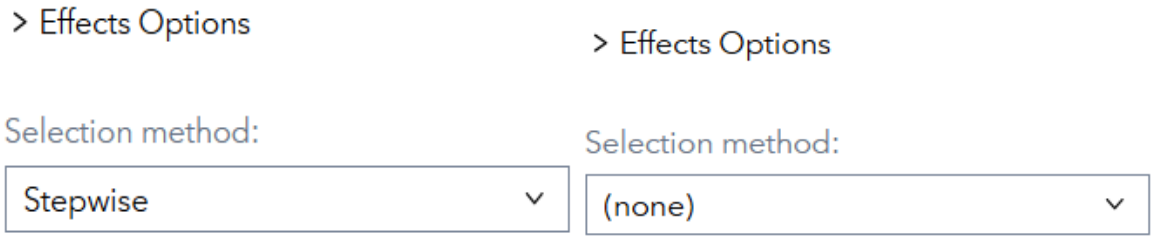

Figure 78: Logistic Regression Effects Options Origin (Left) Tuned (Right)

The logistic regression model was modified for better performance and interpretation. The original setting (on the left) was with Stepwise selection, to automatically include/exclude the variables based on statistical significance. While this may be helpful, it often leads to a fragile model that is very sensitive to small variations in the data.

For a more powerful and less complex model, the adjusted version (right) altered selection method to None. It means to keep all the input variables that addressed to model. The justification is that the exploratory data analysis and variable selection have already isolated the key predictors. Forcing the model to utilize all these pre-vetted variables can result in a more stable and robust model since it removes potential randomness of the stepwise procedure, as well as ensures that all relevant business factors are being accounted for in the final prediction.

3.4.6.2 Modify Model & Origin Model Result Comparison

Model Comparison

Champion	Name	Algorithm Name	Accuracy
true	Logistic Regression (1)	Logistic Regression	0.7759
false	Logistic Regression_Origin	Logistic Regression	0.7663
false	Logistic Regression (2)	Logistic Regression	0.7759

Figure 79: Logistic Regression

The Logistic Regression (1) had the best fitting model with an accuracy of 0.7759 (77.59%), performing a little bit better than the accuracy of 0.7663 (76.63%) obtained from the initial model. It is a small improvement, but suggests that changing the variable selection from Stepwise to None has improved the stability and robustness of the model. The similar accuracy for Logistic Regression (1) and (2) implies that there may be multiple computational combinations thereof which have optimal performance, but we identify one as the champion based upon some implementation details of our system. This in turn demonstrates that even small changes to model settings can result in tangible gains for predictivity.

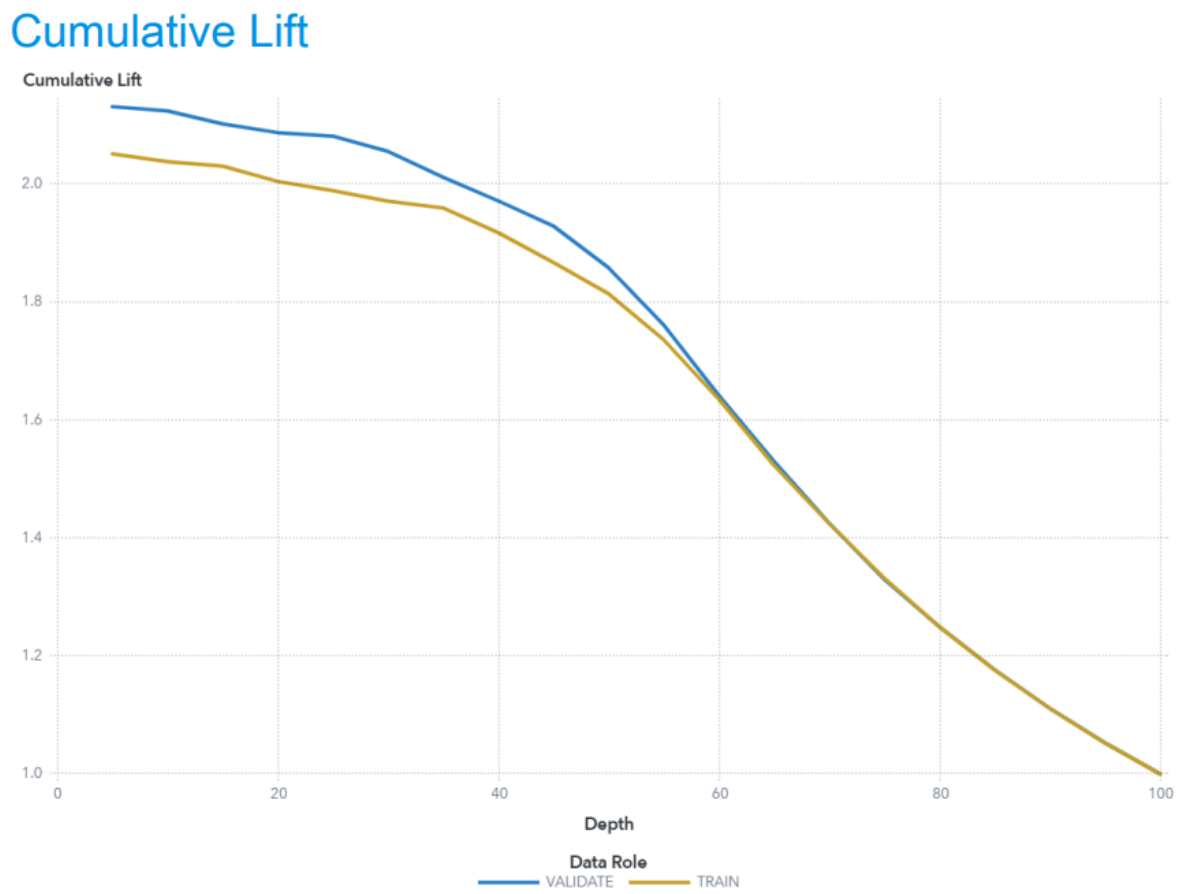

Figure 80: Logistic Regression Cumulative Lift

The Cumulative Lift chart of our champion Logistic Regression model above tells us how well the model is in predicting the most likely people. The VALIDATE curve opens from a high lift value and above baseline of 1 all the depths, confirming that the estimated model enriched customers among which the target is very frequent at those top percentiles. It means that by reaching out to the top 10-20% of customers, as indicated by the model, the marketing team can concentrate far more potential subscribers than a random draw. The narrow gap between the TRAIN and VALIDATE performance curves indicates that the model can generalize well to unseen data while avoiding overfitting and proving its practical value in the marketing campaign.

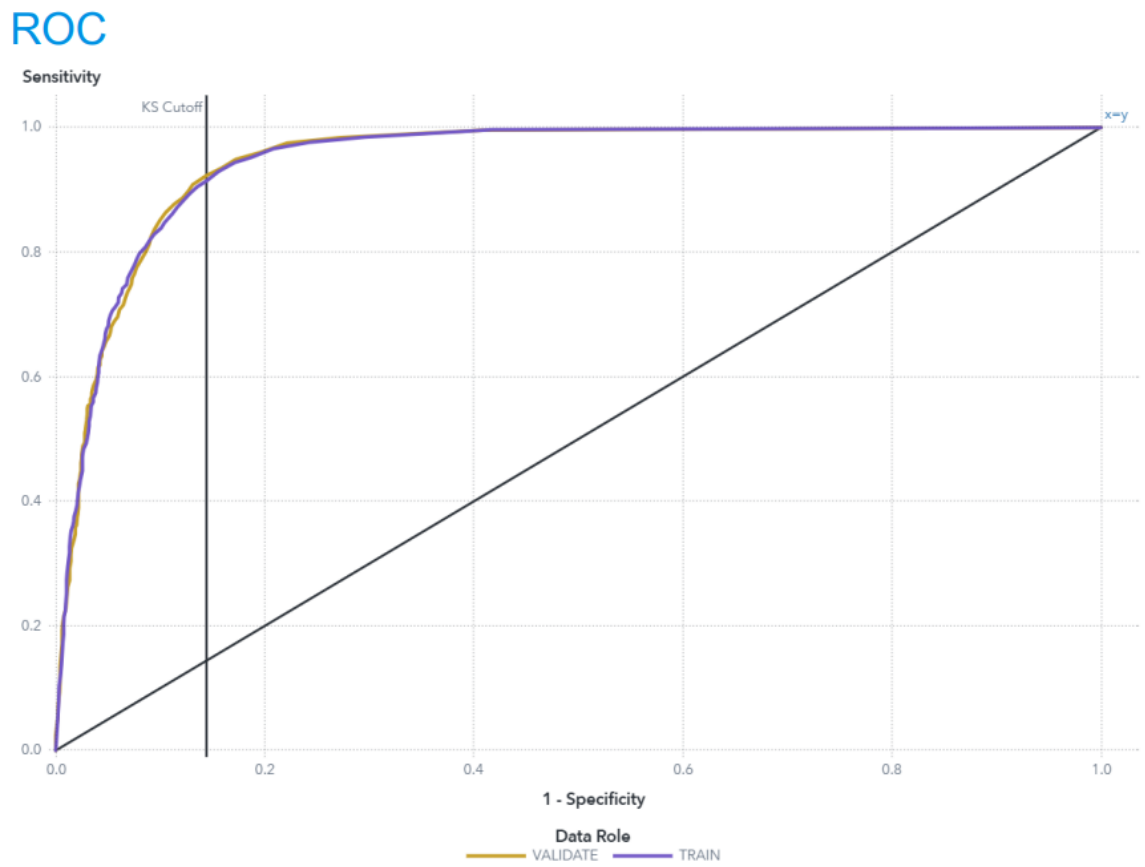

Figure 81: Logistic Regression ROC

ROC curve is a graph that shows the sensitivity (True Positive Rate) against 1-specificity (False Positive Rate), which can be interpreted as a measure of classification derived on confusion matrix. These two indices are calculated at different cutoff points. To assist in determining the optimal cutoff value for scoring your data, a vertical reference line (KS Cutoff) is plotted at the point on the X-axis corresponding to a 1-specificity value where sensitivity and 1-specificity exhibit the maximum difference between them for the VALIDATE partition. The KS Cutoff line is marked by the cutoff 0.07, with 1-specificity = 0.144 and sensitivity = 0.923

