ILLINOIS INSTITUTE OF TECHNOLOGY

CREDIT CARD FRAUD ANALYSIS

Akarsh Bansal	A20405767	abansal11@hawk.iit.edu
Jaideep Whabi	A20403110	jwhabi@hawk.iit.edu
Pragya Mishra	A20406346	pmishra7@hawk.iit.edu

ILLINOIS INSTITUTE OF TECHNOLOGY

Contents

Introduction 1
Data Gathering 1
Tools Used 1
Data Exploration 2
Methods and Processes 5
Data Preparation 6
Data Analysis 7
Data Validation 8
Results 8
Conclusions 13

ILLINOIS INSTITUTE OF TECHNOLOGY

Introduction

In recent years, the credit card issuers in Taiwan faced the cash and credit card debt crisis. To increase the market share, card-issuing banks in Taiwan over-issued cash and credit cards to unqualified applicants. At the same time, most cardholders, irrespective of their repayment ability, overused credit card for consumption and accumulated heavy credit and cash– card debts. The crisis caused a major blow to consumer finance confidence and it is a big challenge for both banks and cardholders. In a well-developed financial system, crisis management is on the downstream and risk prediction is on the upstream. The major purpose of risk prediction is to use financial information, such as business financial statement, customer transaction and repayment records, etc., to predict business performance or individual customers' credit risk and to reduce the damage and uncertainty.

Data Gathering

This dataset contains information on default payments, demographic factors, credit data, history of payment, and bill statements of credit card clients in Taiwan from April 2005 to September 2005. The data has been collected from Kaggle.com. The number of instances is 30,000 and the number of variables is 25.
https://www.kaggle.com/uciml/default-of-credit-card-clients-dataset

Tools Used

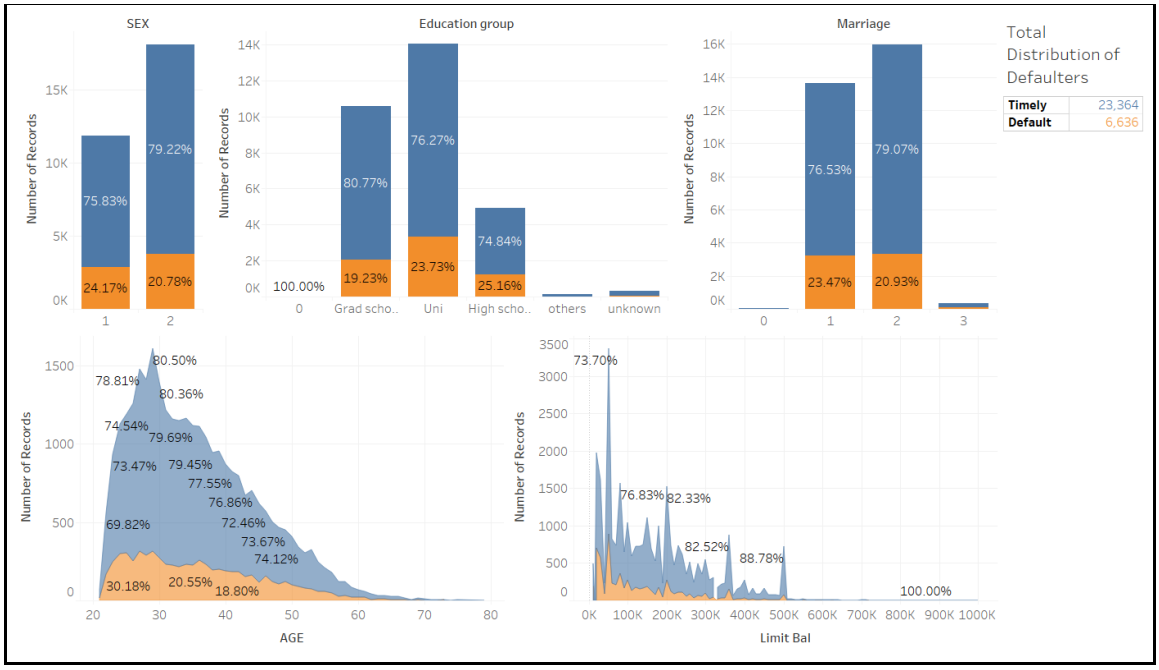
1) RStudio
   It is a free and open-source integrated development environment(IDE) for R, a programming language for statistical computing and graphics.
2) Tableau
   It is used for creating data visualizations, publishing data sources as well as workbooks to Tableau Server.

ILLINOIS INSTITUTE OF TECHNOLOGY

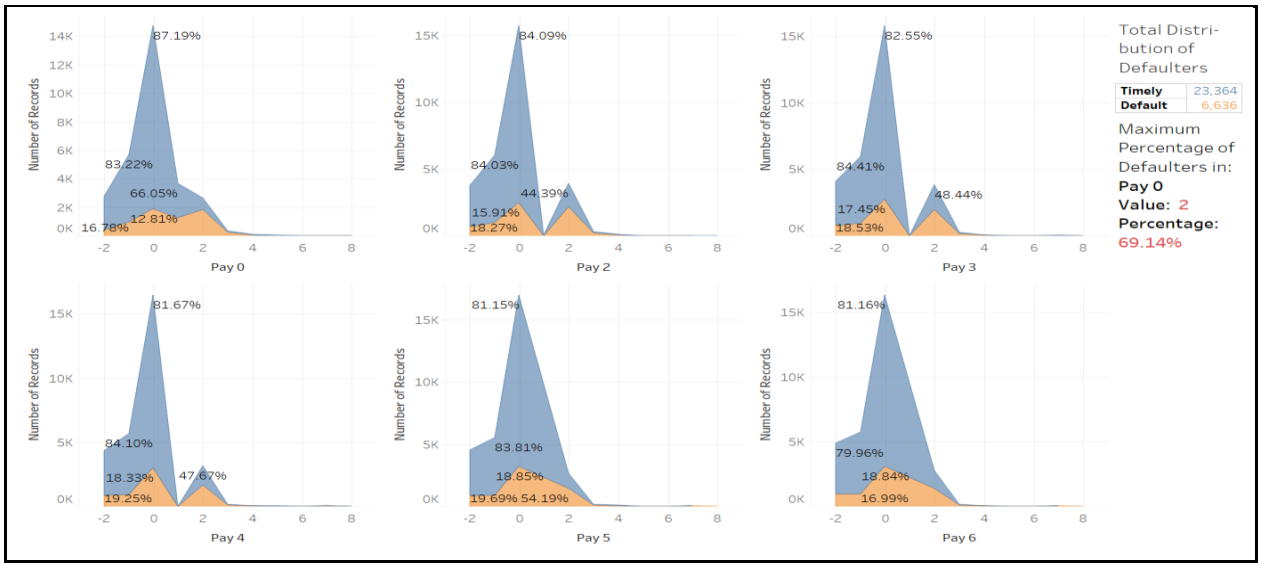
Data Exploration

We check if the data has any biases.
The data has 25 variables:

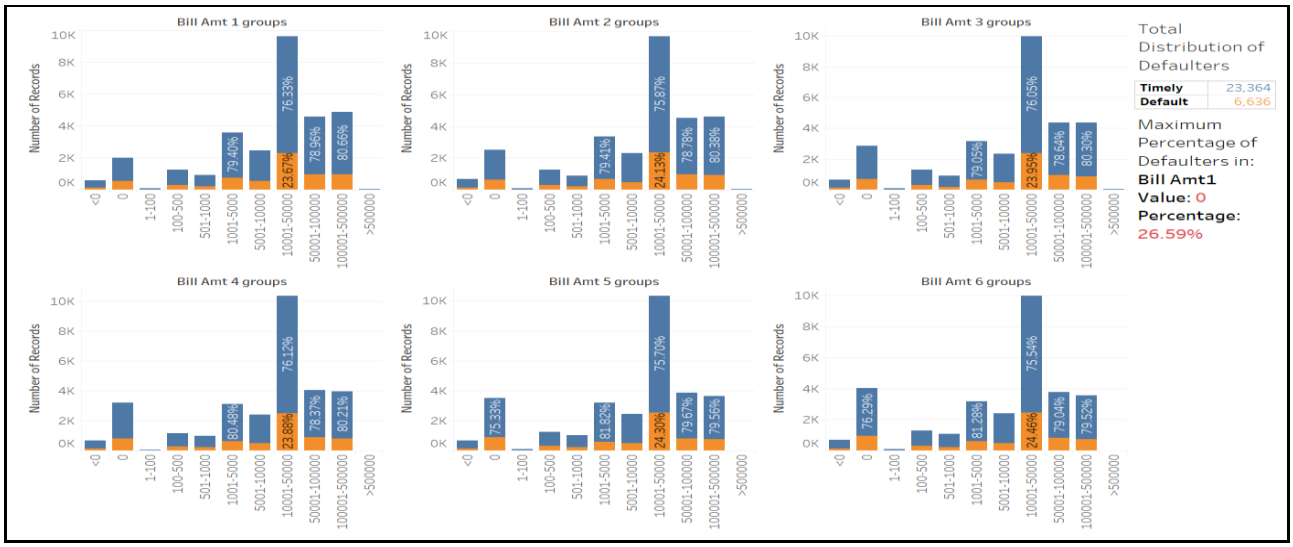
- ID: ID of each client
- LIMIT_BAL: Amount of given credit in NT dollars (includes individual and family/supplementary credit
- SEX: Gender (1=male, 2=female)
- EDUCATION:(1=graduate school, 2=university, 3=high school, 4=others, 5=unknown, 6=unknown)
- MARRIAGE: Marital status (1=married, 2=single, 3=others
- AGE: Age in years



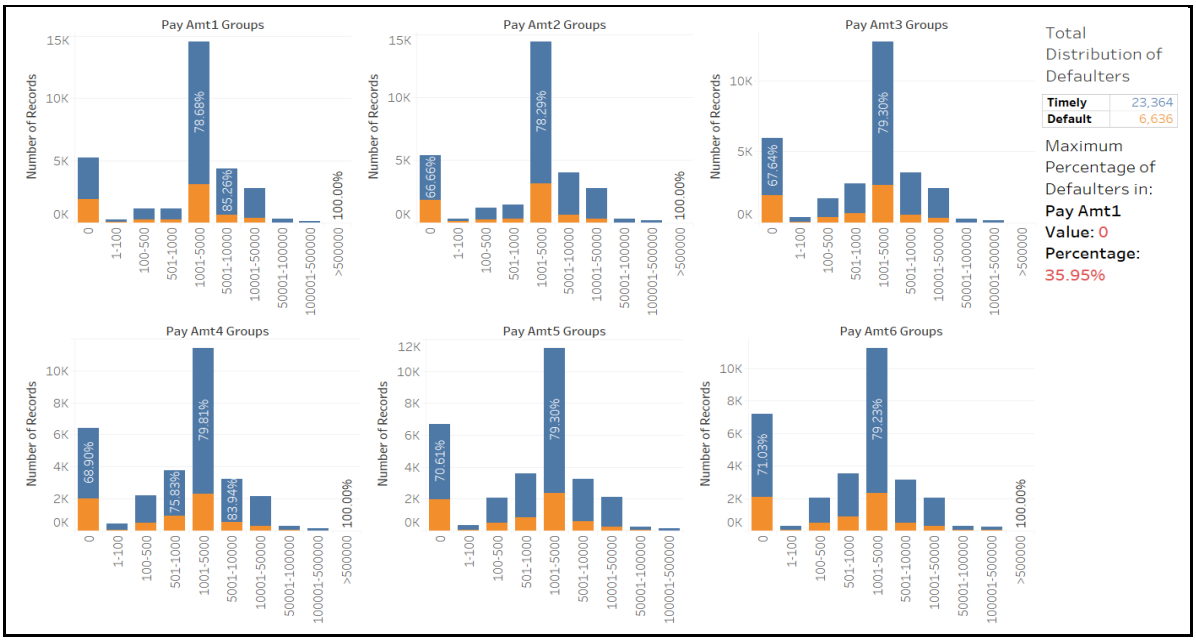
- PAY_0: Repayment status in September, 2005 (-1=pay duly,1=payment delay for one month, 2=payment delay for two months, 8=payment delay for eight months, 9=payment delay for nine months and above)
- PAY_2: Repayment status in August, 2005 (scale same as above)
- PAY_3: Repayment status in July, 2005 (scale same as above)
- PAY_4: Repayment status in June, 2005 (scale same as above)
- PAY_5: Repayment status in May, 2005 (scale same as above)
- PAY_6: Repayment status in April, 2005 (scale same as above)



- BILL_AMT1: Amount of bill statement in September, 2005 (NT dollar)
- BILL_AMT2: Amount of bill statement in August, 2005 (NT dollar)
- BILL_AMT3: Amount of bill statement in July, 2005 (NT dollar)
- BILL_AMT4: Amount of bill statement in June, 2005 (NT dollar)
- BILL_AMT5: Amount of bill statement in May, 2005 (NT dollar)
- BILL_AMT6: Amount of bill statement in April, 2005 (NT dollar)



- PAY_AMT1: Amount of previous payment in September, 2005 (NT dollar)
- PAY_AMT2: Amount of previous payment in August, 2005 (NT dollar)
- PAY_AMT3: Amount of previous payment in July, 2005 (NT dollar)
- PAY_AMT4: Amount of previous payment in June, 2005 (NT dollar)
- PAY_AMT5: Amount of previous payment in May, 2005 (NT dollar)
- PAY_AMT6: Amount of previous payment in April, 2005 (NT dollar)
- default.payment.next.month: Default payment (1=yes, 0=no)



Methods and Processes

Logistic regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes).

