CREDIT SCORING FINAL PROJECT

June 3, 2016

PHAN Trung Hoai Minh

trunghoaiminh.phan@ieseg.fr

Table of Contents

Executive Summary

Over the past 40 year history of finance and banking, credit scoring is considered as one of the most successful applications of statistical and operations research in this industry. Using a scorecard built from a number of characteristics of borrowers (e.g. age, marital status, housing, borrowing purpose, etc.) and a set of decision models (e.g. logistic regression, ensemble, decision tree, etc.), the final score can guide the lenders (the banks) in the providing of borrower credit. Credit scoring technique is fast, stable and high accuracy.

The purpose of this project is to use SAS Enterprise Miner to build a prototype of application scorecard to score the borrowers who apply for credit in the first time. The final score ranks from 0 to 600, higher score means better, the cutoff is set at 525.

This report is going to interpret the final result of the mentioned above scorecard and guide the reader step by step to re-build the credit scoring model. There are some summary numbers to remember, as follow:

- There are 8 variables in the final model
- The AUR = .87 (for Training set) and AUR = .83 (for Validation set)
- At the cutoff score 525, the model cover 63.4% of population

Part I: Application Credit Scorecard

The application scorecard

Scorecard

		Group	Scorecard Points	Weight of Evidence	Event Rate GOOD_BAD = 1	Percentage of Population	Coefficient
Age in years	AGE< 26	1.00	54	-0.34	36.88	17.98	-1.01
	26<= AGE< 28	2.00	77	0.45	20.81	9.63	-1.01
	28<= AGE< 32	3.00	51	-0.44	39.08	16.24	-1.01
	32<= AGE< 42, _MISSING_	4.00	75	0.37	22.32	29.08	-1.01
	42<= AGE	5.00	65	0.02	28.81	27.07	-1.01
Credit amount	AMOUNT< 1352	1.00	58	-0.25	34.75	24.95	-0.82
	1352<= AMOUNT< 1829	2.00	80	0.69	17.26	15.00	-0.82
	1829<= AMOUNT< 3915, _MISSING_	3.00	74	0.41	21.51	34.95	-0.82
	3915<= AMOUNT< 8978	4.00	57	-0.30	35.85	19.98	-0.82
	8978<= AMOUNT	5.00	28	-1.50	65.05	5.12	-0.82
Status of existing checking account	A11	1.00	41	-1.00	53.03	26.05	-0.80
	A12	2.00	52	-0.53	41.19	26.56	-0.80
	A13	3.00	78	0.59	18.63	6.47	-0.80
	A14, _MISSING_	4.00	100	1.55	8.10	40.92	-0.80
Duration in month	DURATION< 9	1.00	96	1.53	8.19	10.60	-0.73
	9<= DURATION< 12	2.00	72	0.36	22.35	8.01	-0.73
	12<= DURATION< 28, _MISSING_	3.00	64	-0.02	29.73	61.13	-0.73
	28<= DURATION< 48	4.00	57	-0.32	36.23	14.68	-0.73
	48<= DURATION	5.00	40	-1.13	56.05	5.57	-0.73
Credit history	A30, A31	1.00	25	-1.63	67.86	8.39	-0.84
	A33	2.00	60	-0.17	32.91	8.30	-0.84
	A32, _MISSING_, _UNKNOWN_	3.00	63	-0.02	29.76	54.40	-0.84
	A34	4.00	83	0.77	16.13	28.92	-0.84
Property	A124	1.00	54	-0.66	44.47	16.19	-0.53
	A122	2.00	60	-0.24	34.37	22.34	-0.53
	A123, _MISSING_, _UNKNOWN_	3.00	66	0.10	27.15	32.78	-0.53
	A121	4.00	73	0.56	19.16	28.69	-0.53
Purpose	A41	1.00	88	0.93	14.06	11.22	-0.91
	A43	2.00	86	0.82	15.37	28.26	-0.91
	A42	3.00	62	-0.06	30.60	17.96	-0.91
	A44, A49	4.00	57	-0.26	34.96	9.79	-0.91
	A40, A410, A45, A46, A48, _MISSING_, _UNKNOWN_	5.00	47	-0.64	44.05	32.77	-0.91
Savings account / bonds	A61, _MISSING_	1.00	56	-0.29	35.71	60.07	-0.91
	A62	2.00	54	-0.38	37.81	10.69	-0.91
	A63	3.00	93	1.11	11.95	7.27	-0.91
	A64	4.00	99	1.32	10.00	4.96	-0.91
	A65	5.00	88	0.92	14.18	17.00	-0.91

The scorecard was built to evaluate borrowers' applications at the first time they apply for credit (application scorecard). The giving score rank from **0** to **600** (higher point is better), cutoff is at **525** points, covering **63.4%** of populations (Step 6, Gain Table, page 15). Below are some statistical description of the scorecard:

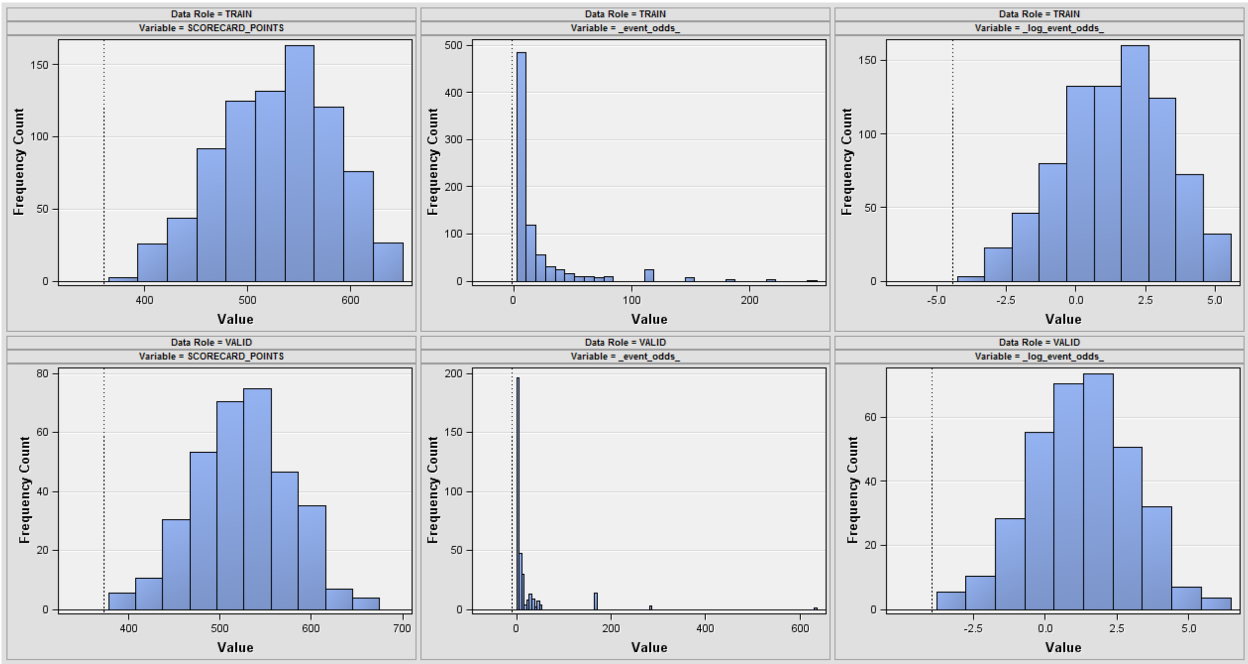


Figure 1: Score Distribution

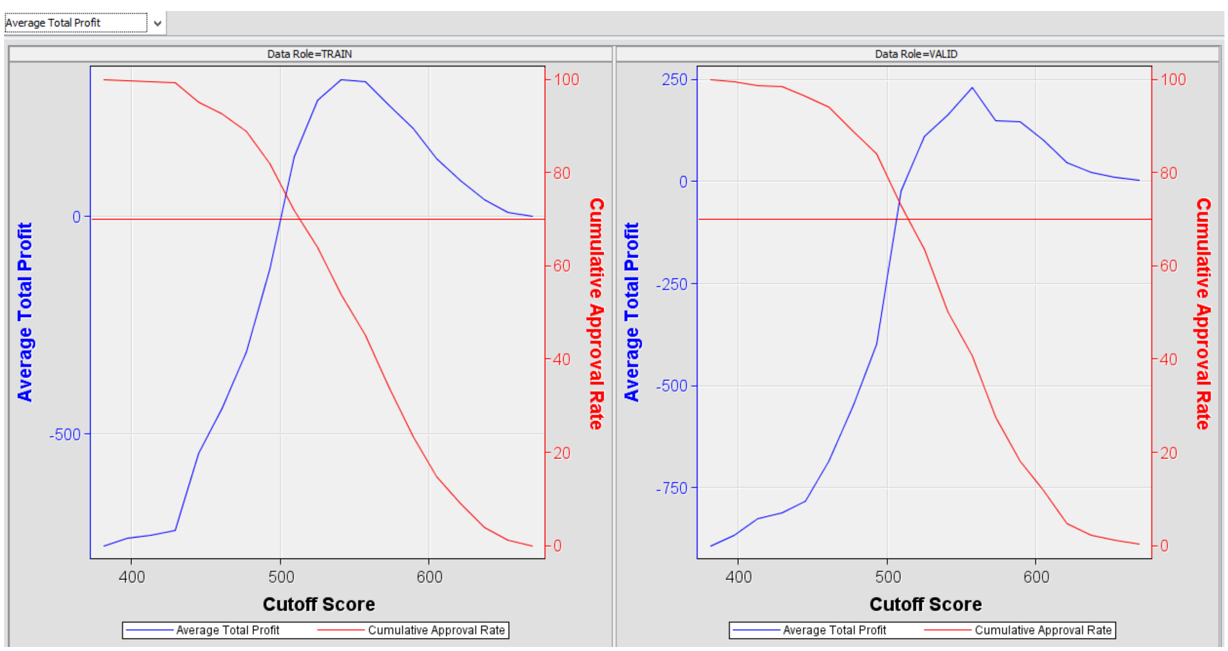


Figure 2: Trade-off Plots – Average Total Profit vs. Cutoff Score

Scorecard interpretation: “higher score is better”

There are **8 variables** have been selected in the scorecard, each variable has been separated into smaller groups (using tree method), each small group has its point considering as the importance of that group to the final credit score.