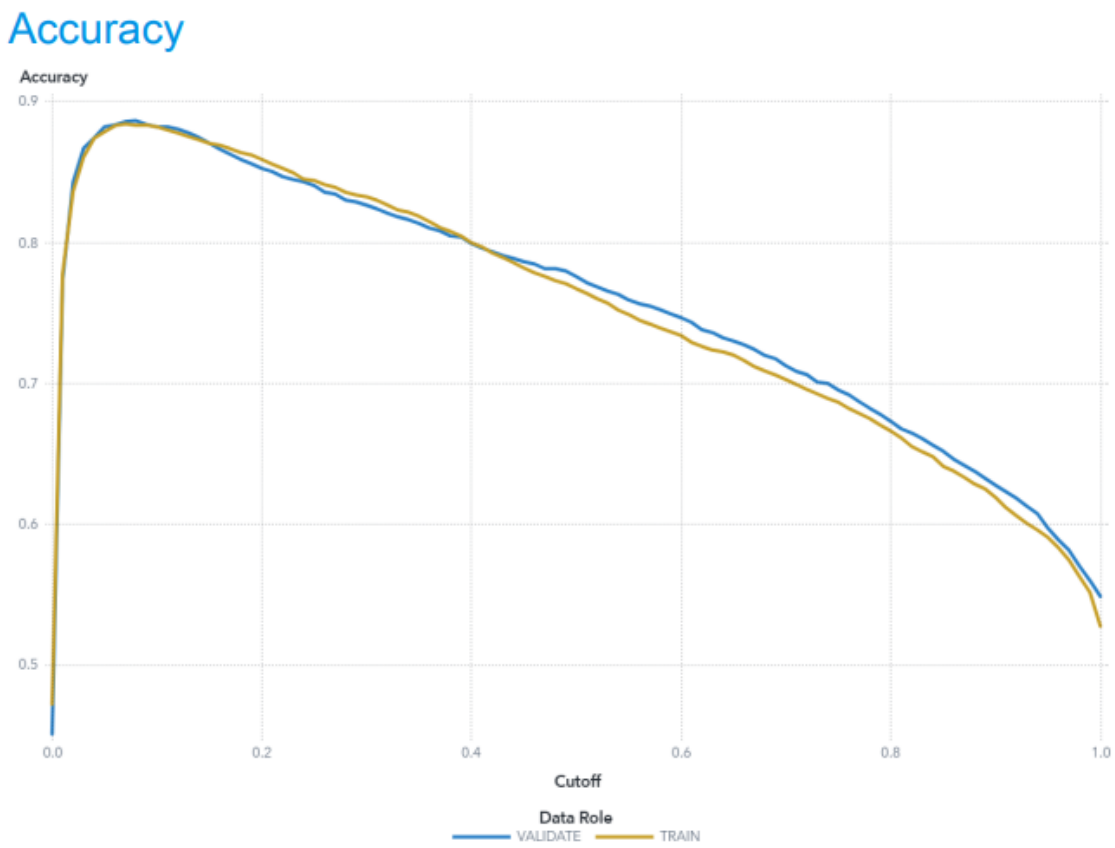

Figure 82: Logistic Regression Accuracy

The TRAIN data accuracy at cutoff 0.5 for this model is 0.767. For such model, metric value of 0.776 in the VALIDATE partition at cutoff-point 0.5. Accuracy is the ratio of correctly classified observations to total observations as event or non-event at different threshold values. Cutoff values are between 0 and 1 (inclusive), with a step of 0.01. For each cutoff value, the predicted classification for the target is based on whether P_yyes (i.e., True probability of "yes" i.e. 1 in our case), is greater than or equal to a certain threshold value. If P_yes is not less than the cutoff, we predict an event; otherwise a non-event. If the predicted classification and actual classification are BOTH event (T), OR if the predicted classification and actual classification are both non-event (F), then would be true positives or true negatives, respectively. If the predicted class differs from the true class of observation, then it is misclassified. Accuracy = (True Positives + True Negatives) / Total Observations.

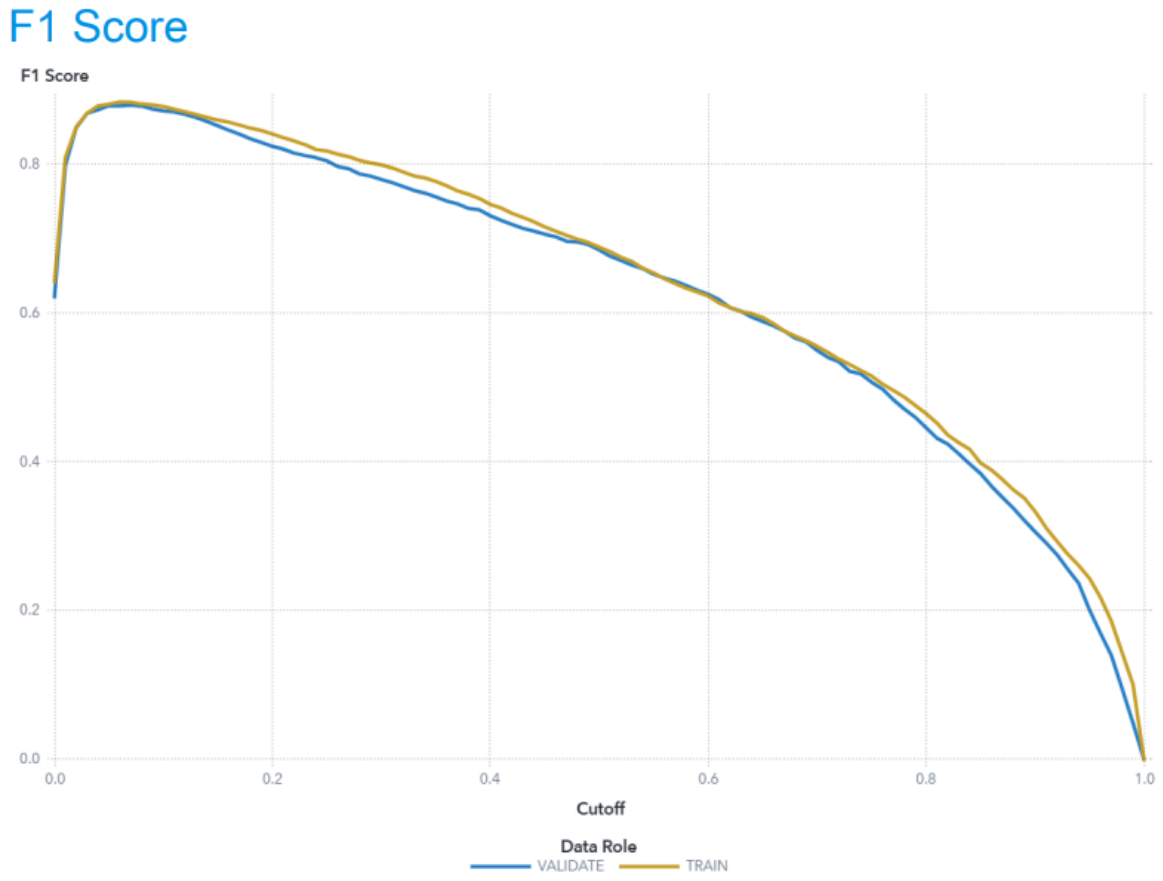

Figure 83: Logistic Regression F1 Score

For this model, F1 score in TRAIN partition at 0.5 cutoff is 0.689 Table 2: Performance in the VALIDATE partition, an F1 cutoff of 0.5 yields a score of 0.685 for this model. The F1 score is the balance between precision and recall (sensitivity) metrics that quantify classification per the confusion matrix across different cut-off values. The cutoff values were from 0 to 1 in steps of 1%. For each cutoff value, the classification for the target is based on whether P_yes (the predicted probability of 'yes' for the target y) is greater or equal to this cutoff value. 2 = event as predicted when P_yes ≥ cutoff threshold otherwise not event. The confusion matrix under each cutoff value consists of four cells that correspond to the true positives for classified events (TP), false positives for classified non-events (FP), false negatives by the classifier (FN) and true negatives by the classifier or correct non-events(TN). Not-Event True negatives are non-event classifications that name a different not-event. 33 Precision is computed as TP / (TP + FP) and recall (or sensitivity) is computed as TP / (TP + FN). F1 score is computed as 2PrecisionRecall / (Precision + Recall), where Precision and Recall are the above defined Precision and Recall. Higher F1 scores correspond to a more accurate model.

Fit Statistics

Target Name	Data Role	Partition Indicator	Formatted Partition
y	TRAIN	1	1
y	VALIDATE	0	0

Number of Observations	Average Squared Error	Divisor for ASE	Root Average Squared Error
4,750	0.1603	4,750	0.4004
3,017	0.1591	3,017	0.3988

Misclassification Rate	Multi-Class Log Loss	KS (Youden)	Area Under ROC
0.2326	0.4887	0.7736	0.9467
0.2241	0.4849	0.7785	0.9469

Gini Coefficient	Gamma	Tau	KS Cutoff
0.8935	0.8994	0.4454	0.0700
0.8938	0.9004	0.4428	0.0700

KS at User-Specified Cutoff	Misclassification Rate at KS Cutoff (Event)	Misclassification Rate (Event)
0.5113	0.1156	0.2326
0.5099	0.1140	0.2241

Figure 84: Logistic Regression Fit Statistics

The Fit Statistics of the winning Logistic Regression model performs well and robustly. The model performs well on the validation set with Area Under ROC equals to 0.9469, and a Gini Coefficient of 0.8938. Its Misclassification Rate of 0.2241 (77.59% accuracy) further demonstrates an acceptable predictive power on novel data. An important point to note is that the model displays strong resistance when tested with training and validation sets across all key metrics – Average Squared Error (0.1603 vs 0.1591) and KS statistics (0.7736 vs 0.7785) which signifies no overfitting. The small Multi-Class Log Loss values (0.4887 vs 0.4849) also support the model's well-calibrated probability estimates, turning it into a reliable and trustworthy instrument for predicting customer subscription.

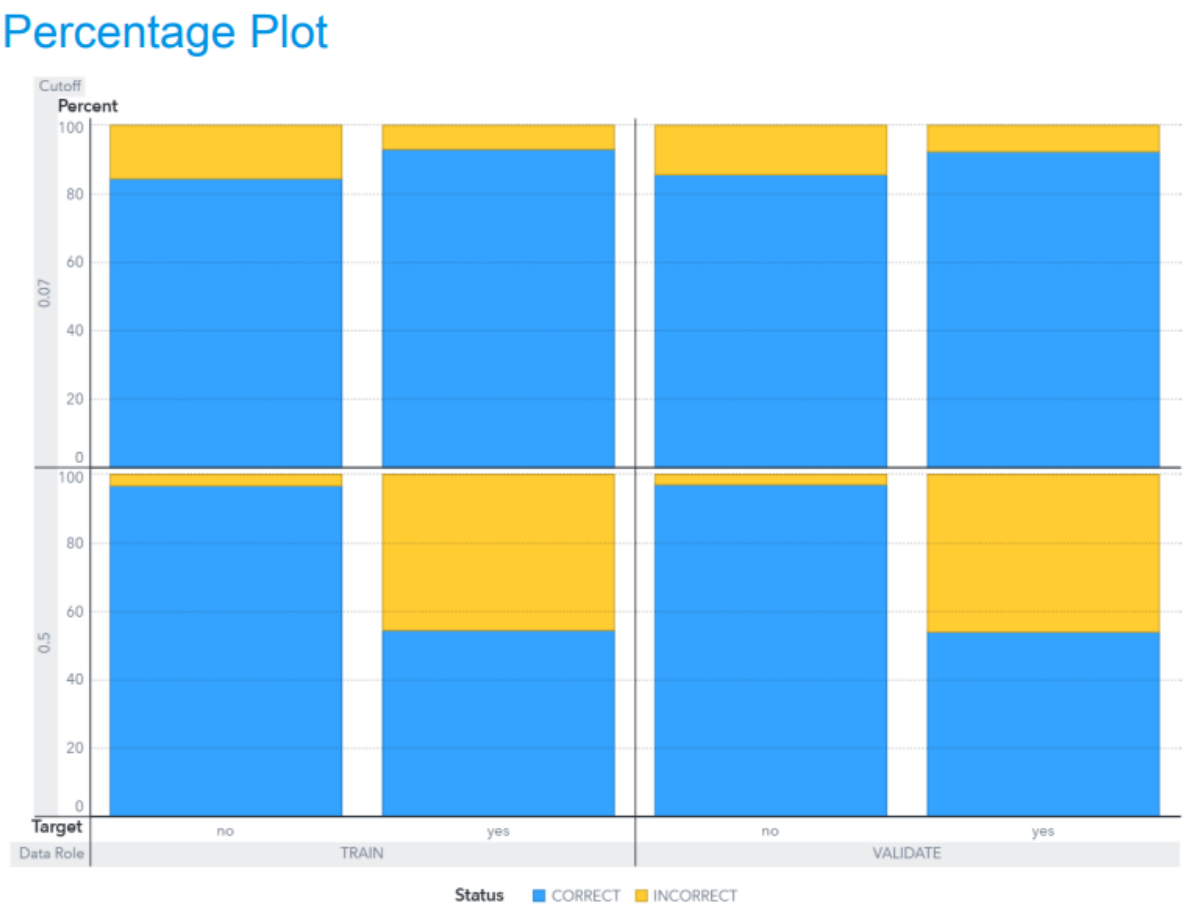

Figure 85: Logistic Regression Percentage Plot

The report of Event Classification is a plot for the confusion matrix at multiple cutoff values for each partition. The KS cutoff for the existing partitions that we used in construction of a plot are: 0.07 (TRAIN), 0.07 (VALIDATE) and the default of 50%. For this data, in the bar for the event-level of y "yes", the part coloured "CORRECT" is true positives.

