Lending Club – Loan Repayment Prediction – Literature Review

Introduction

The goal of each and every business is to make profit. For a lender profit depends on whether or not the borrower pays the interest and the principal. Without repayment the lender will incur a loss and that loss can even potentially be greater than the initial loan when lawyer, court and collection fees are taken into consideration. For these reasons it is critically important for a lender to be able to identify whether a potential borrower can and will make all of his or her loan payments. What I intend to do is identify the characteristics (limited to those found in the dataset) of persons, as well as the causes for which people default on their loans and use this information to predict whether a potential borrower would or would not make all of his or her own payments. In order to make these predictions, I will determine the most significant features in the Learning Club dataset and pass these features along with the corresponding labels (default/no-default) into the machine learning classifiers I select, creating predictive models for each. For each model I will test with the same test data set and compare the test labels to the actual labels to measure accuracy.

Literature Review

Benchmarking state-of-the-art classification algorithms for credit scoring: A ten-year update

http://www.business-school.ed.ac.uk/waf/crc_archive/2013/42.pdf

Essentially this paper is a modern update to the landmark benchmarking study of classification algorithms for credit scoring by Baesens, Van Gestel, Viaene, Stepanova, Suykens and Vanthienen in 2003. Since 2003 many new techniques and algorithms have been developed in predictive modeling. This paper builds upon Baesens et al. by including all of the newer state-of-the-art techniques as well as those covered in the previous study.

I chose this paper because of its goal of comparing of classification algorithms for credit scoring is highly related to the problem I wish to solve in this project. The paper describes a vector x of m dimensions with each dimension as a feature characterizing an application for a credit products such as a loan. It then goes on to discuss a binary response variable which indicates the existence or non-existence of a default event. The probability of a default event given x is classification problem being addressed in the study. Finally, a decision maker will take this probability and if it falls under a given threshold the application will be accepted, otherwise it will be rejected. This is essentially the approach I am taking for this capstone project. The Lending Club data set I am using for this project contains over 100 features characterizing the borrower, my vector x. The data set also contains a feature stating the current status of the loan with various possible values that can easily be grouped into default or non-default statuses. This is effectively my binary response variable described in the study. Finally my goal is to estimate the probability of default given a set of borrower characteristics and use that to determine whether they are likely or unlikely to default. This is parallel to the study.

The study considered the following classification algorithms.

TABLE 2: CLASSIFICATION ALGORITHMS CONSIDERED IN THE BENCHMARKING STUDY

	Base model selection	Classification algorithm	Acronym	Number of models[1]
Individual classifier	n.a.	Bayesian Network	B-Net	4
		CART	CART	10
		Extreme learning machine	ELM	120
		Kernalized ELM	ELM-K	200
		k-nearest neighbor	kNN	22
		J4.8	J4.8	36
		Linear discriminant analysis[2]	LDA	1
		Linear support vector machine	SVM-L	29
		Logistic regression[2]	LR	1
		Multilayer perceptron artificial neural network	ANN	171

		Naive Bayes	NB	1
		Quadratic discriminant analysis[2]	QDA	1
		Radial basis function neural network	RbfNN	5
		Regularized logistic regression	LR-R	27
		SVM with radial basis kernel function	SVM- Rbf	300
		Voted perceptron	VP	5
	Classification models from individual classifiers		**16**	**933**
Homogenous ensembles	n.a.	Alternating decision tree	ADT	5
		Bagged decision trees	Bag	9
		Bagged MLP	BagNN	4
		Boosted decision trees	Boost	48
		Logistic model tree	LMT	1
		Random forest	RF	30
		Rotation forest	RotFor	25
		Stochastic gradient boosting	SGB	9
	Classification models from homogeneous ensembles		**8**	**131**
Heterogeneous ensembles	n.a.	Simple average ensemble	AvgS	1
		Weighted average ensemble	AvgW	1
	Static direct	Complementary measure	CompM	4
		Ensemble pruning via reinforcement learning	EPVRL	4
		GASEN	GASEN	4
		Hill-climbing ensemble selection	HCES	12
		HCES with bootstrap sampling	HCES-Bag	16
		Matchting pursuit optimization of ensemble classifiers	MPOCE	1
		Stacking	Stack	6
		Top-T ensemble	Top-T	12
	Static indirect	Clustering using compound error	CuCE	1
		k-Means clustering	k-Means	1
		Kappa pruning	KaPru	4
		Margin distance minimization	MDM	4
		Uncertainty weighted accuracy	UWA	4
	Dynamic	Probabilistic model for classifier competence	PMCC	1
		k-nearest oracle	kNORA	1
	Classification models from heterogeneous ensembles		**17**	**77**
Overall number of classification algorithms and models			**41**	**1141**

The following table measures the performance of each algorithm in credit scoring classification using Area Under a ROC Curve (AUC). According to the study across all performance measures the top three most accurate classifiers are

Random Forests, Bagged (MLP) Neural Networks, and Bagged Decision Trees.