Variable	Group	Point	Interpretation
AGE (Age in years)	AGE< 26	54	• In this variable group, applicants who are between 26 and 28 years old and between 32 and 42 years old have the highest score (means lowest credit risk). The reason could be people in this group are already settle down with their job or family for a certain period of time; • The group of 28-32 years old has the lowest score, at these ages, people seem to have some significant financial problems or life changing (e.g. new job, start-up, marriage, etc.); • Finally, the youngest group, under 26, also has a low score since people at the very young age seem do not know how to control their spending.
	26<= AGE< 28	77	
	28<= AGE< 32	51	
	32<= AGE< 42, _MISSING_	75	
	42<= AGE	65	
AMOUNT (Credit amount)	AMOUNT< 1352	58	• In this variable group, people who apply for medium credit amount (1352-1829 and 1829-3915) have the highest score. The reason could be an average credit amount is easier for everyone to pay back duly; • People who apply for very highest credit, i.e. 8978 receive a significant low score, means highest credit risk. The bank should be very careful with those applicants.
	1352<= AMOUNT< 1829	80	
	1829<= AMOUNT< 3915, _MISSING_	74	
	3915<= AMOUNT< 8978	57	
	8978<= AMOUNT	28	
CHECKING (Status of existing checking account)	A11	41	• In this variable group, the more money people have in their check account (A11, A12, A13), the higher credit score they get; • However, people who have no checking account (A14) or even missing checking account information get a significant high score. To confirm this case, we should the variable definition, the bank regulations, business strategy and local market characteristics.
	A12	52	
	A13	78	
	A14, _MISSING_	100	
DURATION (Duration in month)	DURATION< 9	96	• For this group, it is crystal clear that people who apply for longer credit periods have the lower score in term of credit safety.
	9<= DURATION< 12	72	
	12<= DURATION< 28, _MISSING_	64	
	28<= DURATION< 48	57	
	48<= DURATION	40	
HISTORY (Credit history)	A30, A31	25	• This variable group is one of the groups has an unclear explanation; • People who paid all credit duly (on time) get a significant low score (A30, A31); people who have a critical account get the highest score (A23). We should check the variable definition, the bank regulations and verify the data to confirm this variable effect.
	A33	60	
	A32, _MISSING_, _UNKNOWN_	63	
	A34	83	
PROPERTY (Property)	A124	54	• For this variable group, it is reasonable that people who own real estate property (A121) have the highest score and lowest credit risk; • On the other hand, people who own nothing (A124) are the very risky credit applicants.
	A122	60	
	A123, _MISSING_, _UNKNOWN_	66	
	A121	73	

PURPOSE (Purpose)	A41	88	• It is understandable that people who apply for credit to buy a 2nd car or a radio or a television are lowest risky applicants since the credit amount usually not much; • However, people who borrow money for education, retraining, buying new car, repairs something or unclear purpose get the very low score. To confirm this case, we should check the local market characteristics to clarify the reason.
	A43	86	
	A42	62	
	A44, A49	57	
	A40, A410, A45, A46, A48, _MISSING_, _UNKNOWN_	47	
SAVINGS (Savings account / bonds)	A61, _MISSING_	56	• In this variable group, people who have much money in their savings account (A63, A64) get the high score and vice versa; • However, people who do not have savings account (A65) also get a medium-high score. To clarify this case, we should check the local market characteristics and customer behavior researches.
	A62	54	
	A63	93	
	A64	99	
	A65	88	

Example of scorecard using

Score rank from 0 to 600. Cutoff score is 525.

Borrower profile:

- AGE (Age in years): 26 → Score: 77
- AMOUNT (Credit amount): 6,000 → Score: 57
- CHECKING (Status of existing checking account): 200 DM → Score: 78
- DURATION (Duration in month): 12 → Score: 64
- HISTORY (Credit history): no credit taken → Score: 25
- PROPERTY (Property): no property → Score: 54
- PURPOSE (Purpose): education → Score: 47
- SAVINGS (Savings account / bonds): 500 DM → Score: 93

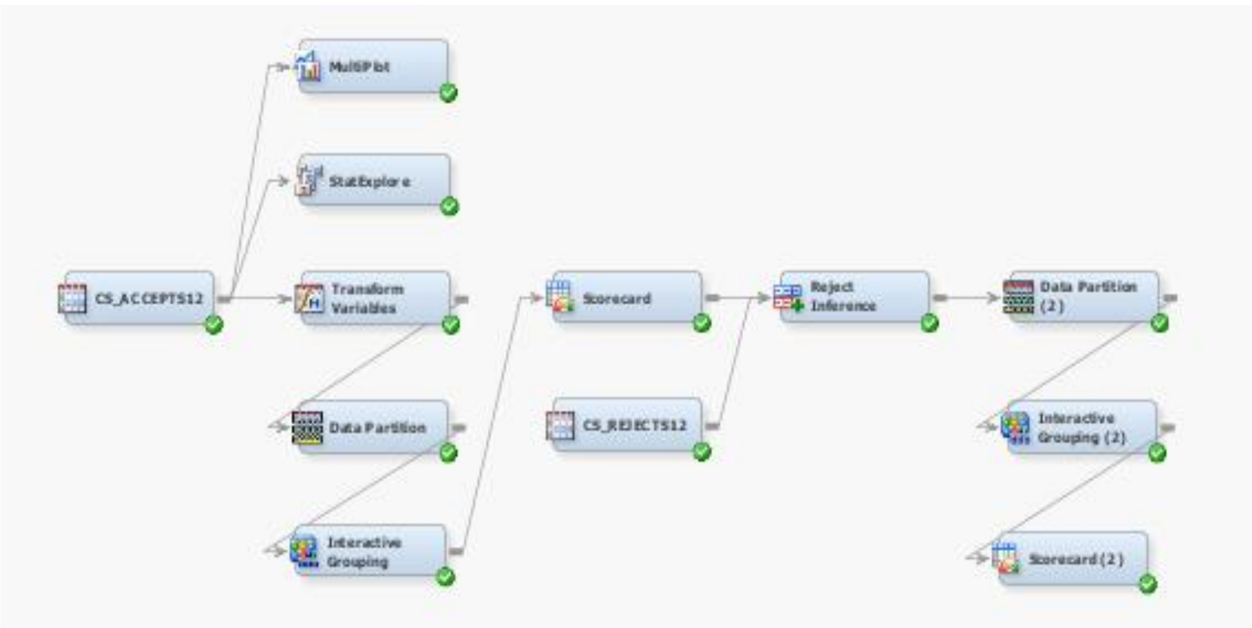
Total score: 495 < 525 (cutoff score) → Credit reject!

Part II: Scorecard Building Procedure

Overview

To build the scorecard mentioned in Part I, we can follow these steps:

- Step 1: Import and convert raw data (using SAS Base, the code is enclosed)
- Step 2: Create a SAS Enterprise Miner project, add SAS data files and diagram
- Step 3: Data exploration, transformation, partitioning and grouping
- Step 4: Build the first scorecard
- Step 5: Reject inference
- Step 6: Build final scorecard model



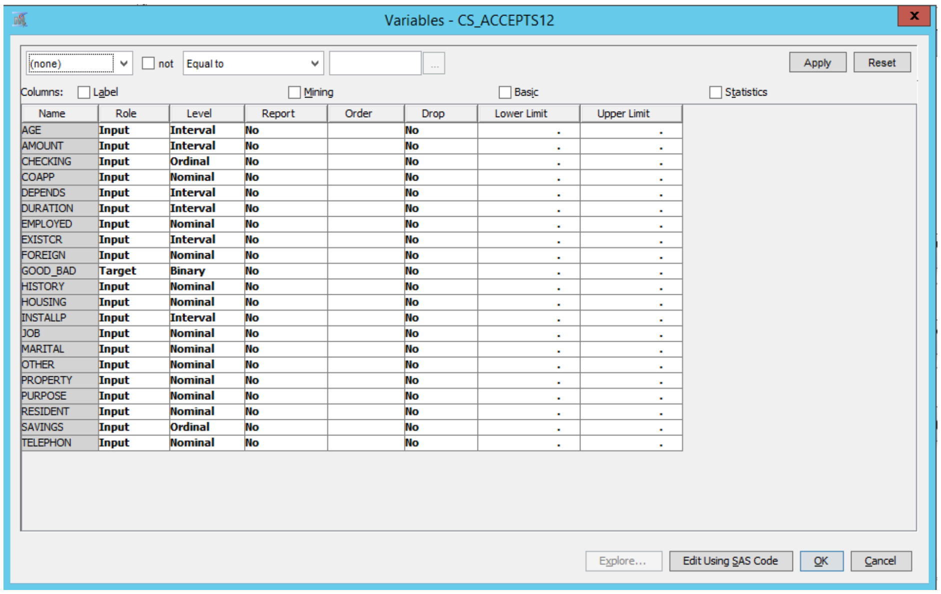
Step 1: Import and convert raw data (using SAS Base, the code is enclosed)