3.5 Critical Interpretation of Outcomes

3.5.1 Bayesian Network (OOI DUN TZI)

Surrogate Model Variable Importance

Variable Label	Variable Name	Relative Importance	Role	Variable Level
	euribor3m	1	INPUT	INTERVAL
	duration	0.9974	INPUT	INTERVAL
Grouped: nr.employed	GRP_nr.employed	0.9961	INPUT	ORDINAL
Grouped: duration	GRP_duration	0.9666	INPUT	ORDINAL
Grouped: euribor3m	GRP_euribor3m	0.9010	INPUT	ORDINAL
	emp.var.rate	0.8316	INPUT	NOMINAL
Grouped: month	GRP_month	0.4046	INPUT	ORDINAL
Grouped: cons.price.idx	GRP_cons.price.idx	0.3712	INPUT	ORDINAL
Grouped: poutcome	GRP_poutcome	0.3374	INPUT	ORDINAL
Weight of Evidence: duration	WOE_duration	0	INPUT	INTERVAL
Weight of Evidence: job	WOE_job	0	INPUT	INTERVAL
Weight of Evidence: month	WOE_month	0	INPUT	INTERVAL
Weight of Evidence: nr.employed	WOE_nr.employed	0	INPUT	INTERVAL
	pdays	0	INPUT	INTERVAL

Figure 86: Surrogate Model Variable Importance Bayesian Network

Table indicates the relative importance of each variable in the Bayesian Network model. It scores all features according to the contribution of each feature in predicting the target.

Moreover, the “Relative Importance” score is a value from 0 to 1, and it will be representing the strength of impact of each variable on the model.

Key Insights:

- The feature euribor3m is the most important, with a relative importance value of 1.0.
- “duration” and “GRP_nr.employed” follow as next and third most important variables with strong prediction power.
- The WOE_ (Weight of Evidence) variables were ignored by the model, that is irrelevant WOE_, since all these features received an importance score of 0. This mean that they did not participate in the final predictions.

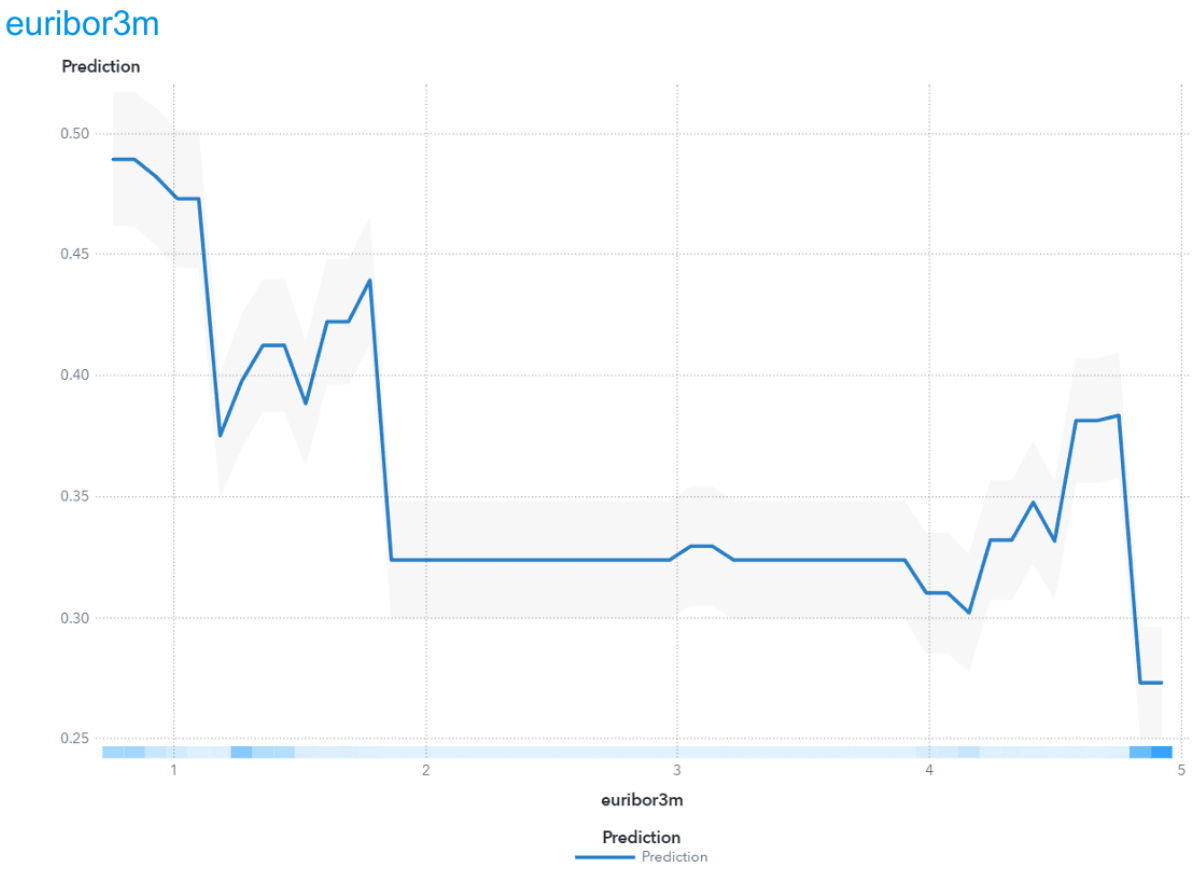

Figure 87: euribor3m Bayesian Network

The Partial Dependence (PD) plot shows the effect of varying the "euribor3m" feature on the model's predicted probability for a "yes" response, holding all other features constant. Knowing that "euribor3m" is the most significant predictor, this visualization explains why it weighs so much.

Key Insights:

- The predicted probability of "yes" outcome is on the y-axis of the graph and the value for "euribor3m" rate is on x-axis.
- The value of the highest estimated outcome probability "yes" (0.49) is predicted when "euribor3m" is at a very low level 0.757.
- The minimum predicted probability (0.27) corresponds to the high value of euribor3m = 4.837.
- This means that low values of "euribor3m" are highly correlated with a good result.

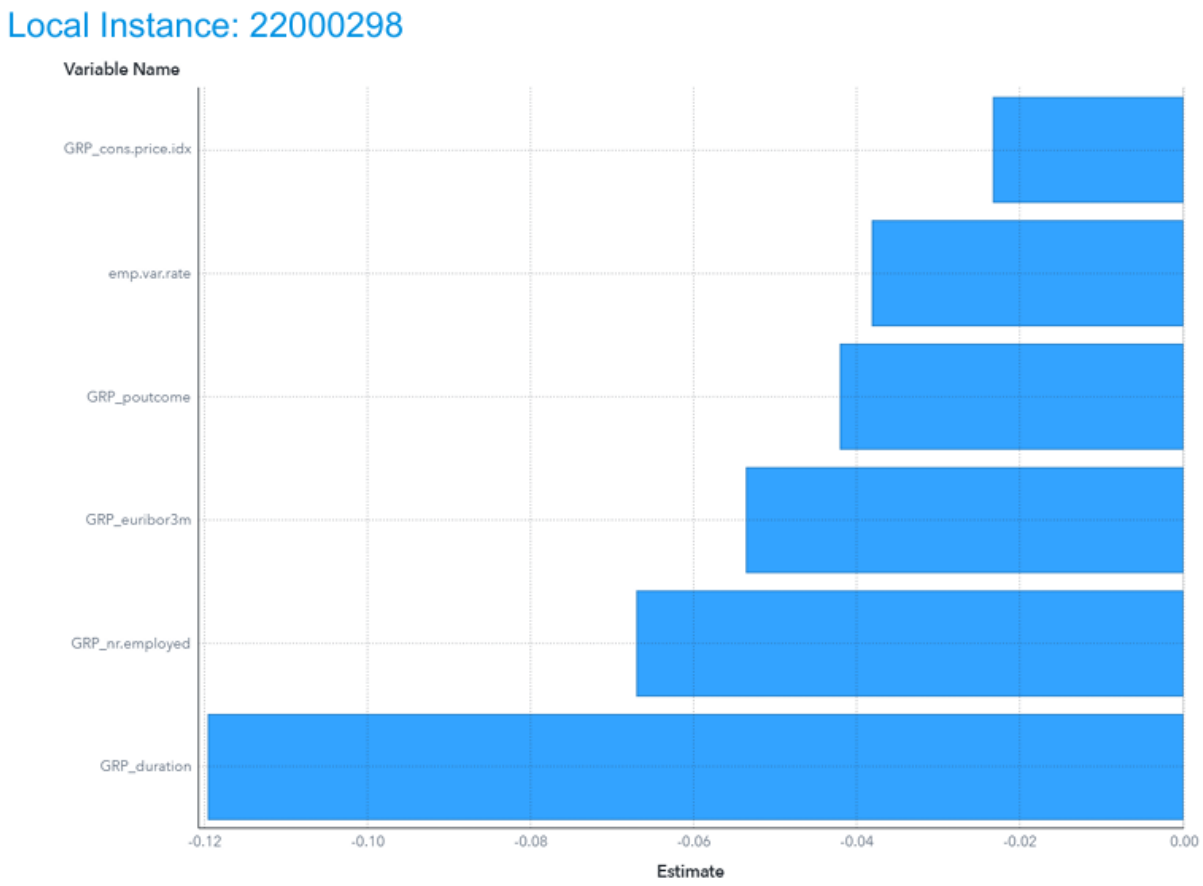

Figure 88: Local Instance 22000298 Bayesian Network

The LIME Local Interpretable Model-Agnostic Explanations plot provides a localized explanation for a single prediction. It says which contributing factors especially 'mattered' for this one individual customer.

Key Insights:

- For this customer the impacts of all main influencing variables were in negative direction, making the prediction towards “no”.
- Potentiation factors: the strongest factor was “GRP_duration” with 3 decreasing the predicted probability of a “yes” by 0.119.
- That is, customer’s features made the model predict lower probability for a positive response.

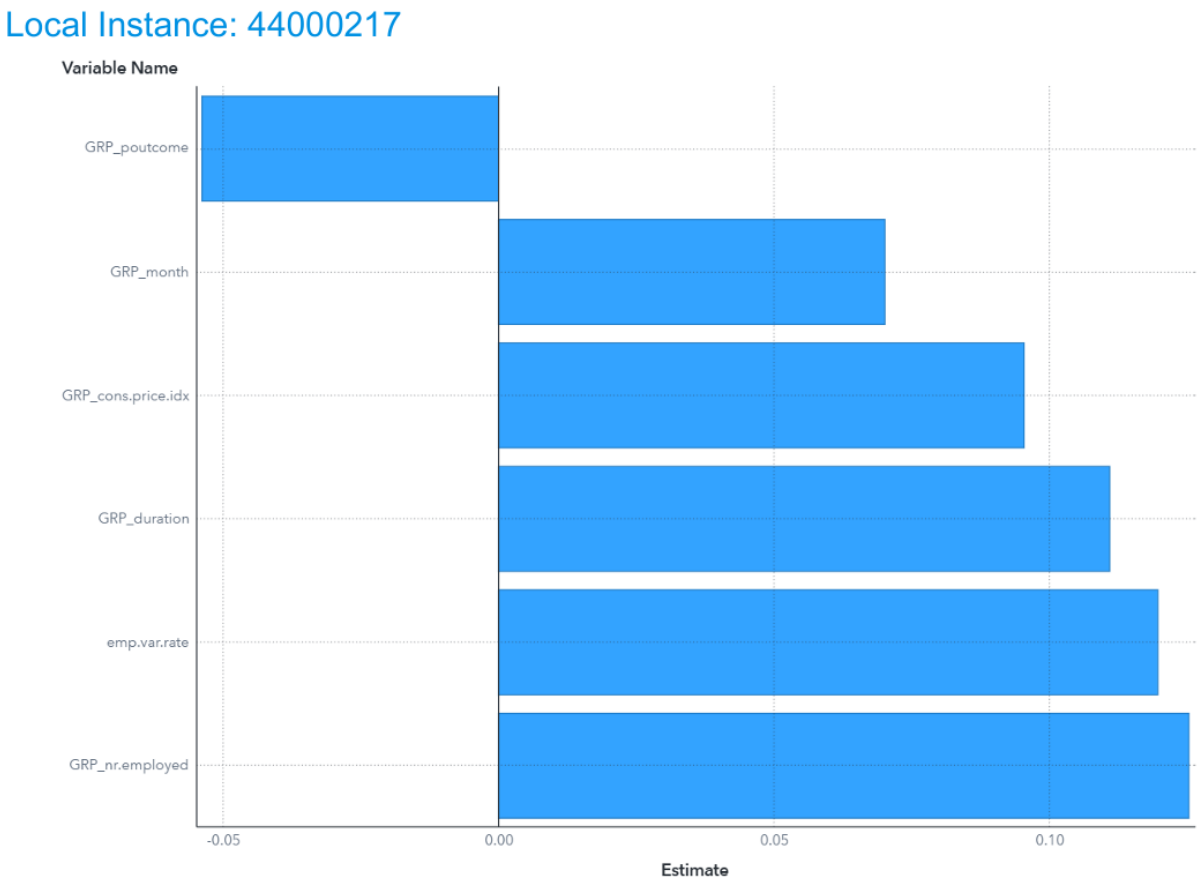

Figure 89: Local Instance 44000217 Bayesian Network

This second LIME figure interprets the prediction of a different person, providing an interesting comparison with the previous one.