In logistic regression, the dependent variable is binary or dichotomous, i.e. it only contains data coded as 1 (TRUE, success, pregnant, etc.) or 0 (FALSE, failure, non-pregnant, etc.).

The goal of logistic regression is to find the best fitting (yet biologically reasonable) model to describe the relationship between the dichotomous characteristic of interest (dependent variable = response or outcome variable) and a set of independent (predictor or explanatory) variables. Logistic regression generates the coefficients (and its standard errors and significance levels) of a formula to predict a *logit transformation* of the probability of presence of the characteristic of interest:

$$logit(p) = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + \ldots + b_kX_k$$

Where **p** is the probability of presence of the characteristic of interest. The logit transformation is defined as the logged odds:

$$odds = \frac{p}{1-p} = \frac{probability\ of\ presence\ of\ characteristic}{probability\ of\ absence\ of\ characteristic}$$

and

$$logit(p) = \ln\left(\frac{p}{1-p}\right)$$

Rather than choosing parameters that minimize the sum of squared errors (like in ordinary regression), estimation in logistic regression chooses parameters that maximize the likelihood of observing the sample values.

Key assumptions needed for a logistic regression model:

Firstly, it does not need a linear relationship between the dependent and independent variables. Logistic regression can handle all sorts of relationships, because it applies a non-linear log transformation to the predicted odds ratio. Secondly, the independent variables do not need to be multivariate normal – although multivariate normality yields a more stable solution. Also, the error terms (the residuals) do not need to be multivariate normally distributed

However, some other assumptions still apply.

Binary logistic regression requires the dependent variable to be binary and ordinal logistic regression requires the dependent variable to be ordinal. Reducing an ordinal or even metric variable to dichotomous level loses a lot of information, which makes this test inferior compared to ordinal logistic regression in these cases.

Secondly, since logistic regression assumes that P(Y=1) is the probability of the event occurring, it is necessary that the dependent variable is coded accordingly. That is, for a binary regression, the factor level 1 of the dependent variable should represent the desired outcome.

Also, the model should be fitted correctly. Neither over fitting nor under fitting should occur. That is only the meaningful variables should be included, but also all meaningful variables should be included. A good approach to ensure this is to use a stepwise method to estimate the logistic regression.

Thirdly, homoscedasticity is not needed. Logistic regression does not need variances to be heteroscedastic for each level of the independent variables. Lastly, it can handle ordinal and nominal data as independent variables. The independent variables do not need to be metric (interval or ratio scaled).

Fourthly, the error terms need to be independent. Logistic regression requires each observation to be independent. That is that the data-points should not be from any dependent samples design, e.g., before-after measurements, or matched pairings. Also the model should have little or no multicollinearity. That is that the independent variables should be independent from each other. However, there is the option to include interaction effects of categorical variables in the analysis and the model. If multicollinearity is present centering the variables might resolve the issue, i.e. deducting the mean of each variable. If this does not lower the multicollinearity, a factor analysis with orthogonally rotated factors should be done before the logistic regression is estimated.

Fifthly, logistic regression assumes linearity of independent variables and log odds. Whilst it does not require the dependent and independent variables to be related linearly, it requires that the independent variables are linearly related to the log odds. Otherwise the test underestimates the strength of the relationship and rejects the relationship too easily, that is being not significant (not rejecting the null hypothesis) where it should be significant. A solution to this problem is the categorization of the independent variables. That is transforming metric variables to ordinal level and then including them in the model. Another approach would be to use discriminant analysis, if the assumptions of homoscedasticity, multivariate normality, and absence of multicollinearity are met.

Lastly, it requires quite large sample sizes. Because maximum likelihood estimates are less powerful than ordinary least squares (e.g., simple linear regression, multiple linear regression); whilst OLS needs 5 cases per independent variable in the analysis, ML needs at least 10 cases per independent variable, some statisticians recommend at least 30 cases for each parameter to be estimated

Data Preparation

1. Checking if there are any missing values.
2. Converting character variables into dummy variables. The character variables in this data set have been already been converted to dummy variables.
3. Checking multicollinearity. This is done through plotting the correlation matrix. No values for this data were greater than 0.9. This implies that there is no multicollinearity.
4. We don't have to check the linearity, normality and homoscedasticity for logistic regression.
5. Splitting the data into training and validation data sets, keeping a constant seed value. The original data has ratio of 8:2 (Non-defaulters: Defaulters), so we maintain the ratio even in the split data sets so that no bias in induced.

Data Analysis

1. Preparing a logistic model using all the variables available in the data, *Model_All_Var.*
   Model_All_Var= glm(train$default.payment.next.month~.,data=train,family = binomial)
2. Execute a step function using direction = "both". *step(Model_All_Var,direction="both")*

Stepwise regression is a method of fitting regression models in which the choice of predictive variables is carried out by an automatic procedure. In each step, a variable is considered for addition to or subtraction from the set of explanatory variables based on some prespecified criterion.
There are 3 main approaches:

- **Forward selection, which involves starting with no variables in the model, testing the addition of each variable using a chosen model fit criterion, adding the variable (if any) whose inclusion gives the most statistically significant improvement of the fit, and repeating this process until none improves the model to a statistically significant extent.**
- **Backward elimination, which involves starting with all candidate variables, testing the deletion of each variable using a chosen model fit criterion, deleting the variable (if any) whose loss gives the most statistically insignificant deterioration of the model fit, and repeating this process until no further variables can be deleted without a statistically significant loss of fit.**
- **Bidirectional elimination, a combination of the above, testing at each step for variables to be included or excluded.**