- Using **PROC IMPORT** to import the **accepts12.csv** and **rejects12.csv** in to SAS Base
- Set the columns **names** for the datasets
- Set the columns **labels** for the datasets
- Extract the **cs_accepts12.sas7bdat** and **cs_rejects12.sas7bdat** datasets to folder **Sasuser**

Step 2: Create a SAS Enterprise Miner project, add SAS data files and diagram

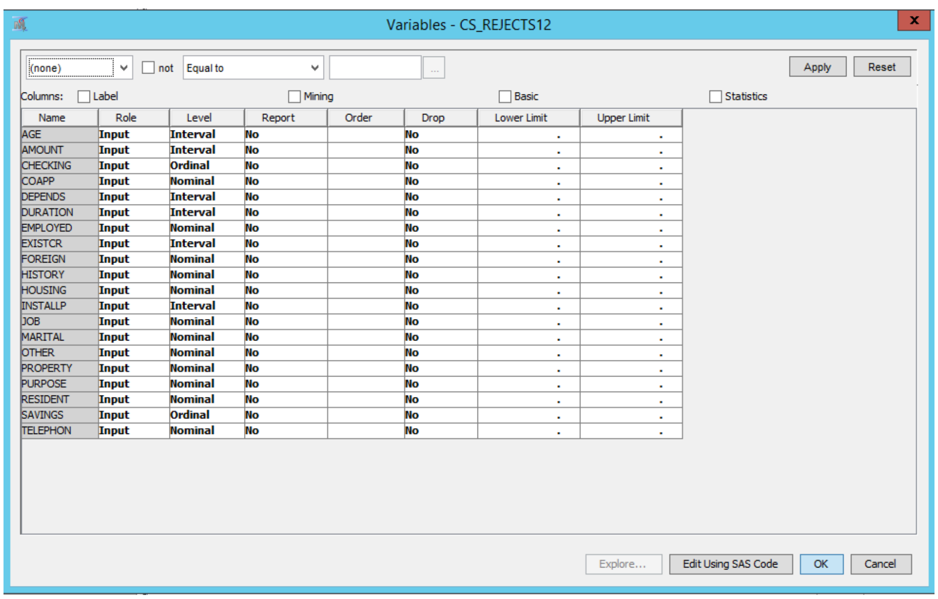
- Open SAS Miner and create a new project, enter project name as **Scorecard**
- Add the 2 datasets **cs_accepts12.sas7bdat** and **cs_rejects12.sas7bdat** from the **Sasuser** folder, name them **cs_accepts12** and **cs_rejects12**, make sure that **variables' roles and levels** are the same as follow:

For **cs_accepts12.sas7bdat**:



Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
AGE	Input	Interval	No		No	.	.
AMOUNT	Input	Interval	No		No	.	.
CHECKING	Input	Ordinal	No		No	.	.
COAPP	Input	Nominal	No		No	.	.
DEPENDS	Input	Interval	No		No	.	.
DURATION	Input	Interval	No		No	.	.
EMPLOYED	Input	Nominal	No		No	.	.
EXISTCR	Input	Interval	No		No	.	.
FOREIGN	Input	Nominal	No		No	.	.
GOOD_BAD	Target	Binary	No		No	.	.
HISTORY	Input	Nominal	No		No	.	.
HOUSING	Input	Nominal	No		No	.	.
INSTALLP	Input	Interval	No		No	.	.
JOB	Input	Nominal	No		No	.	.
MARITAL	Input	Nominal	No		No	.	.
OTHER	Input	Nominal	No		No	.	.
PROPERTY	Input	Nominal	No		No	.	.
PURPOSE	Input	Nominal	No		No	.	.
RESIDENT	Input	Nominal	No		No	.	.
SAVINGS	Input	Ordinal	No		No	.	.
TELEPHON	Input	Nominal	No		No	.	.

For **cs_rejects12.sas7bdat**:

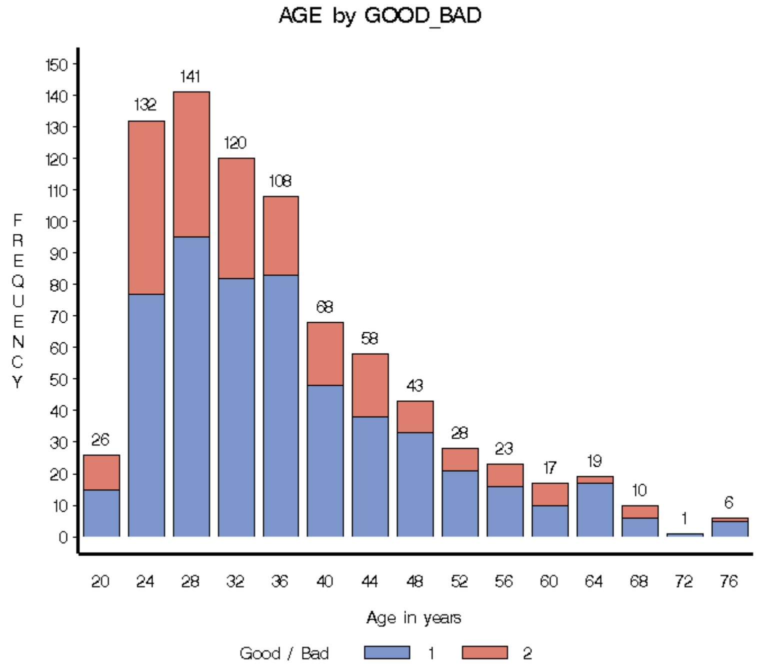


Name	Role	Level	Report	Order	Drop	Lower Limit	Upper Limit
AGE	Input	Interval	No		No	.	.
AMOUNT	Input	Interval	No		No	.	.
CHECKING	Input	Ordinal	No		No	.	.
COAPP	Input	Nominal	No		No	.	.
DEPENDS	Input	Interval	No		No	.	.
DURATION	Input	Interval	No		No	.	.
EMPLOYED	Input	Nominal	No		No	.	.
EXISTCR	Input	Interval	No		No	.	.
FOREIGN	Input	Nominal	No		No	.	.
HISTORY	Input	Nominal	No		No	.	.
HOUSING	Input	Nominal	No		No	.	.
INSTALLP	Input	Interval	No		No	.	.
JOB	Input	Nominal	No		No	.	.
MARITAL	Input	Nominal	No		No	.	.
OTHER	Input	Nominal	No		No	.	.
PROPERTY	Input	Nominal	No		No	.	.
PURPOSE	Input	Nominal	No		No	.	.
RESIDENT	Input	Nominal	No		No	.	.
SAVINGS	Input	Ordinal	No		No	.	.
TELEPHON	Input	Nominal	No		No	.	.

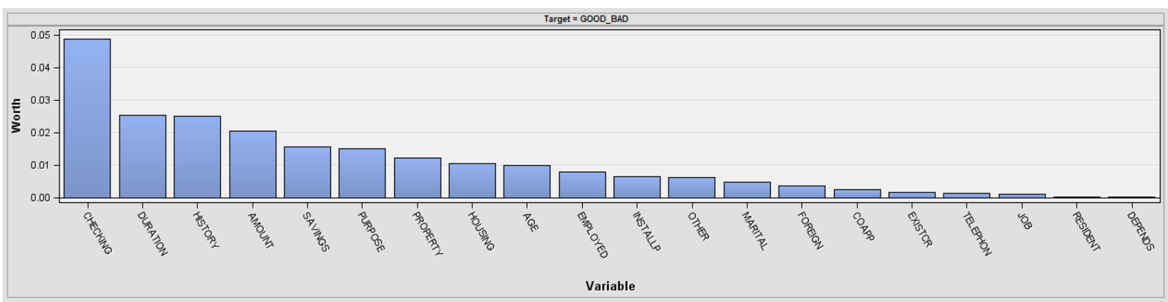
- Select **cs_accepts12** dataset, in **Property** panel, make sure its **Role** is set to **Raw**.
- Select **cs_rejects12** dataset, in **Property** panel, make sure its **Role** is set to **Score**.
- Create a new **diagram**.

Step 3: Data exploration, transformation, partitioning and grouping

- Drag and drop the **cs_accepts12** dataset to the **diagram**, from tab **Explore** add the node **MultiPlot,** connect it with the **cs_accepts12** node. Run it and inspect the results.



- From tab **Explore** add the node **StatExplore**, connect it with the **cs_accepts12** node. Run and inspect the results.