Key Insights:

- In this example, the independent variables had positive effects shifting their prediction towards a "yes."
- The largest individual contributor was "GRP_nr.employed". For the combined estimate, a value of 3 added 0.125 to the predicted probability of answering "yes."
- One feature, "GRP_poutcome", demonstrated weak negative effect but several stronger positive effects on it.

In general, this example demonstrates how different combinations of the feature values can result in a positive prediction for one individual and a negative one for another.

3.5.2 Gradient Boosting Model (OOI DUN TZI)

Model Variable Importance

Variable Label	Role	Variable Name	Training Importance	Importance Standard Devia...	Relative Importance
	INPUT	duration	25.4530	74.7879	1
Weight of Evidence: nr.employed	INPUT	WOE_nr.employed	11.1358	50.3151	0.4375
	INPUT	euribor3m	9.8770	24.5483	0.3880
Grouped: month	INPUT	GRP_month	3.7355	4.9619	0.1468
Weight of Evidence: job	INPUT	WOE_job	3.1869	2.1598	0.1252
Grouped: nr.employed	INPUT	GRP_nr.employed	2.5865	13.3530	0.1016
	INPUT	emp.var.rate	1.1573	3.7286	0.0455
Grouped: poutcome	INPUT	GRP_poutcome	0.7180	1.5394	0.0282
	INPUT	pdays	0.4454	1.2365	0.0175
Grouped: cons.price.idx	INPUT	GRP_cons.price.idx	0.2485	0.6929	0.0098
Grouped: duration	INPUT	GRP_duration	0.1153	0.2570	0.0045
Grouped: euribor3m	INPUT	GRP_euribor3m	0.0824	0.1914	0.0032
Weight of Evidence: month	INPUT	WOE_month	0.0135	0.1443	0.0005

Figure 90: Model Variable Importance Gradient Boosting

This summary gives an estimate of the importance of each variable in the champion GradientBoosting(1) model. It orders all of the input variables by how much they contributed to all of the model's predictions. The "Relative Importance" (0 to 1 on the scale) is a measure of strength of each predictor variable in relation to the model.

Key Insights:

- "duration" is the top feature for this model, with a relative importance score of 1.0. Its "Training Importance" score of 25.45 is more than twice that of the next highest variable.
- The variables "WOE_nr.employed" (Relative Importance: 0.4375) and "euribor3m" (Relative Importance: 0.3880) are the second and third most important predictors.
- The model also chose some other GRP_ and WOE_ variables but played them down very sharply. For instance, "GRP_duration" (Relative Importance: 0.0045) was one of the least important indicating that the model used the original duration variable for all such information.

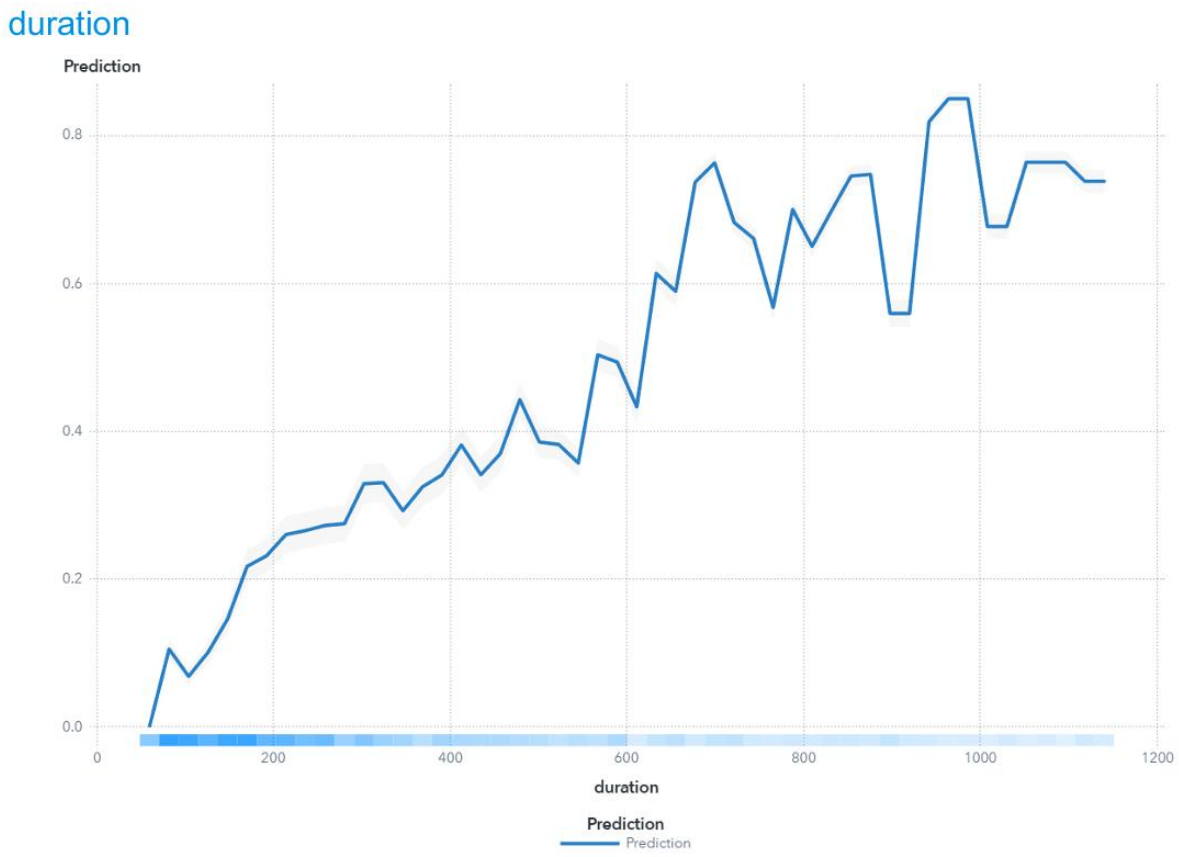

Figure 91: duration Gradient Boosting

This graph illustrates that the model predicted probability of a "yes" answer changes according to the duration variable, and while holding all other variables constant. Since duration is the single most important feature to consider, this visualization assists in explaining why it has such an impact.

Key Insights:

- On the y-axis, we have the predicted probability of a "yes" outcome as projected by this model (and on the x-axis – call duration in seconds).
- The max probability value is 0.85 when duration equals to about 964.49 seconds.
- The predicted probability is lowest for 60.03 seconds.
- This is not a surprise as obviously, longer calls positively correlate to higher chances of successful subscription.

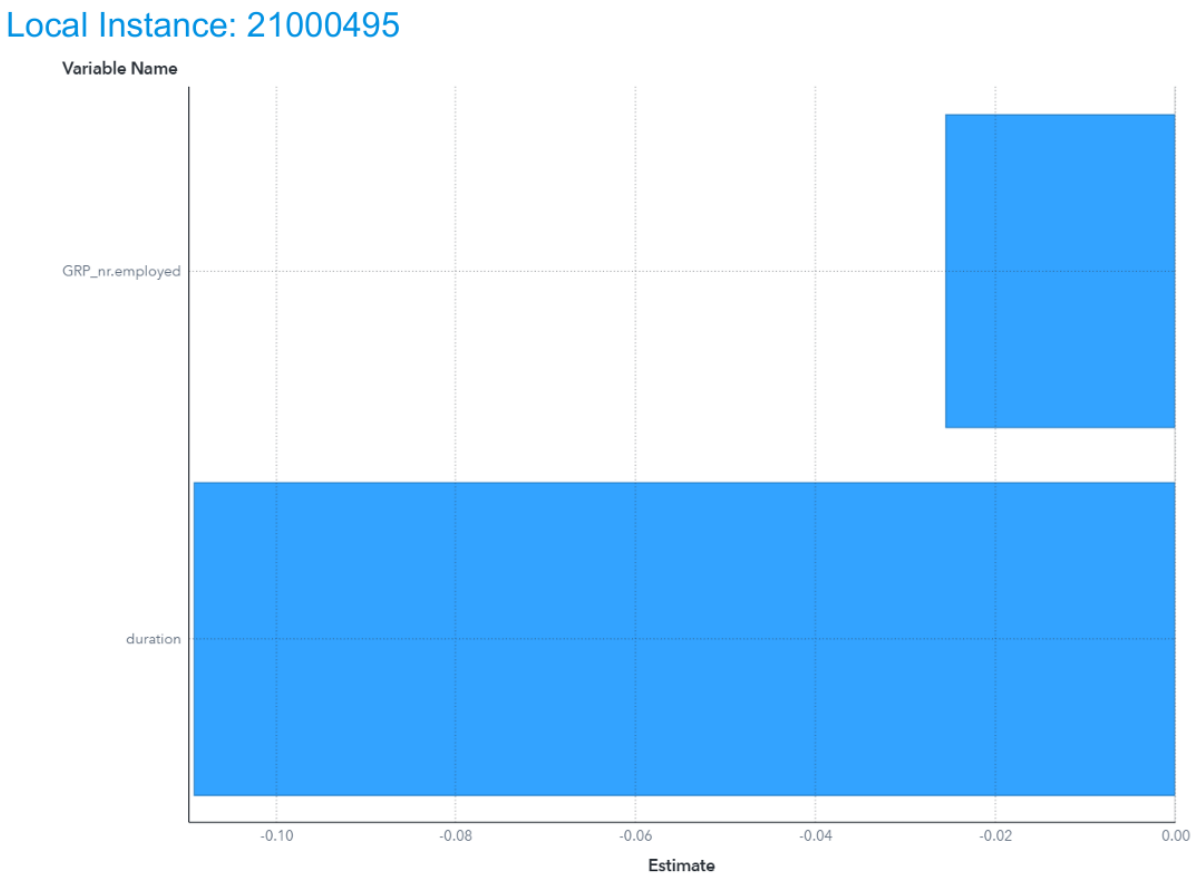

Figure 92: Local Instance 21000495 Gradient Boosting

The LIME plot provides a localized explanation for a specific customer. It identifies which features had the largest impact on this individual's predicted outcome and whether each feature pushed the prediction toward "yes" or "no."