3. Then we eliminate the insignificant variables using the step function, and run another logistic regression *Model_Sig_Var*
   Model_Sig_Var= glm(formula = train$default.payment.next.month ~ LIMIT_BAL + SEX + EDUCATION + MARRIAGE + AGE + PAY_0 + PAY_2 + PAY_3 + PAY_4 + BILL_AMT1 + BILL_AMT5 + PAY_AMT1 + PAY_AMT2 + PAY_AMT3 + PAY_AMT4 + PAY_AMT5 + PAY_AMT6, family = binomial, data = train)
4. Now check if there is any multicollinearity in the 17 variables that we have used in *Model_Sig_Var.*
   No correlation was greater than 0.8, hence no multicollinearity.
5. Find interaction terms. After finding the most correlated variables using the correlation matrix, use conditional plotting to see if there is any interaction between those corelated variables.
6. Prepare another model with these interaction terms called *Model_Int_Terms.*
   *Model_Int_Terms= glm(formula = train$default.payment.next.month ~ LIMIT_BAL*PAY_0 + LIMIT_BAL*PAY_4 + SEX + EDUCATION + MARRIAGE + AGE + PAY_0*PAY_2 + PAY_2*PAY_3 + PAY_2*PAY_4+ PAY_3*PAY_4 + BILL_AMT1*BILL_AMT5 + PAY_AMT1 + PAY_AMT2 + PAY_AMT3 + PAY_AMT4 + PAY_AMT5 + PAY_AMT6, family = binomial, data = train)*
7. Apply stepwise function on it again to eliminate insignificant variables.
8. Then we prepare another logistic regression model called *Model_PayInt.* This model included PAY_0*PAY_2*PAY_3*PAY_4.
   *Model_PayInt= glm(formula = train$default.payment.next.month ~ LIMIT_BAL*PAY_0 + LIMIT_BAL*PAY_4 + SEX + EDUCATION + MARRIAGE + AGE + PAY_0*PAY_2*PAY_3*PAY_4 + BILL_AMT1*BILL_AMT5 + PAY_AMT1 + PAY_AMT2 + PAY_AMT3 + PAY_AMT4 + PAY_AMT5 + PAY_AMT6, family = binomial, data = train)*
9. We created another model *Model_LimitPay_Int.* This model used LIMIT_BAL*PAY_0*PAY_2*PAY_3*PAY_4.
   Model_LimitPay_Int= glm(formula = train$default.payment.next.month ~ SEX + EDUCATION + MARRIAGE + AGE + LIMIT_BAL PAY_0*PAY_2*PAY_3*PAY_4 + BILL_AMT1*BILL_AMT5 + PAY_AMT1 + PAY_AMT2 + PAY_AMT3 + PAY_AMT4 + PAY_AMT5 + PAY_AMT6, family = binomial, data = train)*
10. We create the last model *Model_Overfit* in which we have excluded individual PAY and LIMIT_BAL variables, we just use the interaction term LIMIT_BAL: PAY_0:PAY_2: PAY_3:PAY_4.
    Model_Overfit= glm(formula = train$default.payment.next.month ~ SEX + EDUCATION + MARRIAGE + AGE + LIMIT_BAL: PAY_0:PAY_2:PAY_3:PAY_4 + BILL_AMT1:BILL_AMT5 + PAY_AMT1 + PAY_AMT2 + PAY_AMT3 + PAY_AMT4 + PAY_AMT5 + PAY_AMT6, family = binomial, data = train)

Data Validation

We used the confusion Matrix to validate the data. It shows us that how many "1s" were predicted correctly, how many of the defaulter predictions were right.

We also implemented a small code that calculated the percentage of right predictions.

$$\frac{Number\ of\ "1"s\ Predicted}{Actual\ Number\ of\ "1"s} * 100$$

While calculating the number of "1"s predicted correctly, we had to apply a filter of the probability value. At 0.75, 0.65 and 0.85 probability the *Model_LimitPay_Int* gives satisfactory results, 0.75 being the most accurate one. 0.9/0.5 probability threshold will not be a good judge for the model.

We prepared a different validation set using a different seed value, naming it Test1. Then we used this dataset for testing Overfitting.