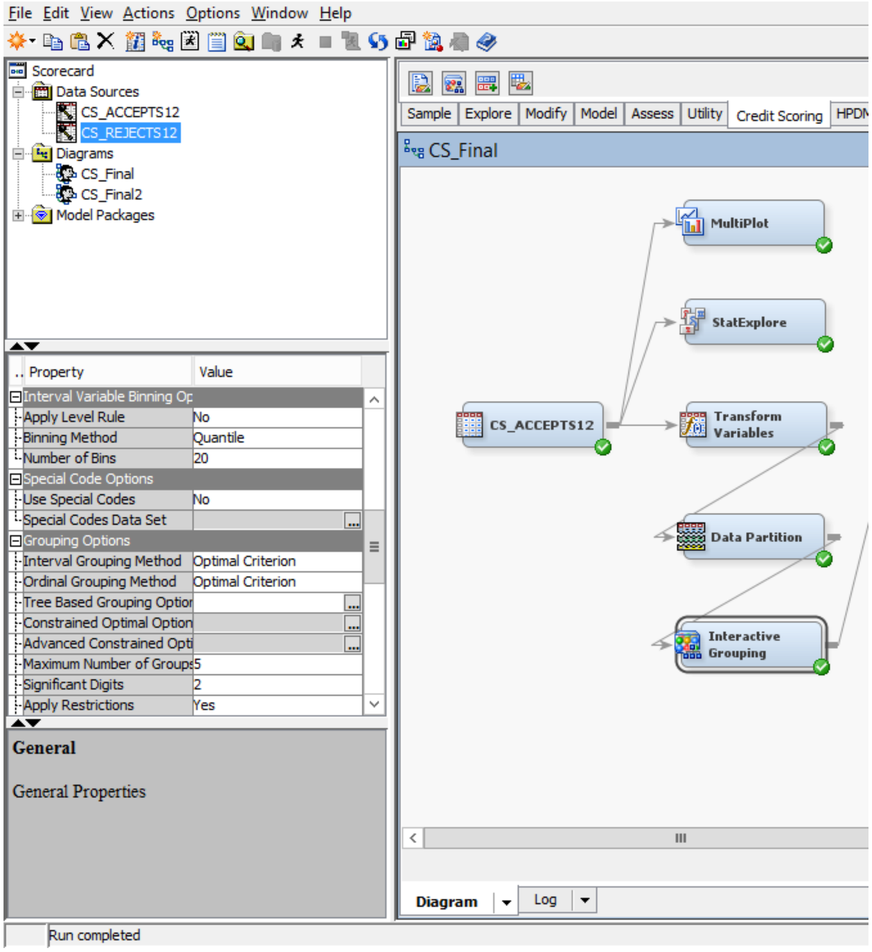


- From tab **Modify** add the node **Transform Variables**, connect it with the **cs_accepts12** node. On the **Property** panel, click the […] box on the right of **SAS Code** property, enter the following code, click **OK**:

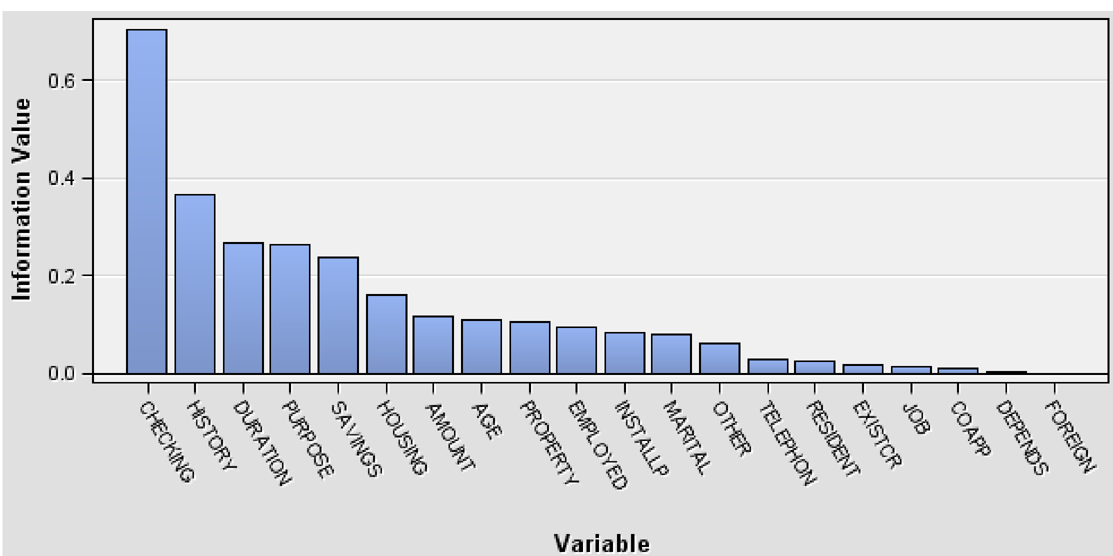
```
if GOOD_BAD=2 then GOOD_BAD=1;
else GOOD_BAD=0;
```

- From tab **Sample** add the node **Data Partition**, connect it with the **Transform Variable**. In **Property** panel select **Partitioning Method** to **Stratified**, select proportion of **Training** dataset to **70** (70%), **Validation** to **30** (30%) and **Test** to **0** (0%).

- From tab **Credit Scoring** add the node **Interactive Grouping**, connect it with the **Data Partition**, use the default option, keep **Interval Grouping Method** and **Ordinal Grouping Method** as **Optimal Criterion** to find the best groupings based on the **Tree Based Criterion**.



- Run and inspect the variables grouping result. In the char below, we can see the **importance** of each variable for the scorecard model.



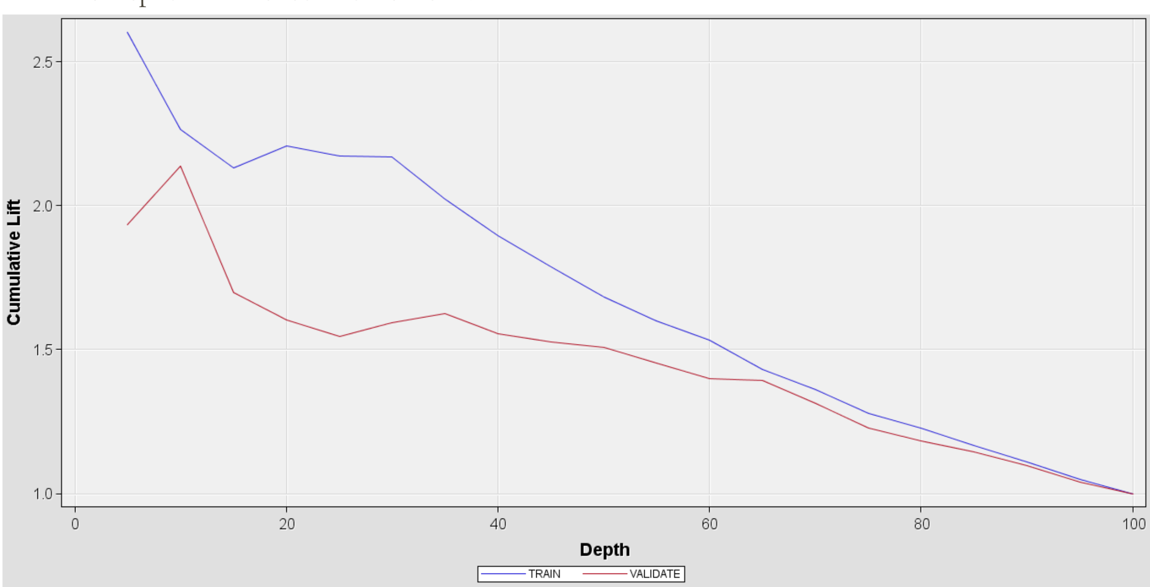
In the result, we can see that the **Tree Based** method has selected **9 variables**, based on their **Information Value (IV)** and drop the other variables by using the **cutoff** of **0.1** (default value). The 9 variables above will show in the final scorecard model.