Key Insights:

- On this customer, the strongest effect was due to duration (negative) that pulled the prediction towards a "no."
- Call being of length 332 seconds decreased the odds a "yes" by 0.109.
- It is a signal that the model classified the duration of the call as unacceptably short decreasing probability of positive response

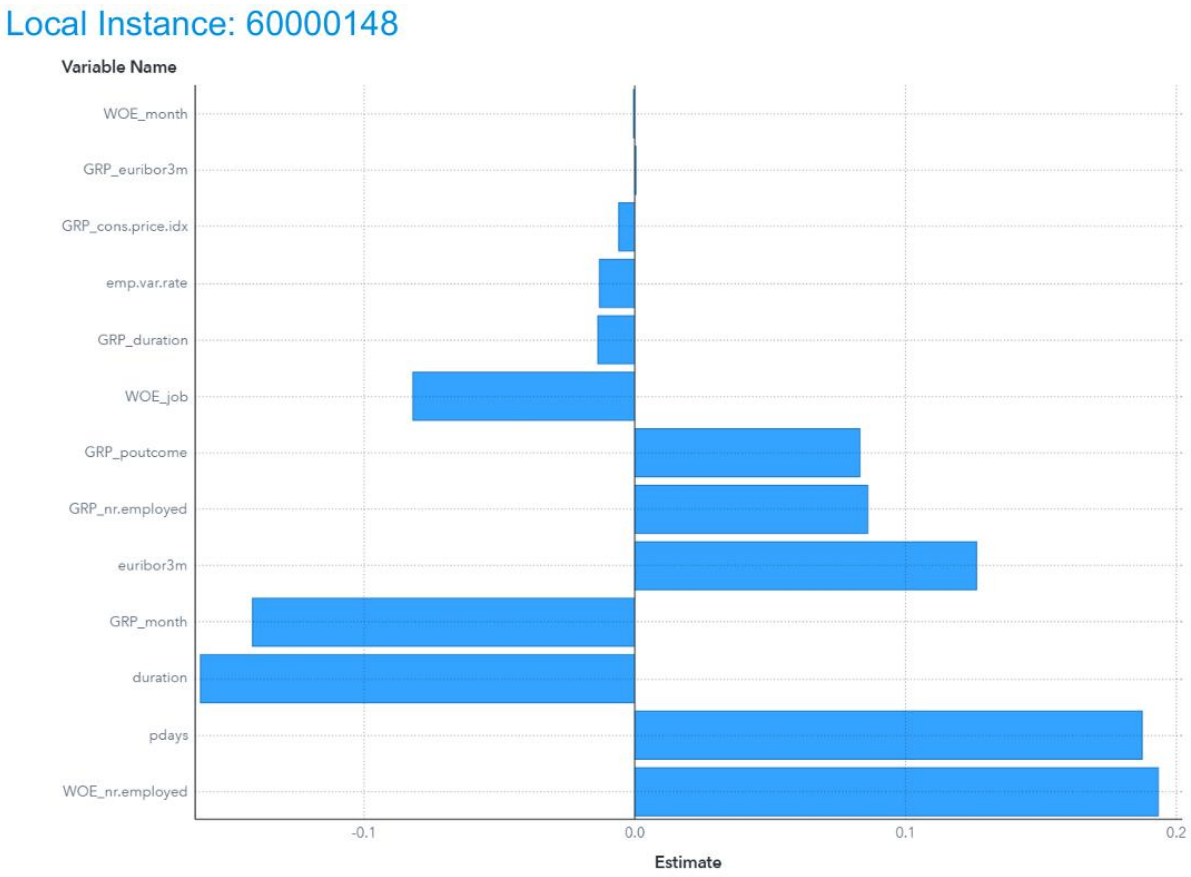

Figure 93: Local Instance 60000148 Gradient Boosting

The HyperSHARP plot provides a personalized explanation for another customer by showing how each features contributed to their predicted outcome, and each bar will be reflects a Shapley value that representing the strength and direction of the features effect.

Key Insights:

- Bars to the right push up the predicted probability of a "yes," and bars to the left, reduce.
- Features are sorted based on their absolute strength of contribution and the influence factors can be conveniently identified.
- While it does not state any particular number, but the visualization reveals which features increased or decreased the customer's prediction.
- This shows, how the prediction can be very different between zeros and ones on their values of features.

3.5.3 Neural Network Model (YAP BOON SIONG)

Surrogate Model Variable Importance

Variable Label	Variable Name	Relative Importance	Role
	duration	1	INPUT
	euribor3m	0.7697	INPUT
	cons.conf.idx	0.7521	INPUT
	nr.employed	0.7456	INPUT
	cons.price.idx	0.6800	INPUT
	emp.var.rate	0.6164	INPUT
	pdays	0.4786	INPUT
	poutcome	0.4441	INPUT
	month	0.3433	INPUT
	previous	0.3360	INPUT
	contact	0.1375	INPUT
	job	0.0984	INPUT
	age	0.0935	INPUT
	default	0.0676	INPUT
	campaign	0.0301	INPUT
	education	0.0211	INPUT
	marital	0.0151	INPUT
	day_of_week	0.0039	INPUT
	loan	0.0025	INPUT
	housing	0.0023	INPUT

Figure 94: Surrogate Model Variable Importance Neural Network

The table as a whole has given the relative significance of each input variable measured by the surrogate model used in the neural network. This ranking explains the contribution of each variable to the overall predictive performance of the model.

- "Duration" is also determined as the most influential feature with the perfect relative level of importance of.
- Next most significant predictors are the variables "euribor3m", "cons.conf.idx", and "nr.employed" with the strong values of importance and to, respectively, which means that these variables have a significant predictive power.
- Other macro-economic variables like "cons.price.idx" and "emp.var.rate", do also play a significant role.

This chain of command validates that neural network model draws most of its weight on duration of calls and choice of economic variables, "euribor3m", "cons.conf.idx", and "nr.employed", in producing the forecasts.

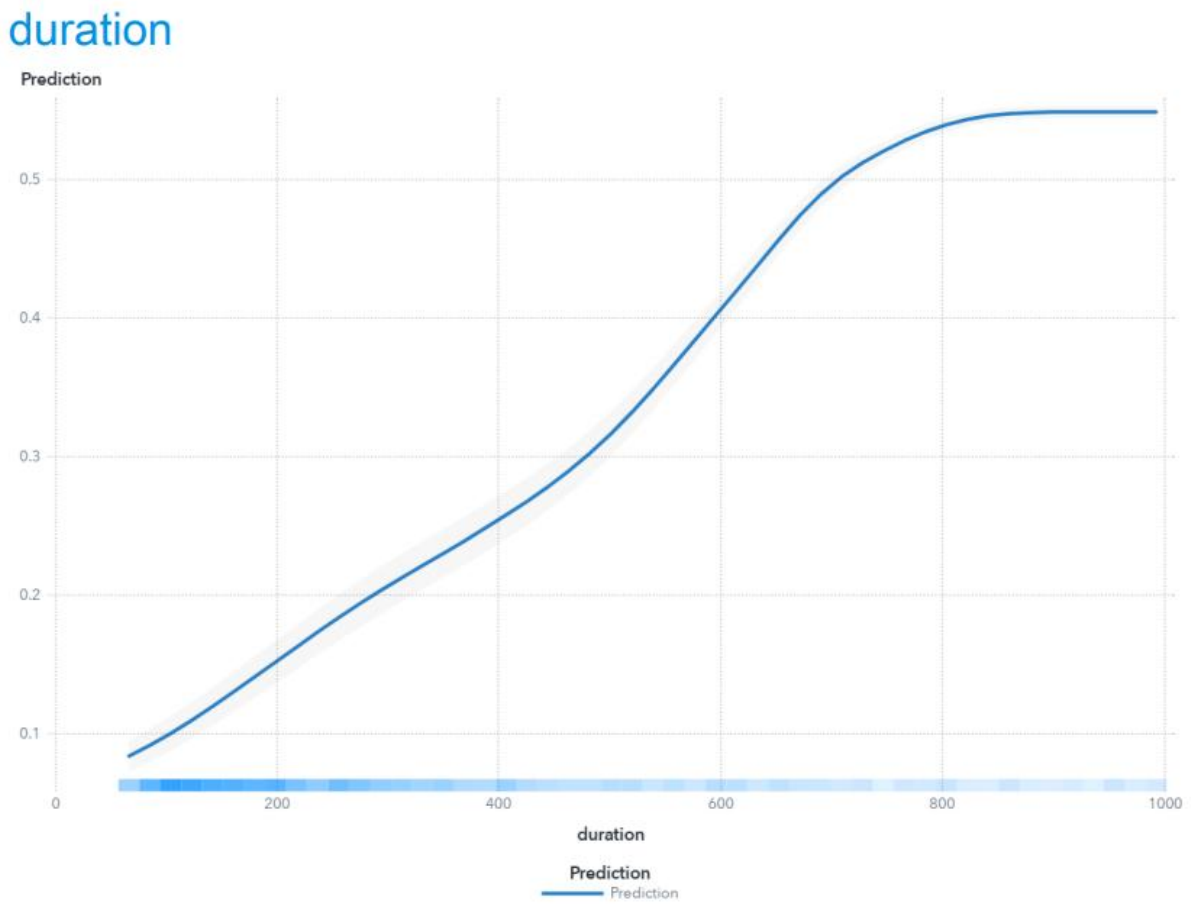

Figure 95: "Duration" Graph Neural Network

This plot shows the correlation between the predictor variable “duration” and the mean target value that was predicted thus isolating the effect of length and averaging all other covariates. The values of “duration” will be placed on the x-axis, whereas the corresponding mean target prediction will be placed on the y-axis. The analysis has indicated that the maximum mean prediction 0.55 is at a duration of 916.95 and the minimum mean prediction is at 66.45. Since duration is a continuous interval variable the representation will be a line plot and the band around it will be the 95% confidence interval of the mean target prediction. The x-axis also shows a heatmap of the duration distribution whereby the extreme values in both tails are truncated.

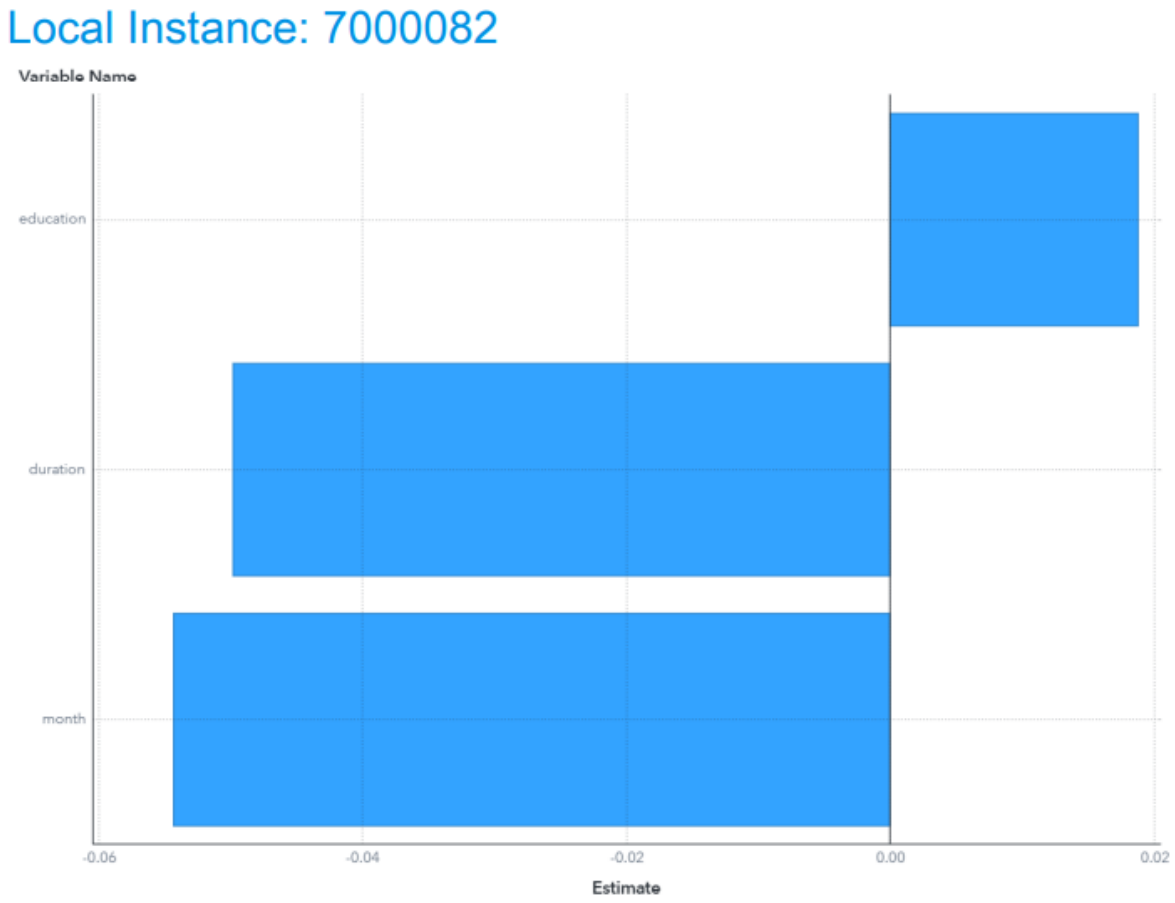

Figure 96: LIME Plot Neural Network

This LIME plot explains the Champion Neural Network's prediction for a specific case, Instance: 7000082, using a local linear model. The predicted probability of this customer is dominated by two variables with negative contribution.