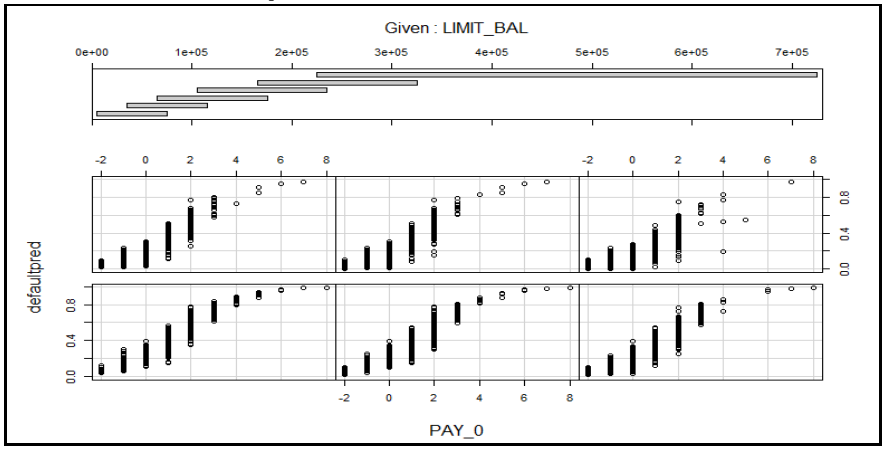
Results

Correlation matrix obtained after generating the model Model_Sig_Var

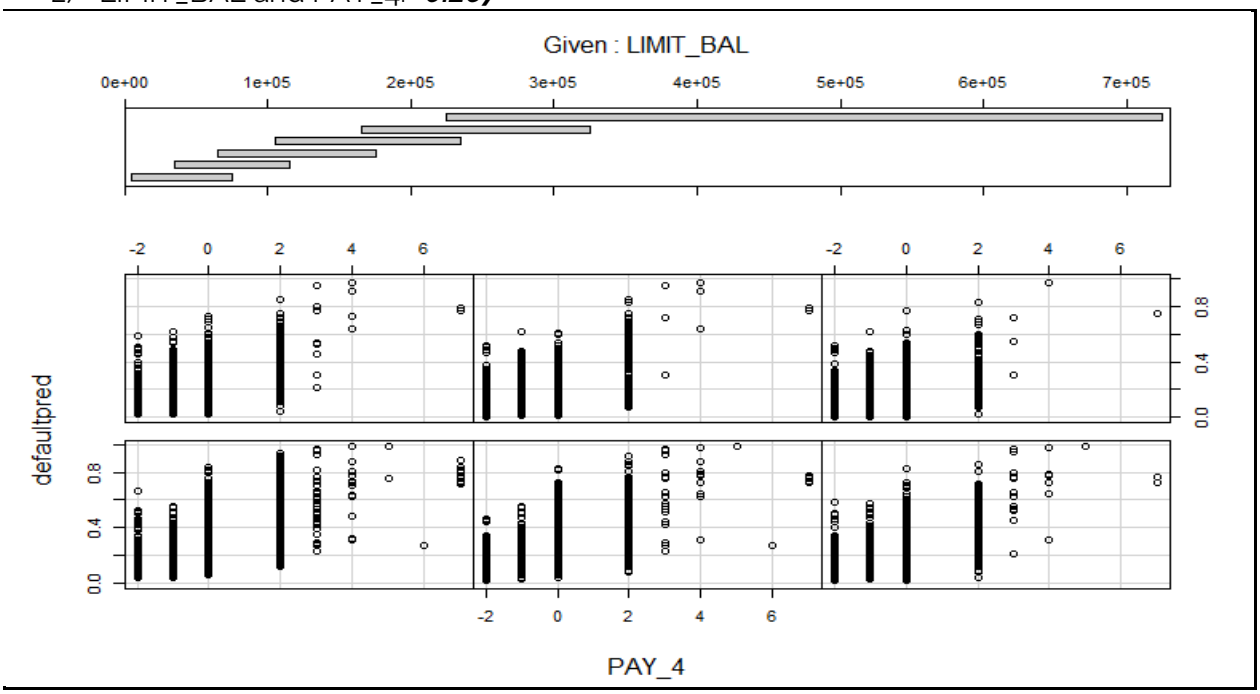
	dataset.LI	dataset.SE	dataset.EI	dataset.M	dataset.A(	dataset.P/	dataset.P/	dataset.P/	dataset.P/	dataset.BI	dataset.BI	dataset.P/	dataset.P/	dataset.P/	dataset.P/	dataset.P/
dataset.LIMIT_BAL	0.024755	-0.21916	-0.10814	0.144713	-0.27121	-0.29638	-0.28612	-0.26746	0.28543	0.295562	0.195236	0.178408	0.210167	0.203242	0.217202	0.219595
dataset.SE	0.024755		0.014232	-0.03139	-0.09087	-0.05764	-0.07077	-0.0661	-0.06017	-0.03364	-0.01701	-0.00024	-0.00139	-0.0086	-0.00223	-0.00167	-0.00277
dataset.EI	-0.21916	0.014232		-0.14346	0.175061	0.105364	0.121566	0.114025	0.108793	0.023581	-0.00757	-0.03746	-0.03004	-0.03994	-0.03822	-0.04036	-0.0372
dataset.M	-0.10814	-0.03139	-0.14346		-0.41417	0.019917	0.024199	0.032688	0.033122	-0.02347	-0.02539	-0.00598	-0.00809	-0.00354	-0.01266	-0.0012	-0.00664
dataset.A(	0.144713	-0.09087	0.175061	-0.41417		-0.03945	-0.05015	-0.05305	-0.04972	0.056239	0.049345	0.026147	0.021785	0.029247	0.021379	0.02285	0.019478
dataset.P/	-0.27121	-0.05764	0.105364	0.019917	-0.03945		0.672164	0.574245	0.538841	0.187068	0.180635	-0.07927	-0.0701	-0.07056	-0.064	-0.05819	-0.05867
dataset.P/	-0.29638	-0.07077	0.121566	0.024199	-0.05015	0.672164		0.766552	0.662067	0.234887	0.221348	-0.0807	-0.05899	-0.0559	-0.04686	-0.03709	-0.0365
dataset.P/	-0.28612	-0.0661	0.114025	0.032688	-0.05305	0.574245	0.766552		0.777359	0.208473	0.225145	0.001295	-0.06679	-0.05331	-0.04607	-0.03586	-0.03586
dataset.P/	-0.26746	-0.06017	0.108793	0.033122	-0.04972	0.538841	0.662067	0.777359		0.202812	0.242902	-0.00936	-0.00194	-0.06924	-0.04346	-0.03359	-0.02657
dataset.BI	0.28543	-0.03364	0.023581	-0.02347	0.056239	0.187068	0.234887	0.208473	0.202812		0.829779	0.140277	0.099355	0.156887	0.158303	0.167026	0.179341
dataset.BI	0.295562	-0.01701	-0.00757	-0.02539	0.049345	0.180635	0.221348	0.225145	0.242902	0.829779		0.217031	0.181246	0.252305	0.293118	0.141574	0.164184
dataset.P/	0.195236	-0.00024	-0.03746	-0.00598	0.026147	-0.07927	-0.0807	0.001295	-0.00936	0.140277	0.217031		0.285576	0.252191	0.199558	0.148459	0.185735
dataset.P/	0.178408	-0.00139	-0.03004	-0.00809	0.021785	-0.0701	-0.05899	-0.06679	-0.00194	0.099355	0.181246	0.285576		0.24477	0.180107	0.180908	0.157634
dataset.P/	0.210167	-0.0086	-0.03994	-0.00354	0.029247	-0.07056	-0.0559	-0.05331	-0.06924	0.156887	0.252305	0.252191	0.24477		0.216325	0.159214	0.16274
dataset.P/	0.203242	-0.00223	-0.03822	-0.01266	0.021379	-0.064	-0.04686	-0.04607	-0.04346	0.158303	0.293118	0.199558	0.180107	0.216325		0.15183	0.157834
dataset.P/	0.217202	-0.00167	-0.04036	-0.0012	0.02285	-0.05819	-0.03709	-0.03586	-0.03359	0.167026	0.141574	0.148459	0.180908	0.159214	0.15183		0.154896
dataset.P/	0.219595	-0.00277	-0.0372	-0.00664	0.019478	-0.05867	-0.0365	-0.03586	-0.02657	0.179341	0.164184	0.185735	0.157634	0.16274	0.157834	0.154896	

The highlighted cells indicate the 5 most positively and negatively correlated variables each.
Then we generated the Coplots for these 10 pairs, and identified 7 considerable pairs. Which are listed below:

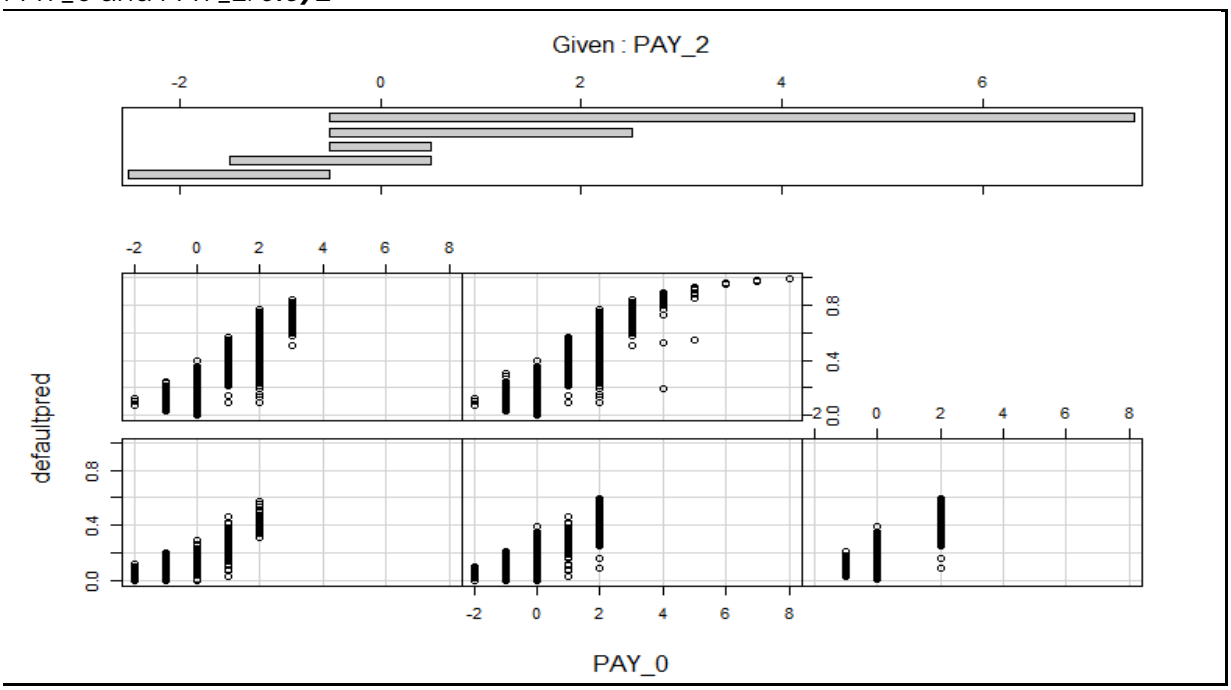
1) PAY_0 and LIMIT_Bal: ***-0.271***



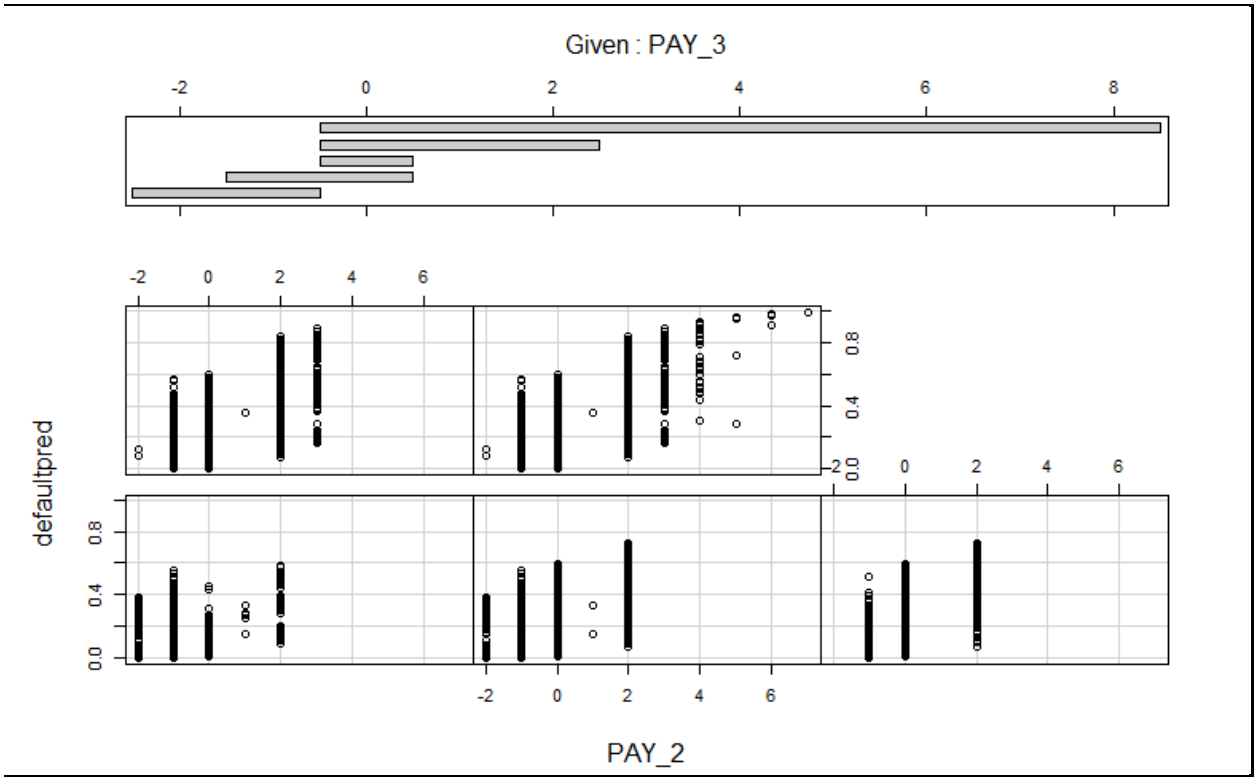
2) LIMIT_BAL and PAY_4: ***-0.267***



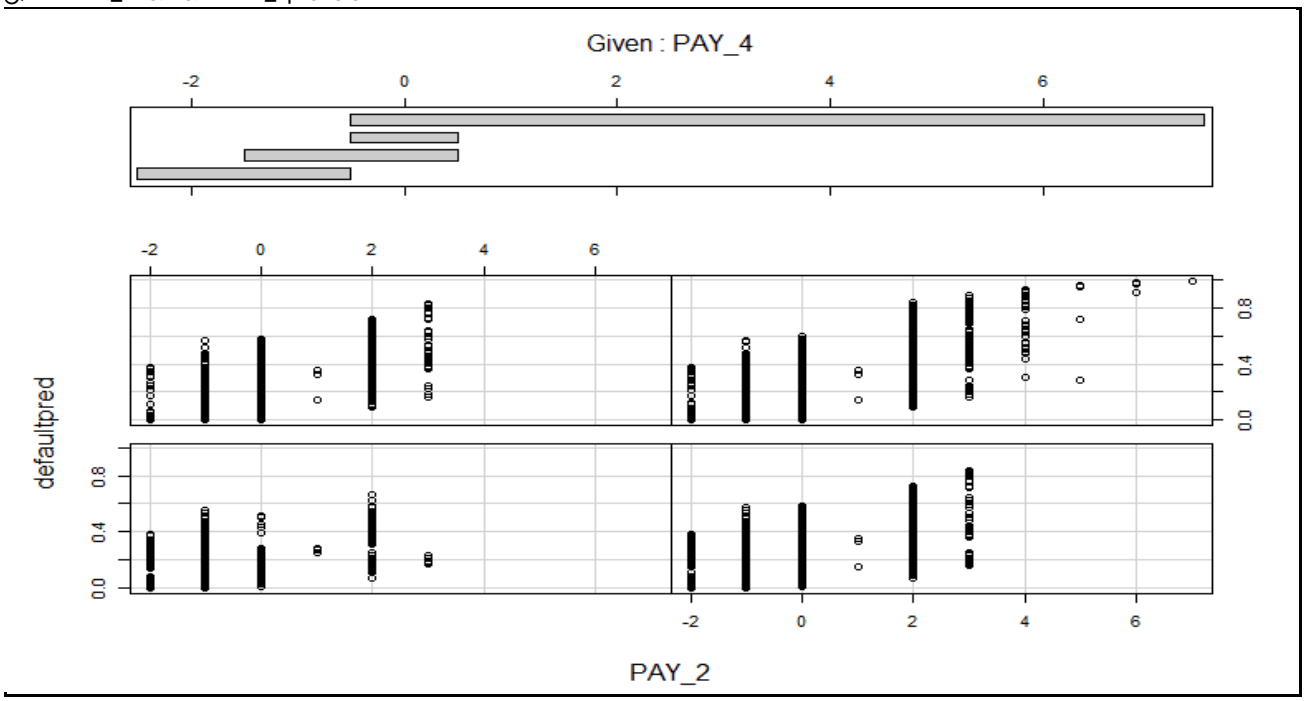
3) PAY_0 and PAY_2: ***0.672***



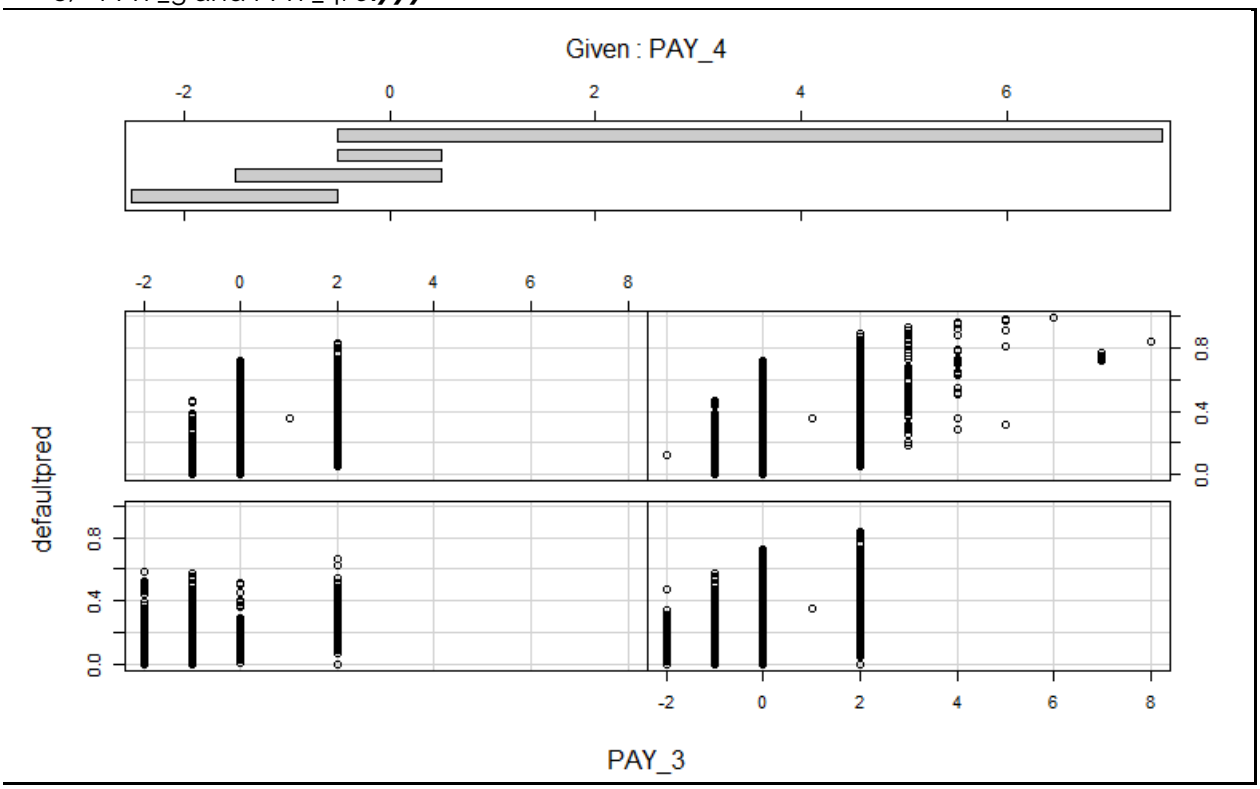
4) PAY_2 and PAY_3: ***0.766***



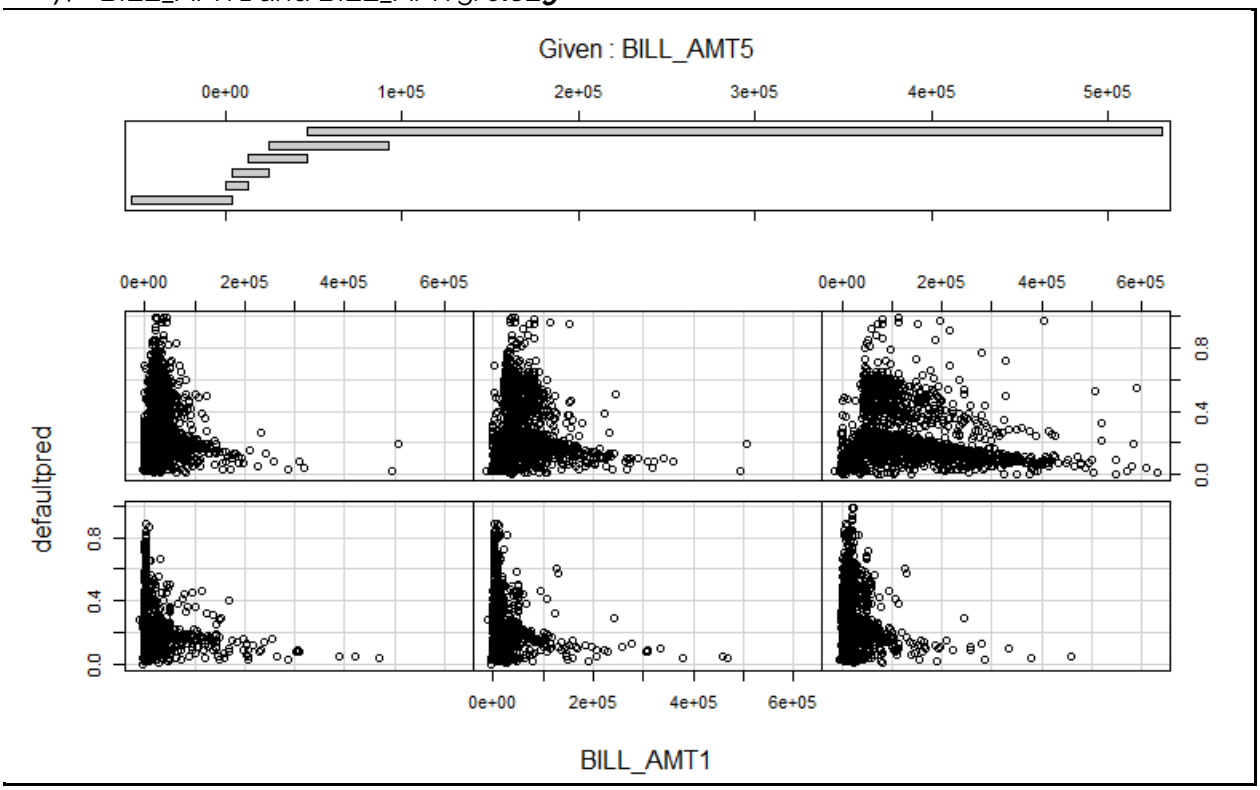
5) PAY_2 and PAY_4: ***0.662***



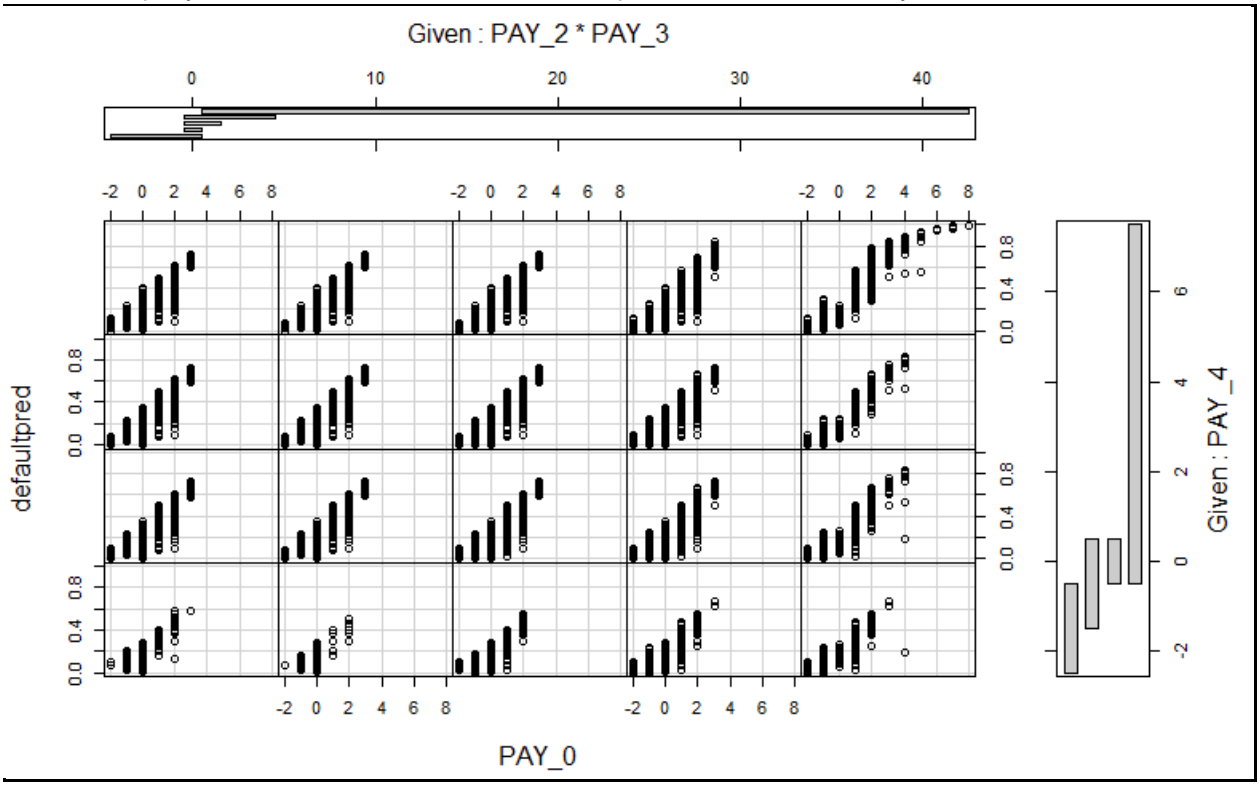
6) PAY_3 and PAY_4: ***0.777***



7) BILL_AMT1 and BILL_AMT5: ***0.829***



Based on these plots and correlations we generated a model called Model_Int_Terms. Another proposal for an interaction term is PAY_0*PAY_2*PAY_3*PAY_4. The coplot for this term is displayed below and it has been incorporated in Model_PayInt.



This table does a comparison of all the Models created and tested:

Name	Model_All_Var	Model_Sig_Var	Model_Int_Terms	Model_PayInt	Model_LimitPay_Int	Model_Overfit
AIC Value	20117	20106	19869	19521	**19452**	22189
Accuracy (at 0.75 Cut-off)	72.84	72.84	76.39	75.78	81.57	**83.33**
Neg. Pred. Value (0.75)	72.5	72.5	76.22	75.59	**81.42**	80
Accuracy (at 0.50 Cut-off)	73.94	**73.71**	70.42	69.39	69.35	68.75
Neg. Pred. Value (0.50)	73.9	**73.68**	70.38	69.36	69.32	66.67
Accuracy (at 0.90 Cut-off)	60	60	70.42	75	78.57	**100**