Output Variables

Variable	Gini Statistic	Information Value	Level for Interactive	Calculated Role ▲	New Role	Pre-Defined Grouping	Level	Label	Information Value Ordering
CHECKING	42.59	0.705	ORDINAL	Input	Default		ORDINAL	Status of ex...	1
HISTORY	28.543	0.367	NOMINAL	Input	Default		NOMINAL	Credit history	2
DURATION	24.996	0.266	INTERVAL	Input	Default		INTERVAL	Duration in ...	3
PURPOSE	27.823	0.261	NOMINAL	Input	Default		NOMINAL	Purpose	4
SAVINGS	21.393	0.238	ORDINAL	Input	Default		ORDINAL	Savings ac...	5
HOUSING	18.637	0.161	NOMINAL	Input	Default		NOMINAL	Housing	6
AMOUNT	17.753	0.115	INTERVAL	Input	Default		INTERVAL	Credit amo...	7
AGE	17.475	0.109	INTERVAL	Input	Default		INTERVAL	Age in years	8
PROPERTY	14.724	0.105	NOMINAL	Input	Default		NOMINAL	Property	9
EMPLOYED	16.419	0.094	NOMINAL	Rejected	Default		NOMINAL	Present em...	10
INSTALLP	14.863	0.083	INTERVAL	Rejected	Default		INTERVAL	Installment ...	11
MARITAL	14.051	0.078	NOMINAL	Rejected	Default		NOMINAL	Personal st...	12
OTHER	9.466	0.059	NOMINAL	Rejected	Default		NOMINAL	Other instal...	13
TELEPHON	8.1	0.028	NOMINAL	Rejected	Default		NOMINAL	Telephone	14
RESIDENT	7.91	0.023	NOMINAL	Rejected	Default		NOMINAL	Present res...	15
EXISTCR	6.095	0.016	INTERVAL	Rejected	Default		INTERVAL	Number of ...	16
JOB	4.795	0.013	NOMINAL	Rejected	Default		NOMINAL	Job	17
COAPP	2.088	0.01	NOMINAL	Rejected	Default		NOMINAL	Other debto...	18
DEPENDS	1.556	0.002	INTERVAL	Rejected	Default		INTERVAL	Number of ...	19
FOREIGN	0	0	NOMINAL	Rejected	Default		NOMINAL	Foreign wor...	20

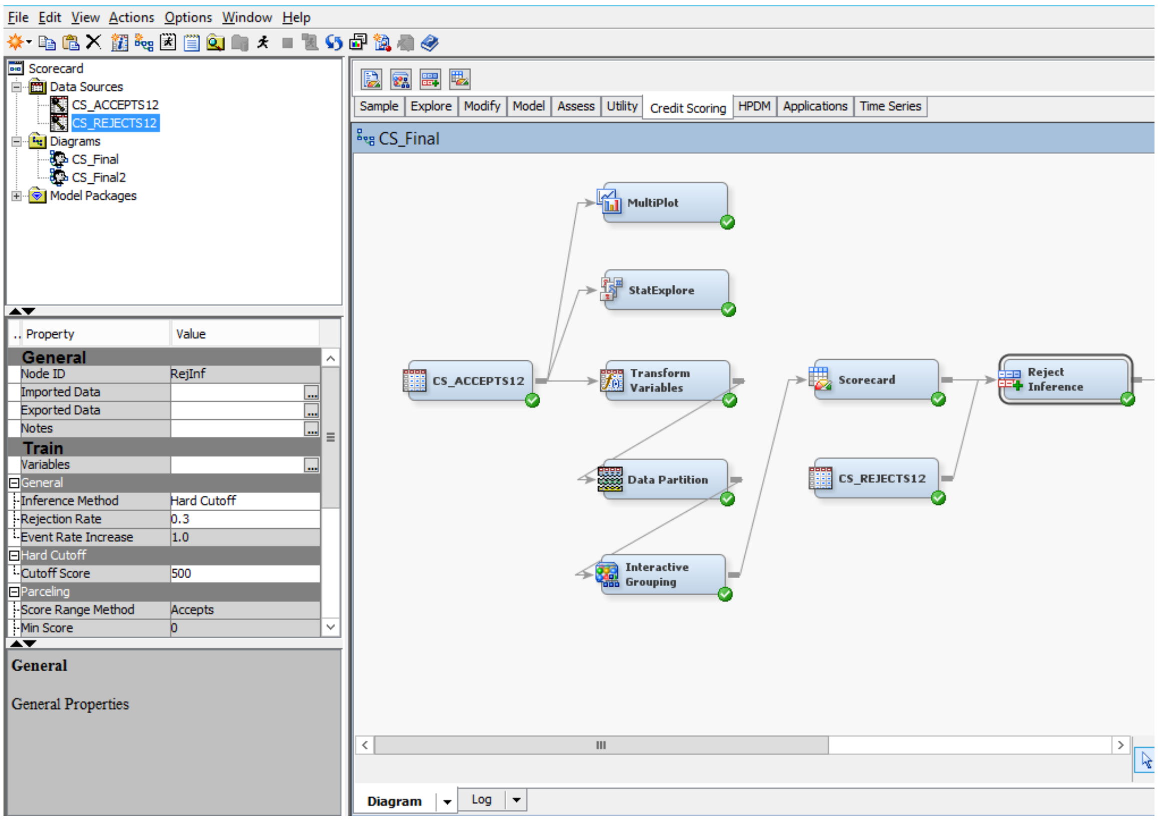
Step 4: Build the first scorecard

- From tab **Credit Scoring** add **Scorecard** node, connect it with the **Interactive Grouping**. In the **Property** panel, change **Scorecard Points** to **600** to scale the rank of the score from 0 to 600; change **Scorecard Type** to **Detailed**; set the **Number of Buckets** to **20**; set the **Revenue Accepted Good** to **1000** and **Cost Accepted Bad** to **5000**; keep the default **Current Approval Rate** of **70** and **Current Event Rate** of **2.5**; set **Generate Characteristic Analysis** to **Yes**.
- In the **Adverse Characteristic Options** on **Property** panel, change **Method** to **Weighted Average Score**.
- Run and inspect the first scorecard result.

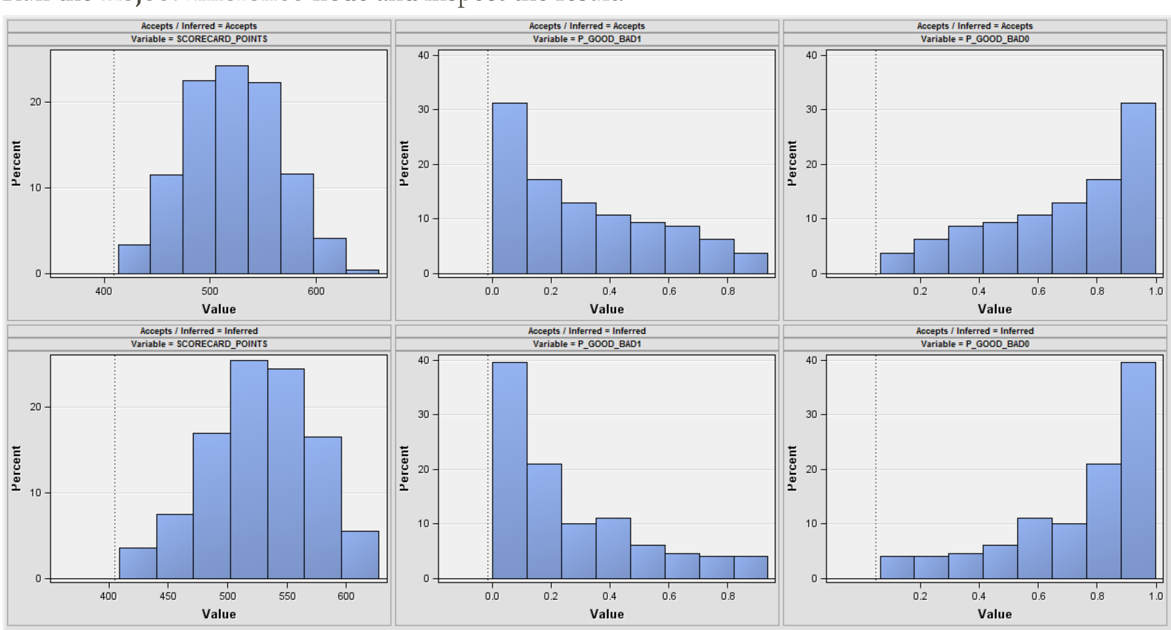


Step 5: Reject inference

- From **Credit Scoring** tab add the node **Reject Inference**, connect it with the **Scorecard**. In the **Property** panel, select **Inference Method** to **Hard Cutoff**, set the value of **Cutoff Score** to **500**.
- Add the **cs_rejects12** dataset to the **diagram**, connect it with the **Reject Inference**. The diagram should look as follow:

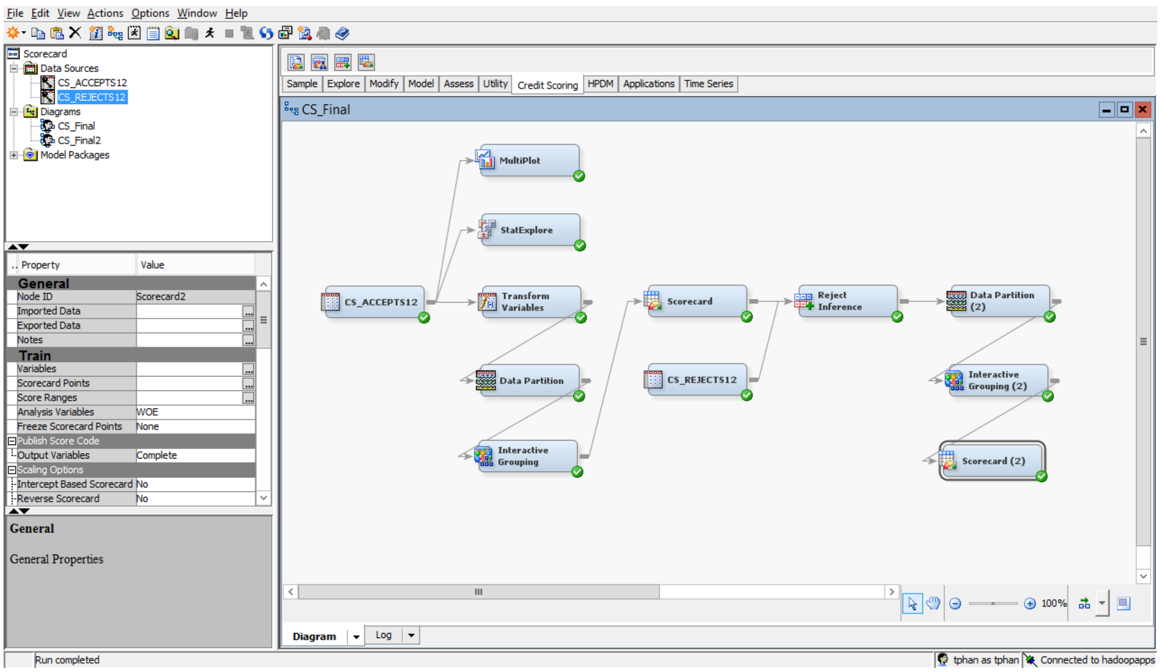


- Run the **Reject Inference** node and inspect the result.