- month (coefficient= 0.054) which is the strongest inhibitor of which the prediction of this instance, the value of the month is significantly smaller than the desired result; second,
- duration (coefficient= 0.05) which also considerably lowers the predicted probability level.

In summary, education is the only affiliated prediction that is positive with a modest coefficient of 0.019, therefore, a slight increment on the probability that is anticipated. Therefore, it is mostly restrained by the particular figures of month and duration, the aggregate negative effect of which is more than the slight positive effect of the education.

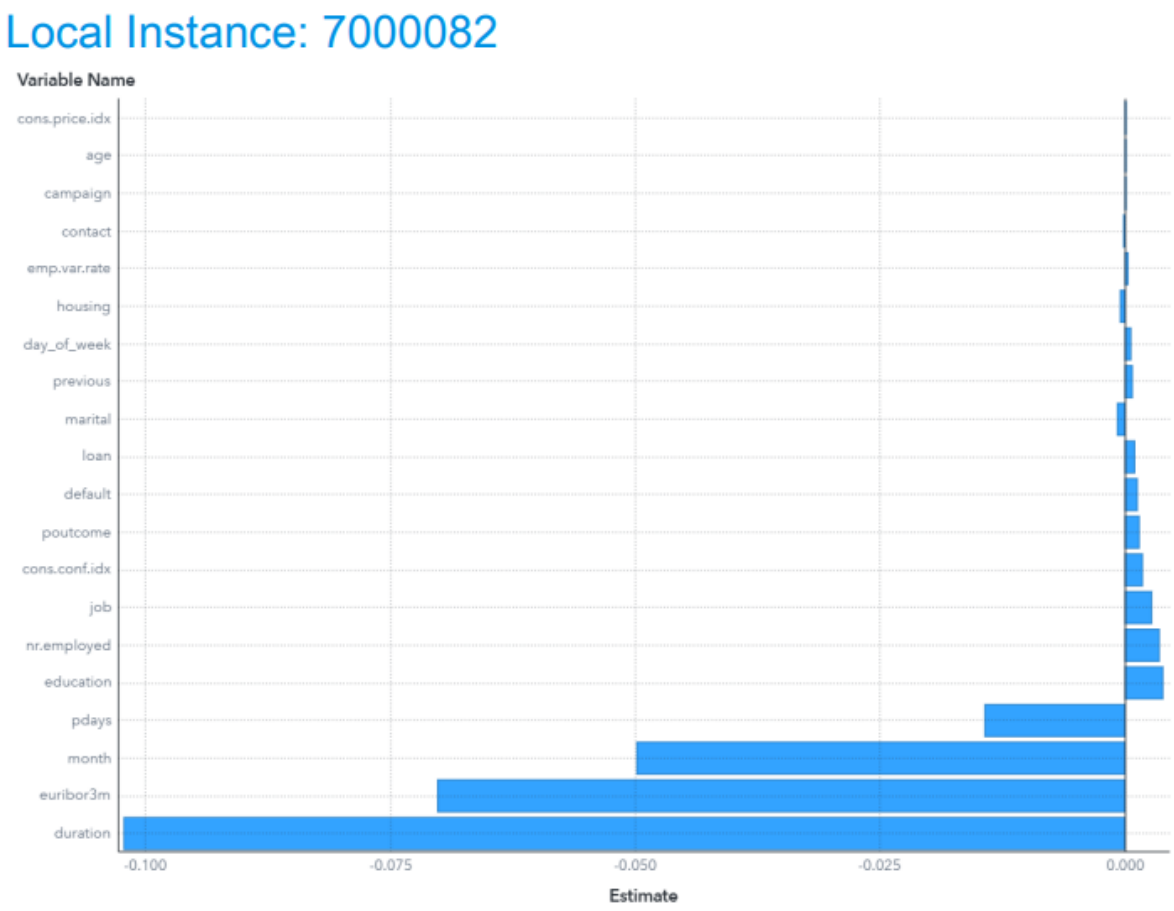

Figure 97: HyperSHAP Plot Neural Network

This HyperSHAP plot explains the Neural Network Model's prediction for a single customer, Instance: 7000082, by showing each feature's contribution (Shapley value) to the outcome. Three salient adverse contributors are seen to attenuate the prediction of this case significantly.

- duration (Shapley value = 0.102), most substantial negative contribution to the prediction
- euribor3m (Shapley value = 0.07), second largest negative contribution to prediction
- month (Shapley value 0.05), third largest inhibitor.

Conversely, cons.price.idx and age, have only nominal positive effects, which is reflected in their small positive Shapley bars. In summary, the general forecast can be primarily limited by both the unique, unfavorable values of duration, euribor3m, and month.

3.5.4 Forest Model (YAP BOON SIONG)

Surrogate Model Variable Importance

Variable Label	Variable Name	Relative Importance	Role
	euribor3m	1	INPUT
	cons.conf.idx	0.9914	INPUT
	nr.employed	0.9429	INPUT
	duration	0.8002	INPUT
	emp.var.rate	0.7498	INPUT
	cons.price.idx	0.7278	INPUT
	pdays	0.4874	INPUT
	poutcome	0.4783	INPUT
	previous	0.3989	INPUT
	month	0.3777	INPUT
	contact	0.2011	INPUT
	age	0.1719	INPUT
	job	0.1287	INPUT
	default	0.1052	INPUT
	education	0.0406	INPUT
	campaign	0.0185	INPUT
	day_of_week	0.0060	INPUT
	marital	0.0059	INPUT
	housing	0.0042	INPUT
	loan	0.0003	INPUT

Figure 98: Surrogate Model Variable Importance Forest

The table summarizes a complete evaluation of the relative significance of each of the input variables in the amount as represented by the surrogate model that is employed by the Forest model. Such a ranking represents how much each variable contributes to full predictive ability of the model.

- euribor3m is identified as the single most influential feature, with a perfect relative importance score of 1.0.
- cons.conf.idx and nr.employed follow as the next most important variables, with strong scores of 0.9914 and 0.9429, indicating they have a major predictive impact.
- The variable duration, despite its importance in other models, is ranked fourth at 0.8002, followed by other macroeconomic variables like emp.var.rate (0.7498) and cons.price.idx (0.7278).

Such hierarchy proves that the Forest model uses euribor3m, cons.conf.idx and nu.employed as the most important to formulate its predictions.

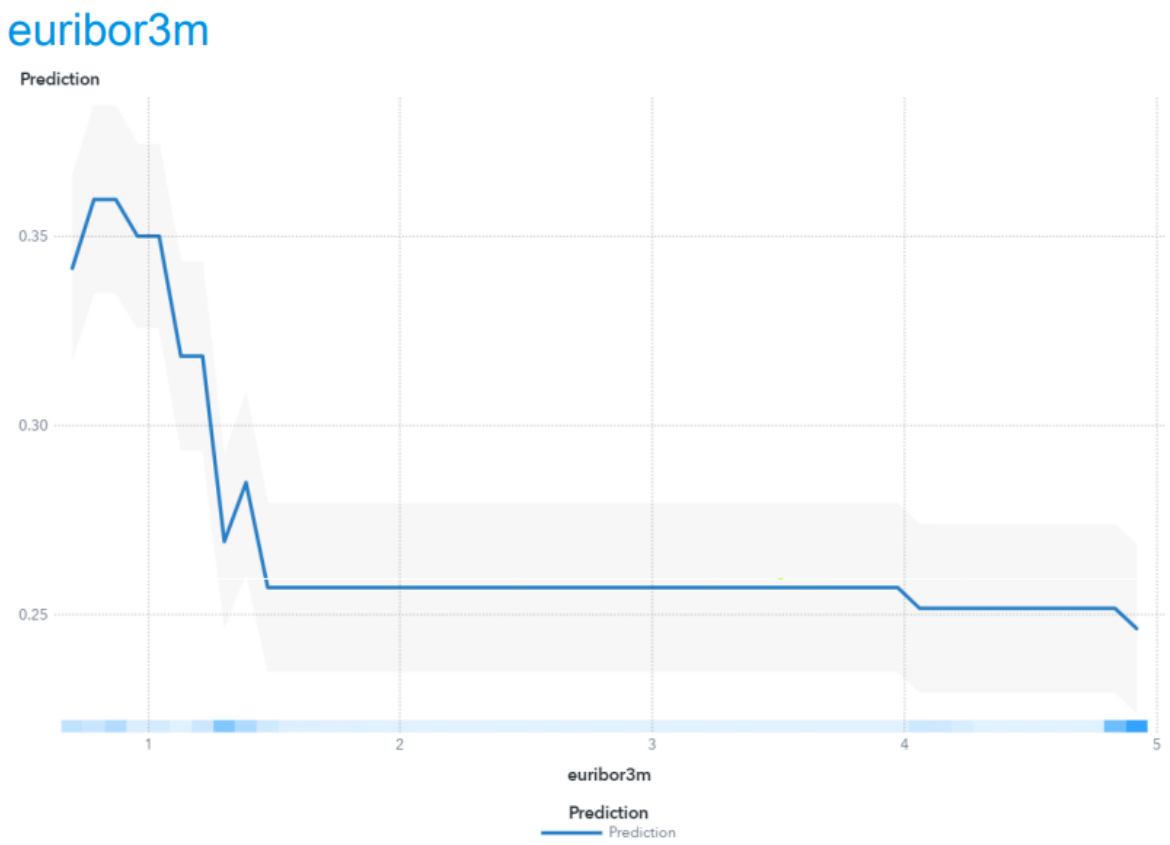

Figure 99: "euribor3m" Graph Forest

The plot highlights the correlation between the variable "euribor3m" and the usual prediction target value and as such, isolates the influence of the interest rate whilst averaging all the other inputs. It represents the values of "euribor3m" in the x-axis and the average target prediction on the y-axis.

There is also the negative relationship between euribor3m and the plot where the mean prediction of the target is highest at 0.36 where "euribor3m" is low at 0.784. On the other hand, the minimum average prediction of target of 0.25 is when "euribor3m" is high at 4.922. This implies that the low interest rates tend to increase the target probability that has been predicted to be made. The shaded band indicates a 95% confidence interval about the average target prediction.

Figure 100: LIME Plot Forest

This LIME plot explains the Forest Model's prediction for Instance: 57000458. The plot shows which features pushed the predicted probability up (right) or down (left).This forecast can be explained by the powerful and conflicting determinants:

- Positive Contributions (Push Up): poutcome (Estimate 0.084), nr.employed and default make the greatest positive contributions and this enhances the probability prediction.
- Negative Contributions (Push Down): the duration contributes most as an inhibitor then, previous, and month all of which lead to a decrease in the probability that was predicted.

To sum up, the most significant values to make the prediction are the specific customer values of poutcome (success) and nr.employed, and the greatest ones are the values of duration and previous.