Neg. Pred. Value (0.90)	58.3	58.3	70	73.68	77.78	**100**
Accuracy (at 0.75 Cut-off) on Test1	73.84	73.85	75.75	75.83	**76.58**	60
Neg. Pred. Value (0.75) on Test1	73.43	73.43	75.57	75.63	**76.36**	50

Conclusions

1) Descriptive Analytics

In conclusion, the data exploration of credit card default dataset shows

- No Missing values were found.
- larger percentage of females than males in default payment category
- percentage of customers having graduate/Uni school degree is higher in default payment category.
- individuals having single status have higher percentage of default than married
- age peaks around 28-29 years in default payment category.
- default rate increases during the data collection from April 2005 to September 2005.
- As per the data collection, the number of defaulters lies in the range of 20±5 % usually across all predictors.
- No direct bias was found in the dataset.
- Co-plot generations show multiple interactions exist among the predictors. Also, a correlation matrix gives significant values for some combinations.
- Since logistic regression requires factors rather than character label values, the dataset is found appropriate in this regard.
- No Multicollinearity was found in the data.

2) Regression Analytics

We tested the accuracy of all the models on both Test and Test1 datasets. Model_Overfit shows very good accuracy in some cases, whereas does not perform very well in certain conditions. This clearly indicates that the model has a very erratic behavior across different cut offs and different data sets. As shown in the table above, it shows a 100% negative prediction value at 0.9 threshold (Test data) and then gives a 50% negative prediction value for Test1 data. This implies that Model_Overfit is a classic example of *Model Overfitting*. Model_LimitPay_Int emerges as the best model after due analysis of the results. It generally performs better than all the other models across different thresholds and testing data sets.

Looking at the P-Values we infer that the 5 most significant parameters while a default payment for the next month will be (in order of significance):

I. Last known payment status (PAY_0)
II. Credit limit of the customer (LIMIT_BAL)
III. Last known Bill Statement (BILL_AMT1)
IV. Marital Status of the consumer (MARRIAGE)
V. Age of the customer (AGE)

We also found 6 insignificant variables (apart from ID), they are listed below:

I. Pay 5
II. Pay 6
III. Bill_Amt 2
IV. Bill_Amt 3
V. Bill_Amt 4
VI. Bill_Amt 6