Step 6: Build final scorecard model

- From tab **Sample** add the second node of **Data Partition**, connect it with the **Reject Inference** node. Set the **Property** as same as the first **Data Partition** node in **Step 3**.
- From tab **Credit Scoring** add the second node of **Interactive Grouping**, connect it with the above **Data Partition** node. Set the **Property** as same as the first **Interactive Grouping** node in **Step 3**.
- From tab **Credit Scoring** add the second node of **Scorecard**, connect it with the above **Interactive Grouping** node. Set the **Property** as same as the first **Scorecard** node in **Step 4**.
- The final diagram should look like follow:



- Run the second **Scorecard** node and inspect the result.

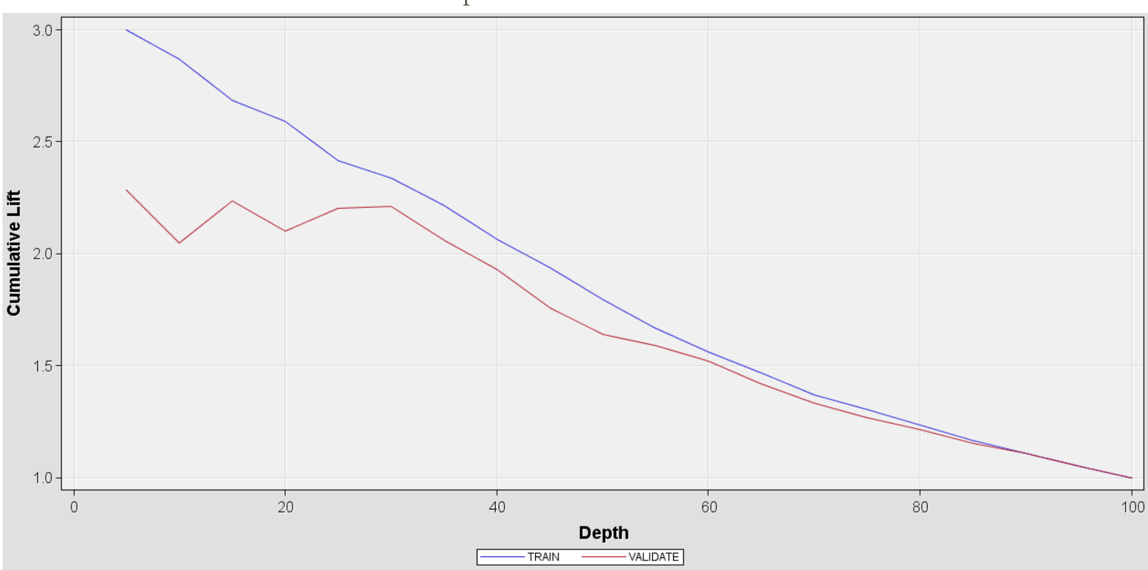
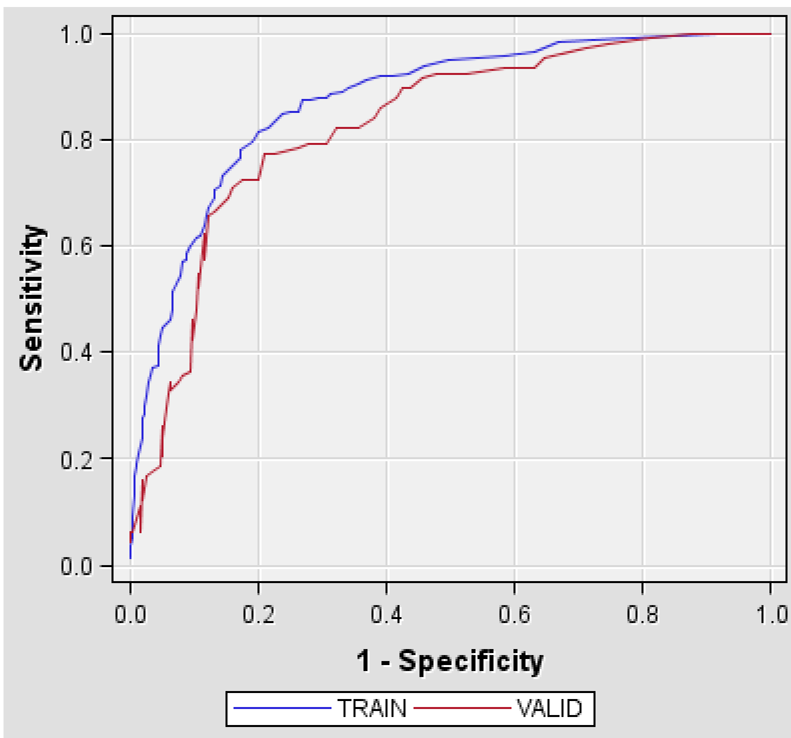


Figure 3: Score Ranking Overlay: Good / Bad

Fit statistical results in very encouraged, **AUR = .87** (for Training) and **AUR = 0.83** (for Validation).

Target	Target Label	Fit Statistics	Statistics Label	Train	Validation
GOOD_BAD	Good / Bad	_AIC_	Akaike's Information Criterion	664.7603	.
GOOD_BAD	Good / Bad	_ASE_	Average Squared Error	0.127873	0.155262
GOOD_BAD	Good / Bad	_AVERR_	Average Error Function	0.401358	0.474668
GOOD_BAD	Good / Bad	_DFE_	Degrees of Freedom for Error	796.7143	.
GOOD_BAD	Good / Bad	_DFM_	Model Degrees of Freedom	9	.
GOOD_BAD	Good / Bad	_DFT_	Total Degrees of Freedom	805.7143	.
GOOD_BAD	Good / Bad	_DIV_	Divisor for ASE	1611.429	674.2857
GOOD_BAD	Good / Bad	_ERR_	Error Function	646.7603	320.0621
GOOD_BAD	Good / Bad	_FPE_	Final Prediction Error	0.130762	.
GOOD_BAD	Good / Bad	_MAX_	Maximum Absolute Error	0.987187	0.970021
GOOD_BAD	Good / Bad	_MSE_	Mean Square Error	0.129318	0.155262
GOOD_BAD	Good / Bad	_NOBS_	Sum of Frequencies	805.7143	337.1429
GOOD_BAD	Good / Bad	_NW_	Number of Estimate Weights	9	.
GOOD_BAD	Good / Bad	_RASE_	Root Average Sum of Squares	0.357594	0.394032
GOOD_BAD	Good / Bad	_RFPE_	Root Final Prediction Error	0.361611	.
GOOD_BAD	Good / Bad	_RMSE_	Root Mean Squared Error	0.359608	0.394032
GOOD_BAD	Good / Bad	_SBC_	Schwarz's Bayesian Criterion	706.9859	.
GOOD_BAD	Good / Bad	_SSE_	Sum of Squared Errors	206.0586	104.6907
GOOD_BAD	Good / Bad	_SUMW_	Sum of Case Weights Times F...	1611.429	674.2857
GOOD_BAD	Good / Bad	_MISC_	Misclassification Rate	0.186525	0.219492
GOOD_BAD	Good / Bad	_KS_	Kolmogorov-Smirnov Statistic	0.616438	0.564416
GOOD_BAD	Good / Bad	_AUR_	Area Under ROC	0.870928	0.825799
GOOD_BAD	Good / Bad	_Gini_	Gini Coefficient	0.741857	0.651597
GOOD_BAD	Good / Bad	_ARATIO_	Accuracy Ratio	0.741857	0.651597

The **ROC** Plot:



- Based on **Gain Table**, we can see at the score **525**, from both Training and Validation set, the bank begin to have **positive profit**. Under **525** score, the bank is losing money (negative profit). Therefore, we decided to choose the **cutoff score** at **525**. At the cutoff score of **525**, the model will cover **63.4%** of the **population**, this is a very good result to the bank.