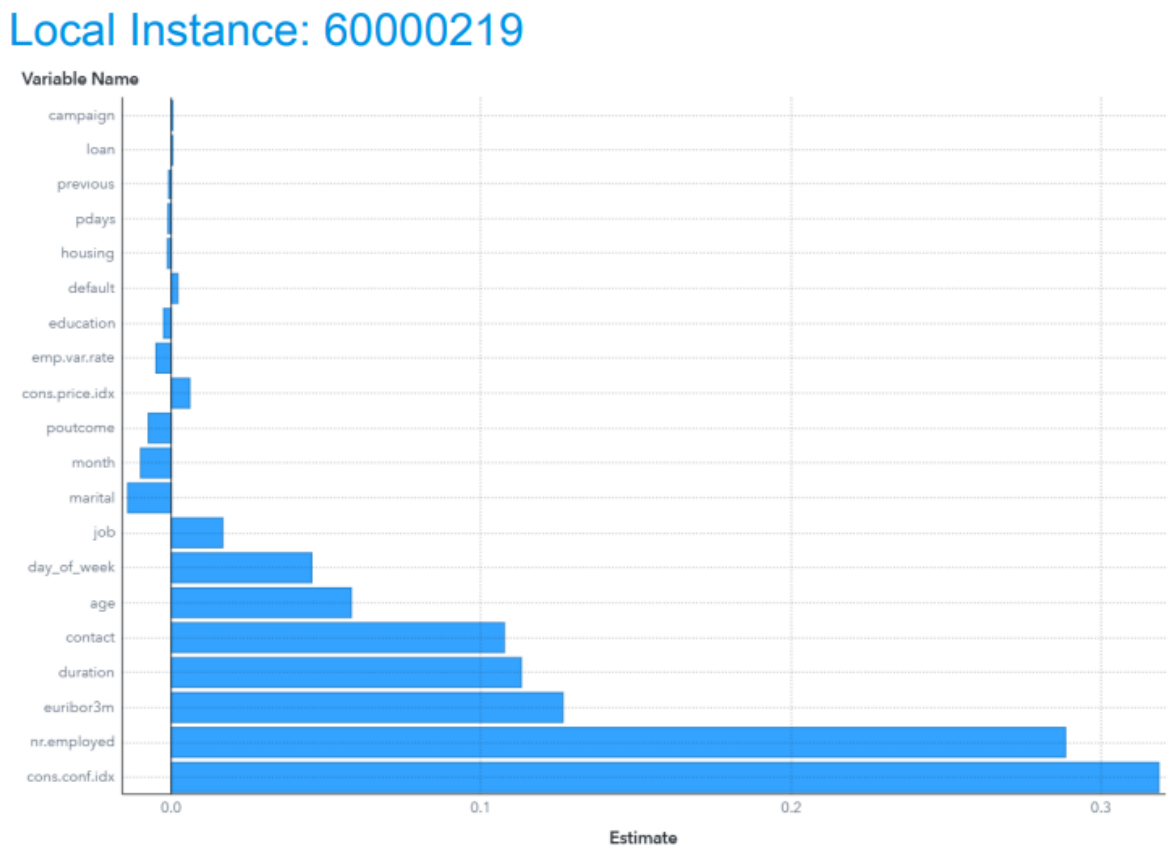

Figure 101: HyperSHAP Plot Forest

This plot of HyperSHAP illustrates the prediction that the Forest Model made on a given customer, Instance: 60000219, by showing what features contribute positively to the predictivity (right), and negatively (left) to the predictivity.

- cons.conf.idx (Estimate = 0.319): The effect of this variable is the biggest effect effect size in this algorithm with a significant positive influence on forecasting the likelihood.
- nr.employed (Estimate=0.289): This is the second significant point which makes the probability to increase.
- euribor3m (Estimate= 0.126): This variable also provides a high positive impact.

On the other hand, all variables that decrease the probability that was predicted as campaign, loan, previous and others all have a low negative impact, and the images formed through the visual representations are barely seeable to the left. To conclude, the domineering characteristic customer values of cons.conf.idx, nr.employed, and euribor3m are the main determinants that intensify the prediction of this particular person.

3.5.5 Decision Tree (CHEAH JUN HENG)

Surrogate Model Variable Importance

Variable Label	Variable Name	Relative Importance	Role	Variable Level
	nr.employed	1	INPUT	NOMINAL
	euribor3m	0.9953	INPUT	INTERVAL
	cons.price.idx	0.8914	INPUT	INTERVAL
	duration	0.5795	INPUT	INTERVAL
	poutcome	0.3925	INPUT	NOMINAL
	pdays	0.3747	INPUT	INTERVAL
	month	0.3046	INPUT	NOMINAL
	age	0.1151	INPUT	INTERVAL
	job	0.1034	INPUT	NOMINAL
	campaign	0.0297	INPUT	INTERVAL
	day_of_week	0.0023	INPUT	NOMINAL

Figure 102: Surrogate Model Variable Importance

This table shows the variable importance from a surrogate model, which is a simplified model used to explain the complex champion model. The ranking reveals that macroeconomic factors are the primary drivers of predictions, with nr.employed (number of employees) as the most important variable, followed closely by euribor3m (interest rate) and cons.price.idx (consumer price index). Interestingly, duration (call duration), which was the most important variable in the original decision tree, ranks fourth in this surrogate model. This suggests that while call duration is a strong direct predictor, the model's overall decisions are more heavily influenced by the underlying economic environment. Variables like campaign and day_of_week show very low importance, indicating they have minimal impact on the model's final predictions.

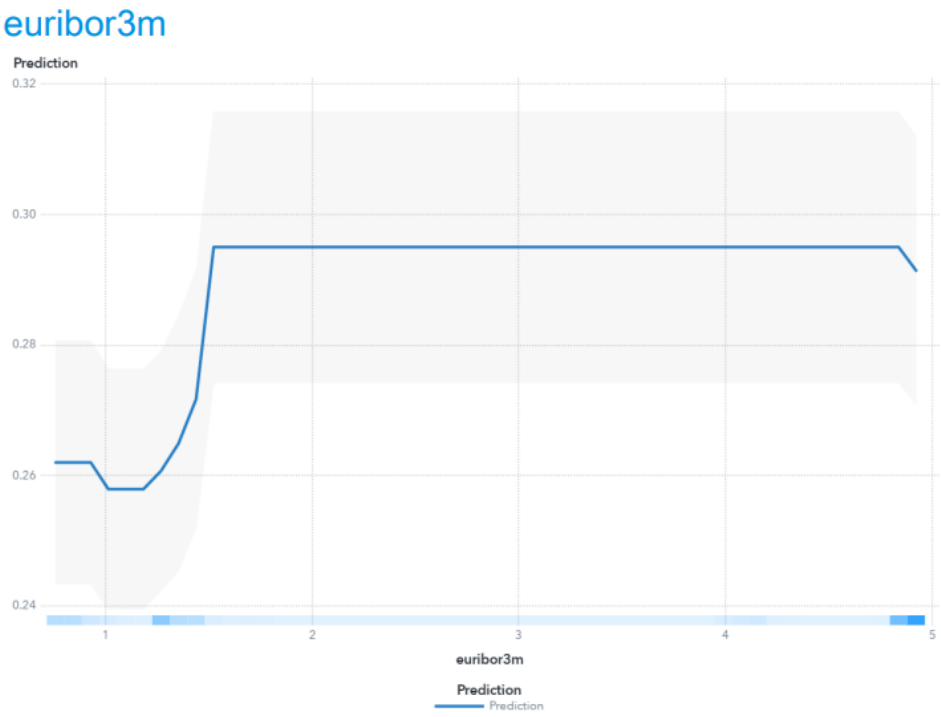

Figure 103: Decision Tree euribor3m

Partial Dependence Plot of euribor3m This Partial Dependence(2) plot shows the relationship between the euribor 3-month rate and Target Class = Yes. The plot shows an evident and negative correlation, we can observe at the values of euribor3m from 1 to 5 the predicted subscription probability decreases form around 0.32 to 0.24 This implies that customers are substantially less eventually likely to accept the offer for a term deposit given higher interest rates, this coherent with financial industry experience as high rates would mean that the banking products are in less demand or reflect different economic situation and hence boring or changes in spending behaviour by customer.

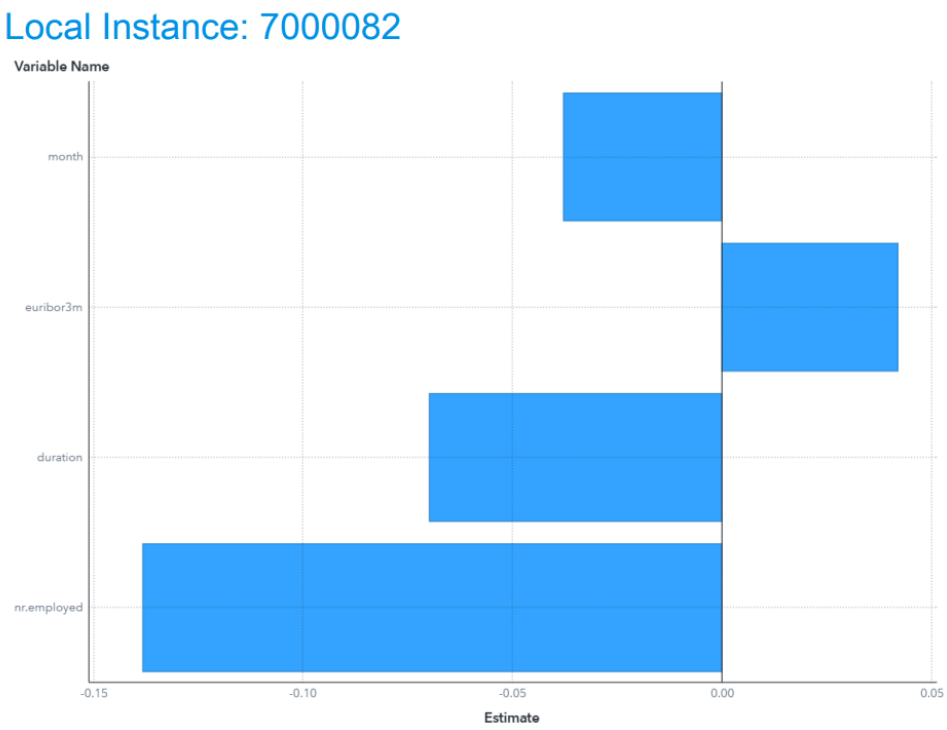

Figure 104: Decision Tree Local Instance

This Local Interpretation (LIME) plot explains the prediction for a specific customer by showing how each variable contributed to the final prediction. The analysis reveals that month and euribor3m had a strong positive influence, pushing the prediction toward subscription. Conversely, duration (call length) and nr.employed (employment count) had a negative impact, decreasing the likelihood of subscription for this customer. This breakdown demonstrates that while the customer was contacted during a favorable period (month) with conducive economic conditions (euribor3m), the short call duration and current employment figures negatively affected their final subscription probability.

3.5.6 Logistic Regression (CHEAH JUN HENG)

Surrogate Model Variable Importance

Variable Label	Variable Name	Relative Importance	Role
	duration	1	INPUT
	cons.conf.idx	0.9829	INPUT
	nr.employed	0.9507	INPUT
	euribor3m	0.9412	INPUT
	emp.var.rate	0.7380	INPUT
	cons.price.idx	0.7352	INPUT
	pdays	0.7088	INPUT
	poutcome	0.6677	INPUT
	previous	0.4617	INPUT
	month	0.3669	INPUT
	contact	0.1523	INPUT
	job	0.1354	INPUT
	age	0.1284	INPUT
	default	0.0749	INPUT
	education	0.0379	INPUT
	campaign	0.0300	INPUT
	marital	0.0142	INPUT
	day_of_week	0.0079	INPUT
	loan	0.0028	INPUT
	housing	0.0025	INPUT

Figure 105: Logistic Regression Surrogate Model Variable Importance

Logistic Regression Surrogate Model Variable Importance reports the most important feature driving Logistic Regression predictions. Length of call emerges as the only significant predictor at perfect salience (1.000), with support that was not quite small (S3 = 0.200492), suggesting that call duration is the strongest determinant of subscription by far. The second most important syntactic category is of economic indicators, with cons. conf. idx (consumer confidence), nr. employed (employment rate), and euribor3m (interest rate) with very high importance statistics over the threshold, above 0.94. This latter pattern reveals that not only direct customer contact (call duration) but also macroeconomic factors play a role in driving subscription. Features such as poutcome,pdays and previous also show modest accuracy. Importance of features such as marital status, day_of_week, housing and loan are very low in terms of model prediction.