  In addition, the value of **Revenue Accepted Good** = 1000 and **Cost Accepted Bad** = 5000 we set in **Step 4** are also impact to the calculation and selection of the cutoff value, since they affect directly to the profit calculation.

al ent	Cumulative Event Rate	Cumulative Non-Event Rate	Average Predicted Probability	Low Predicted Probability Threshold	High Predicted Probability Threshold	Cumulative Approval Rate	Data Role	Average Marginal Profit	Average Total Profit	Cutoff Score	Population Percentage	Type	Frequency	Average Scorecard Points	Empirical Odds	Predicted Odds
0	0	0	.	.	.	0	TRAIN	1000	0	.	0	.	.	.	0	.
0	0	0	.	.	.	0	TRAIN	1000	0	669	0	.	.	.	0	.
100	0	100	0.004874	0.003877	0.005348	1.046099	TRAIN	1000	10.46099	653	1.046099	0	7	642.7143	-2.8824	-5.31893
100	0	100	0.008646	0.005825	0.010112	3.847518	TRAIN	1000	38.47518	637	3.847518	0	19	625.3684	-3.83174	-4.74196
66667	1.353965	98.64603	0.013905	0.010019	0.017068	9.166667	TRAIN	918.7621	84.21986	621	9.166667	0	35	611.7429	-3.73426	-4.26149
7.7707	1.684717	98.31528	0.024552	0.017675	0.029652	14.73404	TRAIN	898.917	132.4468	605	14.73404	0	37	594.9459	-3.78094	-3.68211
10145	2.130898	97.8691	0.039907	0.029734	0.050445	23.29787	TRAIN	872.1461	203.1915	589	23.29787	0	54	580.0926	-3.51155	-3.18049
83333	4.022989	95.97701	0.069139	0.049301	0.085686	33.93617	TRAIN	758.6207	257.4468	573	33.93617	0	75	563.2267	-2.41991	-2.59999
16719	5.21978	94.78022	0.109017	0.086453	0.138988	45.1773	TRAIN	686.8132	310.2837	557	45.1773	0	77	548.8442	-2.33422	-2.10082
41296	6.903353	93.09665	0.177044	0.133079	0.2175	53.93617	TRAIN	585.7988	315.9574	541	53.93617	0	62	532.3871	-1.68928	-1.53651
35714	9.661299	90.3387	0.276761	0.215833	0.325992	63.86525	TRAIN	420.322	268.4397	525	63.86525	0	70	515.5286	-1.11775	-0.96058
07064	13.48952	86.51048	0.388205	0.328257	0.463964	71.89716	TRAIN	190.6289	137.0567	509	71.89716	0	59	500.8136	-0.24403	-0.45486
45936	19.13006	80.86994	0.530888	0.456258	0.596109	81.93262	TRAIN	-147.804	-121.099	493	81.93262	0	73	483.7123	0.386361	0.123709
59591	22.50599	77.49401	0.64907	0.591308	0.706312	88.86525	TRAIN	-350.359	-311.348	477	88.86525	0	53	469.4717	0.506736	0.614955
01961	24.59739	75.40261	0.769622	0.71991	0.817064	92.48227	TRAIN	-475.844	-440.071	461	92.48227	0	27	452.2222	1.151605	1.206181
68531	26.21758	73.78242	0.855343	0.819462	0.88326	95.01773	TRAIN	-573.055	-544.504	445	95.01773	0	19	435.8947	1.759499	1.777137
96581	28.80386	71.19614	0.907595	0.887046	0.930546	99.16667	TRAIN	-728.232	-722.163	429	99.16667	0	27	420.963	1.995672	2.284622
33333	28.94549	71.05451	0.94039	0.929104	0.954071	99.53901	TRAIN	-736.73	-733.333	413	99.53901	0	3	407	0.693147	2.758476
0	29.03398	70.96602	0.96007	0.96007	0.96007	99.66312	TRAIN	-742.039	-739.539	397	99.66312	0	1	396	1.098612	3.179877
0	29.27305	70.72695	0.982237	0.978535	0.985939	100	TRAIN	-756.383	-756.383	381	100	0	2	371	1.860752	4.012711
100	0	100	0.001573	0.001573	0.001573	0.29661	VALID	1000	2.966102	.	0.29661	0	1	675	-1.09861	-6.4535
0	0	100	.	.	.	0.29661	VALID	1000	2.966102	669	0.29661	.	.	.	0	.
100	0	100	0.003759	0.003707	0.00381	1.101695	VALID	1000	11.01695	653	1.101695	0	2	649.5	-1.86075	-5.57995
100	0	100	0.008204	0.00694	0.009355	2.288136	VALID	1000	22.88136	637	2.288136	0	4	627.5	-2.19722	-4.79484
100	0	100	0.01492	0.010546	0.018245	4.661017	VALID	1000	46.61017	621	4.661017	0	8	610.25	-2.83321	-4.19001
85799	2.508961	97.49104	0.023499	0.017815	0.030175	11.82203	VALID	849.4624	100.4237	605	11.82203	0	22	596.4091	-3.14169	-3.72703
27027	3.278689	96.72131	0.036795	0.029083	0.050462	18.09322	VALID	803.2787	145.339	589	18.09322	0	19	582.4737	-3.00285	-3.26491
09091	7.573416	92.42658	0.06622	0.0526	0.083631	27.41525	VALID	545.5951	149.5763	573	27.41525	0	30	564.7	-1.66501	-2.64625
3.3121	7.284079	92.71592	0.108284	0.082998	0.137521	40.72034	VALID	562.9553	229.2373	557	40.72034	0	37	549.027	-2.63565	-2.10839
74888	11.23311	88.76689	0.180687	0.137874	0.216693	50.16949	VALID	326.0135	163.5593	541	50.16949	0	29	531.4483	-0.93204	-1.5117
67732	13.76086	86.23914	0.277267	0.229867	0.320772	63.4322	VALID	174.3487	110.5932	525	63.4322	0	39	515.4872	-1.19018	-0.95806
17699	17.17934	82.82066	0.390939	0.327005	0.450679	73.00847	VALID	-30.7603	-22.4576	509	73.00847	0	28	500.4643	-0.41285	-0.44337
81992	24.54637	75.45363	0.53285	0.467565	0.59793	84.0678	VALID	-472.782	-397.458	493	84.0678	0	33	483.697	1.003778	0.131588
97345	26.94325	73.05675	0.659702	0.598893	0.701001	88.85593	VALID	-616.595	-547.881	477	88.85593	0	14	468.0714	0.801361	0.661968
51613	28.81585	71.18415	0.755278	0.718342	0.814198	94.11017	VALID	-728.951	-686.017	461	94.11017	0	17	454.5882	0.425668	1.126964

Figure 4: Gain Table – Identify Cutoff Score

Appendix 1: Data Description

Attribute description for German

- **Attribute 1**: (qualitative) Status of existing checking account
  - A11 : … < 0 DM
  - A12 : 0 <= … < 200 DM
  - A13 : … >= 200 DM / salary assignments for at least 1 year
  - A14 : no checking account
- **Attribute 2**: (numerical) Duration in month

- **Attribute 3**: (qualitative) Credit history
  - A30 : no credits taken/all credits paid back duly
  - A31 : all credits at this bank paid back duly
  - A32 : existing credits paid back duly till now
  - A33 : delay in paying off in the past
  - A34 : critical account/other credits existing (not at this bank)
- **Attribute 4**: (qualitative) Purpose
  - A40 : car (new)
  - A41 : car (used)
  - A42 : furniture/equipment
  - A43 : radio/television
  - A44 : domestic appliances
  - A45 : repairs
  - A46 : education
  - A47 : (vacation - does not exist?)
  - A48 : retraining
  - A49 : business
  - A410 : others
- **Attribute 5**: (numerical) Credit amount

- **Attribute 6**: (qualitative) Savings account/bonds
  - A61 : … < 100 DM
  - A62 : 100 <= … < 500 DM
  - A63 : 500 <= … < 1000 DM
  - A64 : .. >= 1000 DM
  - A65 : unknown/ no savings account
- **Attribute 7**: (qualitative) Present employment since
  - A71 : unemployed
  - A72 : … < 1 year
  - A73 : 1 <= … < 4 years
  - A74 : 4 <= … < 7 years
  - A75 : .. >= 7 years
- **Attribute 8**: (numerical) Installment rate in percentage of disposable income

- **Attribute 9**: (qualitative) Personal status and sex

- A91 : male : divorced/separated
- A92 : female : divorced/separated/married
- A93 : male : single
- A94 : male : married/widowed
- A95 : female : single

- **Attribute 10**: (qualitative) Other debtors / guarantors
  - A101 : none
  - A102 : co-applicant
  - A103 : guarantor
- **Attribute 11**: (numerical) Present residence since
- **Attribute 12**: (qualitative) Property
  - A121 : real estate
  - A122 : if not A121 : building society savings agreement/life insurance
  - A123 : if not A121/A122 : car or other, not in attribute 6
  - A124 : unknown / no property
- **Attribute 13**: (numerical) Age in years
- **Attribute 14**: (qualitative) Other installment plans
  - A141 : bank
  - A142 : stores
  - A143 : none
- **Attribute 15**: (qualitative) Housing
  - A151 : rent
  - A152 : own
  - A153 : for free
- **Attribute 16**: (numerical) Number of existing credits at this bank
- **Attribute 17**: (qualitative) Job
  - A171 : unemployed/ unskilled - non-resident
  - A172 : unskilled - resident
  - A173 : skilled employee / official
  - A174 : management/ self-employed/highly qualified employee/ officer
- **Attribute 18**: (numerical) Number of people being liable to provide maintenance for
- **Attribute 19**: (qualitative) Telephone
  - A191 : none
  - A192 : yes, registered under the customers name
- **Attribute 21**: (qualitative) foreign worker
  - A201 : yes
  - A202 : no

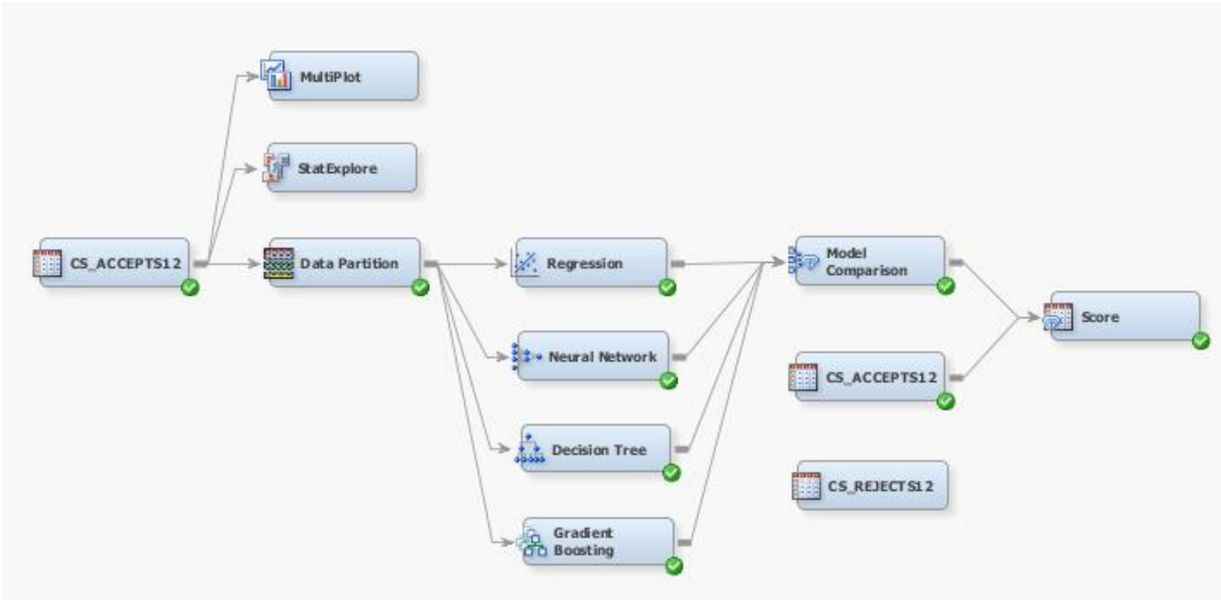
Appendix 2: SAS Code to Read and Convert Data

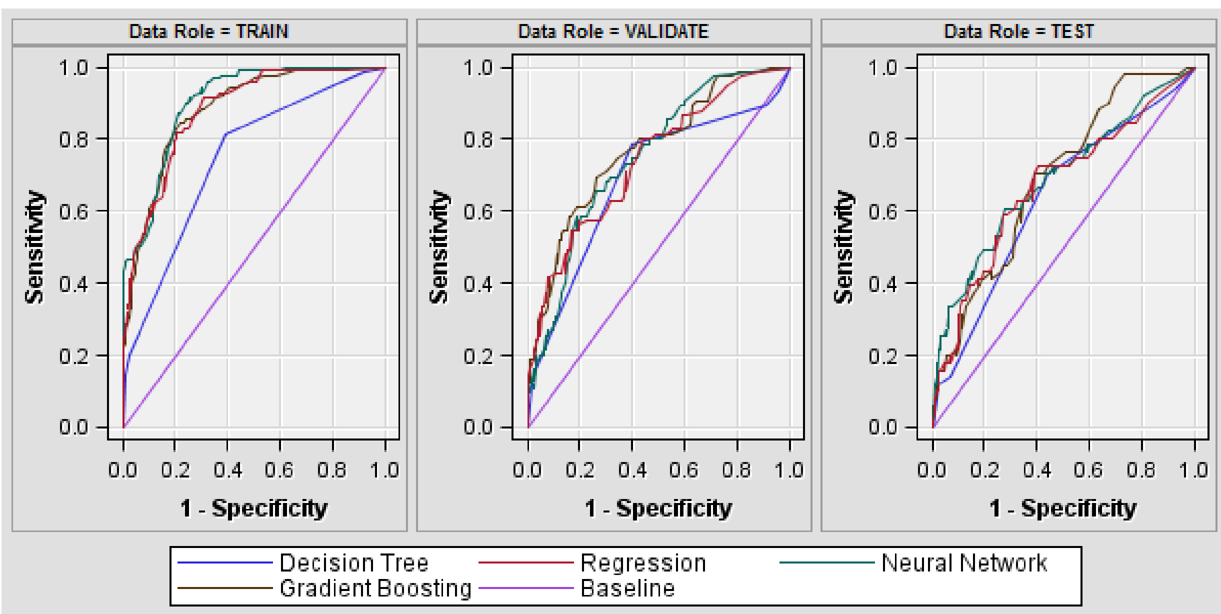
```
/*****************************************************************/
/* Import accepts12.csv file                                     */
/*****************************************************************/
proc import datafile="C:\Users\tphan\Desktop\Final project\data\accepts12.csv"
out=cs_accepts12 dbms=dlm replace;
	delimiter=',';
	getnames=no;
run;

/* Set columns names */
data cs_accepts12;
	set cs_accepts12;
	rename VAR1=CHECKING;
	rename VAR2=DURATION;
	rename VAR3=HISTORY;
	rename VAR4=PURPOSE;
	rename VAR5=AMOUNT;
	rename VAR6=SAVINGS;
	rename VAR7=EMPLOYED;
	rename VAR8=INSTALLP;
	rename VAR9=MARITAL;
	rename VAR10=COAPP;
	rename VAR11=RESIDENT;
	rename VAR12=PROPERTY;
	rename VAR13=AGE;
	rename VAR14=OTHER;
	rename VAR15=HOUSING;
	rename VAR16=EXISTCR;
	rename VAR17=JOB;
	rename VAR18=DEPENDS;
	rename VAR19=TELEPHON;
	rename VAR20=FOREIGN;
	rename VAR21=GOOD_BAD;
run;

/* Set columns labels */
data cs_accepts12;
	set cs_accepts12;
	label CHECKING="Status of existing checking account";
	label DURATION="Duration in month";
	label HISTORY="Credit history";
	label PURPOSE="Purpose";
	label AMOUNT="Credit amount";
	label SAVINGS="Savings account / bonds";
	label EMPLOYED="Present employment since";
	label INSTALLP="Installment rate in percentage of disposable income";
	label MARITAL="Personal status and sex";
	label COAPP="Other debtors / guarantors";
	label RESIDENT="Present residence since";
	label PROPERTY="Property";
	label AGE="Age in years";
	label OTHER="Other installment plans";
	label HOUSING="Housing";
	label EXISTCR="Number of existing credits at this bank";
	label JOB="Job";
	label DEPENDS="Number of people being liable to provide maintenance for";
	label TELEPHON="Telephone";
	label FOREIGN="Foreign worker";
	label GOOD_BAD="Good / Bad";
run;
```

```
/******************************************************************/
/* Import rejects12.csv file                                      */
/******************************************************************/
proc import datafile="C:\Users\tphan\Desktop\Final project\data\rejects12.csv"
out=cs_rejects12 dbms=dlm replace;
	delimiter=',';
	getnames=no;
run;

/* Set columns names */
data cs_rejects12;
	set cs_rejects12;
	rename VAR1=CHECKING;
	rename VAR2=DURATION;
	rename VAR3=HISTORY;
	rename VAR4=PURPOSE;
	rename VAR5=AMOUNT;
	rename VAR6=SAVINGS;
	rename VAR7=EMPLOYED;
	rename VAR8=INSTALLP;
	rename VAR9=MARITAL;
	rename VAR10=COAPP;
	rename VAR11=RESIDENT;
	rename VAR12=PROPERTY;
	rename VAR13=AGE;
	rename VAR14=OTHER;
	rename VAR15=HOUSING;
	rename VAR16=EXISTCR;
	rename VAR17=JOB;
	rename VAR18=DEPENDS;
	rename VAR19=TELEPHON;
	rename VAR20=FOREIGN;
run;

/* Set columns labels */
data cs_rejects12;
	set cs_rejects12;
	label CHECKING="Status of existing checking account";
	label DURATION="Duration in month";
	label HISTORY="Credit history";
	label PURPOSE="Purpose";
	label AMOUNT="Credit amount";
	label SAVINGS="Savings account / bonds";
	label EMPLOYED="Present employment since";
	label INSTALLP="Installment rate in percentage of disposable income";
	label MARITAL="Personal status and sex";
	label COAPP="Other debtors / guarantors";
	label RESIDENT="Present residence since";
	label PROPERTY="Property";
	label AGE="Age in years";
	label OTHER="Other installment plans";
	label HOUSING="Housing";
	label EXISTCR="Number of existing credits at this bank";
	label JOB="Job";
	label DEPENDS="Number of people being liable to provide maintenance for";
	label TELEPHON="Telephone";
	label FOREIGN="Foreign worker";
run;
```

Appendix 3: Compare Other Models

Comparing the power of Logistic Regression, Neural Network, Decision Tree and Gradient Boosting.





Model Node	Train: Roc Index	Valid: Roc Index	Test: Roc Index
Boost	0.879	0.768	0.685
Reg	0.879	0.739	0.664
Tree	0.743	0.692	0.631
Neural	0.903	0.751	0.686