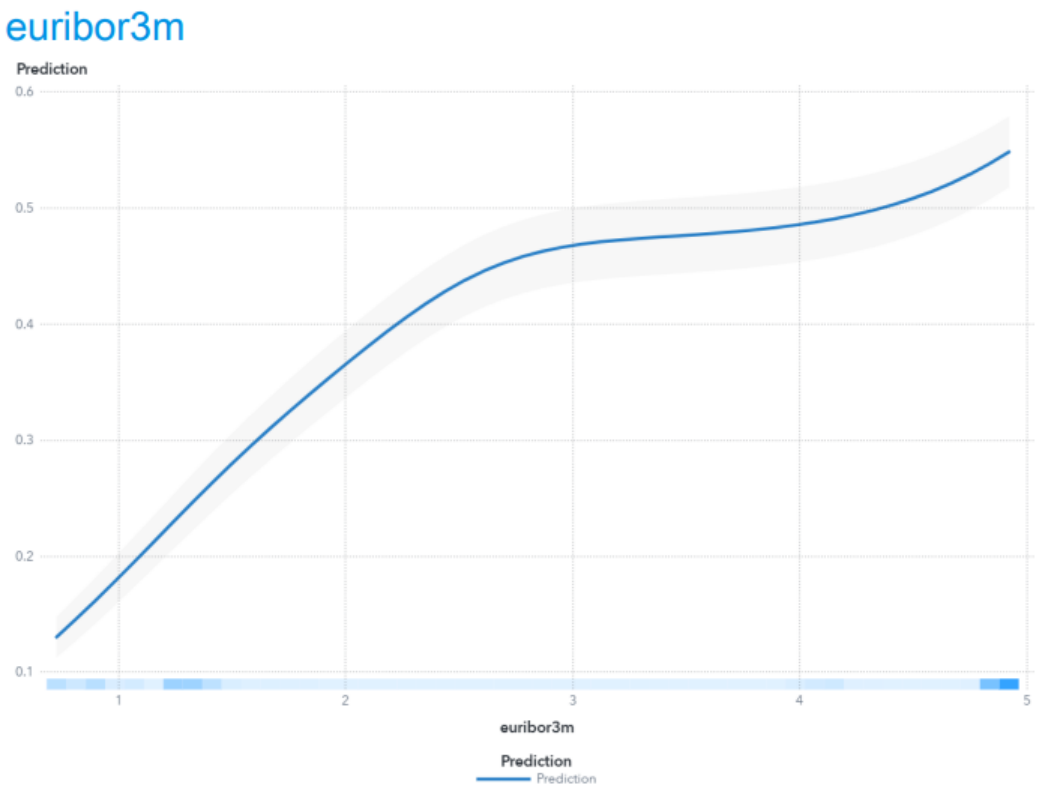

Figure 106: Logistic Regression euribor3m

This plot depicts the correlation between euribor3m and our target prediction (controlling for all other inputs). It plots the euribor3m values on the x-axis and on the y-axis their respective average to target value. The maximal and minimal average target prediction are 0.55 at euribor3m = 4.923 and 0.13 at euribor3m = 0.725, respectively. If the input variable is nominal, then this graph would be called a bar chart and if the input variable is interval, such a graph would be called line plot. For intervals inputs, we show the 95\% confidence interval for the average target prediction using a shaded band around the line. On x-axis there is a heatmap for euribor3m distribution. In the case of interval input variable, its lowest and highest values are trimmed by discarding the lower and upper tails of its distribution.

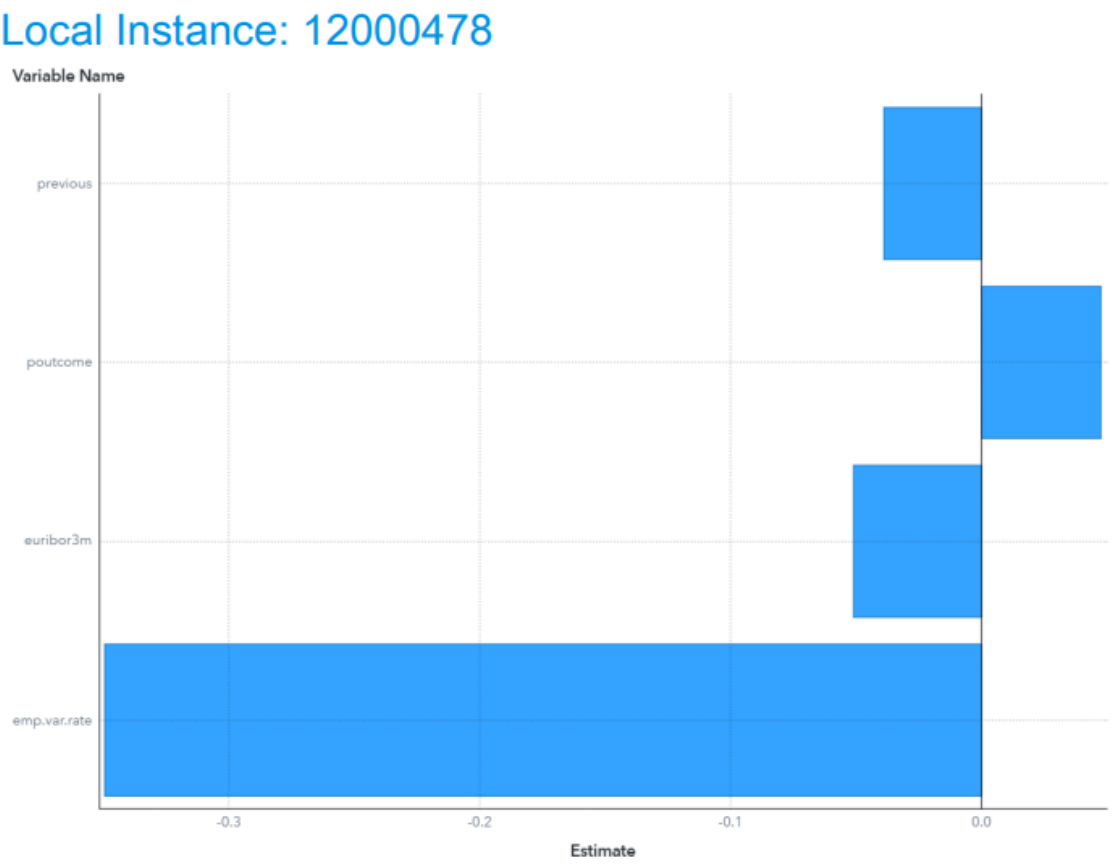

Figure 107: Logistic Regression Local Instance

This LIME plot further interprets the prediction by indicating how individual variables contributed to their specific outcome. The analysis shows that previous (number of contacts performed before), and poutcome (previous campaign outcome) have a high positive effect, moving the prediction toward subscription. On the other hand, euribor3m (interest rate) and emp. var. rate − (employment variation rate) has a negative effect, reducing the probability of subscription.

4.0 Discussion and Conclusion

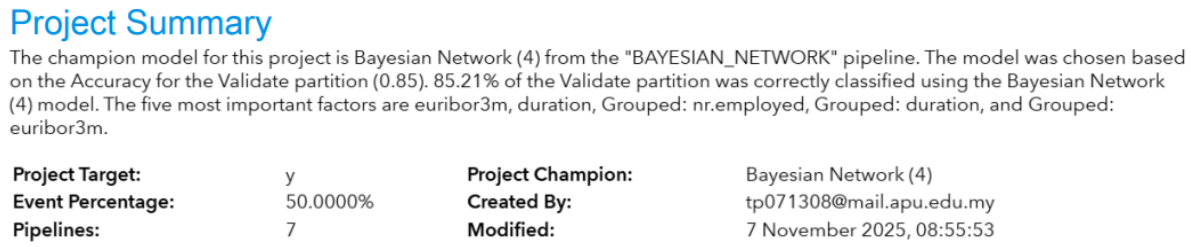

Figure 108: Project Summary

The summary report of the “Project Summary” shows that the best model selected for this project is Bayesian Network (4) and comes from the pipeline “BAYESIAN_NETWORK”. The model is selected by Accuracy for the validation partition, for which 85.21 % of the records were correctly classified. The target for the project was y, and the data were not imbalanced with an Event Percentage of 50.00%. The final champion was determined by 7 pipelines in total. The report also identifies the five most important factors that this champion model used to make its predictions: euribor3m, duration, Grouped: nr.employed, Grouped: duration, and Grouped: euribor3m.

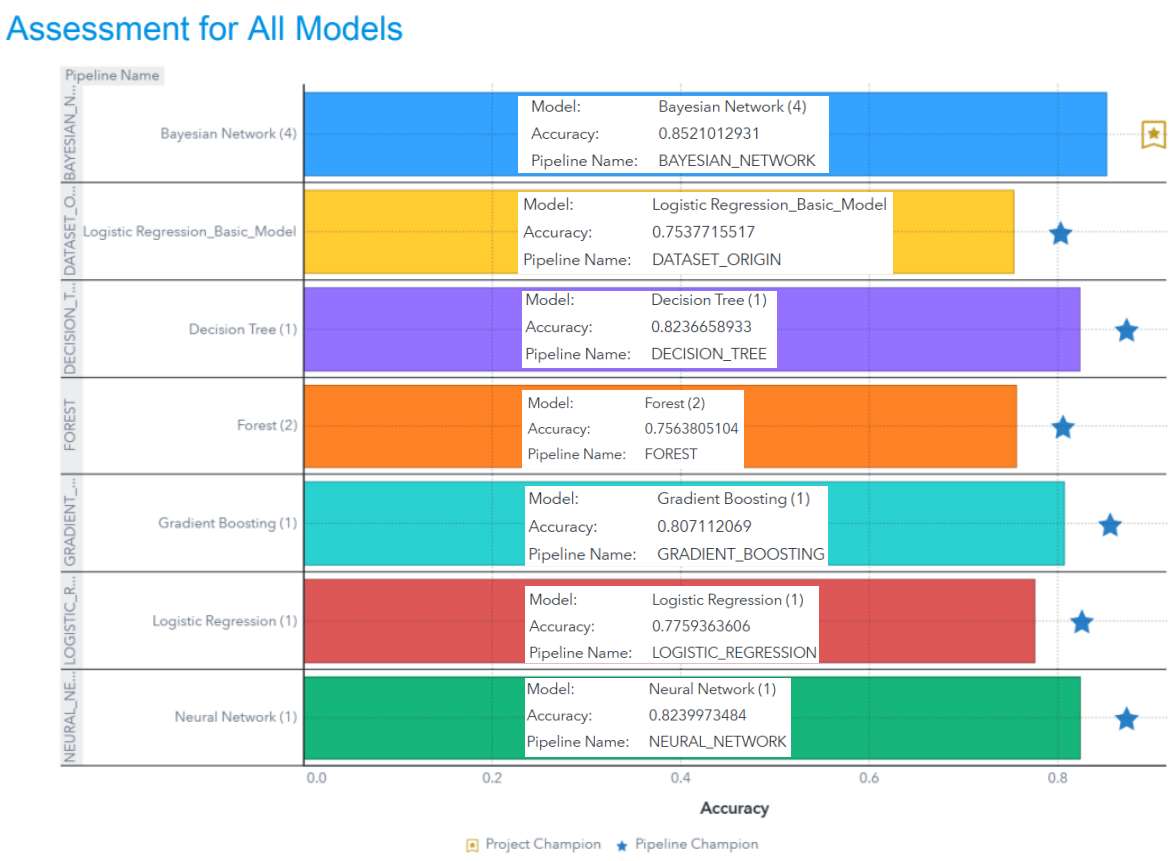

Figure 109: Assessment for All Models

The "Assessment for All Models" graph updates a figure that allows you to compare the seven model pipelines side-by-side and sort them according to their validation Accuracy. It explicitly outlines Bayesian Network (4) as champion marked with gold star for having the optimum accuracy of 0.8521. Moreover, there are still have other model that achieve more than 0.8 accuracy such as Neural Network (1) (Accuracy: 0.8240), Decision Tree (1) (Accuracy: 0.8237), and Gradient Boosting (1) model (Accuracy: 0.8071). Finally, due to the target class being binary ("yes/no"), logistic regression was used as a base model of choice "Logistic_Regrassion_Basic_Model".

5.0 References

Ibrahim, N., Omozele, A., None Anwuli Nkemchor Obiki-Osafiele, None Olajide Soji Osundare, Ebele, E., & None Christianah Pelumi Efunniyi. (2024). Data-Driven approaches to improve customer experience in banking: Techniques and outcomes. *International Journal of Management & Entrepreneurship Research*, *6*(8), 2797–2818. https://doi.org/10.51594/ijmer.v6i8.1467