TABLE 5: PERFORMANCE OF INDIVIDUAL CLASSIFIERS AND HOMOGENEOUS ENSEMBLES IN TERMS OF THE AUC

		AC		GC		Bene1		Bene2		UK		PAK		GMC	
Individual classifiers	ANN	.926	(.011)	.791	(.014)	.791	(.009)	.802	(.005)	.742	(.008)	.644	(.004)	.859	(.003)
	B-Net	.922	(.011)	.764	(.015)	.771	(.009)	.786	(.009)	.703	(.023)	.623	(.004)	.860	(.003)
	CART	.856	(.019)	.706	(.031)	.706	(.021)	.713	(.021)	.684	(.012)	.565	(.015)	.797	(.025)
	ELM	.911	(.011)	.778	(.012)	.766	(.010)	.761	(.006)	.650	(.009)	.599	(.003)	.717	(.004)
	ELM-K	.926	(.012)	.794	(.015)	.787	(.007)	.788	(.005)	.734	(.009)	.643	(.004)	.702	(.004)
	J4.8	.915	(.014)	.734	(.020)	.761	(.012)	.747	(.011)	.500	(.000)	.500	(.000)	.500	(.000)
	k-NN	.906	(.016)	.772	(.010)	.765	(.009)	.754	(.007)	.725	(.014)	.600	(.005)	.739	(.004)
	LDA	.929	(.009)	.784	(.012)	.775	(.011)	.779	(.008)	.715	(.010)	.626	(.003)	.692	(.004)
	LR	**.931**	(.011)	.784	(.012)	.773	(.012)	.791	(.006)	.720	(.011)	.626	(.003)	.693	(.005)
	LR-R	.925	(.012)	.778	(.015)	.787	(.007)	.798	(.004)	.690	(.012)	.635	(.004)	.623	(.006)
	NB	.893	(.020)	.777	(.017)	.747	(.013)	.724	(.010)	.701	(.019)	.613	(.006)	.671	(.003)
	RbfNN	.902	(.019)	.762	(.013)	.760	(.009)	.739	(.007)	.701	(.014)	.604	(.003)	.755	(.007)
	QDA	.917	(.018)	.674	(.148)	.765	(.011)	.780	(.006)	.703	(.012)	.612	(.004)	.811	(.003)
	SVM-L	.924	(.013)	.782	(.014)	.786	(.007)	.796	(.003)	.659	(.014)	.636	(.004)	.733	(.017)
	SVM-Rbf	.926	(.012)	.799	(.011)	.786	(.008)	.795	(.004)	.666	(.028)	.630	(.004)	.815	(.009)
	VP	.810	(.030)	.680	(.020)	.698	(.013)	.621	(.017)	.554	(.018)	.567	(.003)	.568	(.024)
Homogeneous ensemble classifiers	ADT	.929	(.010)	.758	(.012)	.786	(.008)	.794	(.010)	.732	(.008)	.641	(.004)	.860	(.004)
	Bag	.930	(.014)	.788	(.014)	.794	(.008)	.805	(.006)	.742	(.007)	.643	(.003)	**.864**	(.003)
	BagNN	.927	(.012)	**.802**	(.010)	.793	(.008)	.802	(.004)	**.745**	(.008)	**.646**	(.004)	.838	(.004)
	Boost	.930	(.010)	.772	(.012)	**.795**	(.007)	**.808**	(.005)	.741	(.010)	.643	(.004)	.860	(.003)
	LMT	.930	(.013)	.747	(.015)	.780	(.007)	.787	(.006)	.720	(.010)	.630	(.004)	.833	(.017)
	RF	.931	(.014)	.789	(.013)	.794	(.008)	.805	(.006)	.742	(.007)	.643	(.003)	**.864**	(.003)
	RotFor	.929	(.013)	.773	(.015)	.788	(.007)	.794	(.007)	.502	(.016)	.635	(.002)	.820	(.005)
	SGB	.928	(.013)	.751	(.015)	.786	(.007)	.797	(.006)	.735	(.012)	.642	(.004)	.860	(.003)

An Empirical Comparison of Supervised Learning Algorithms

https://www.cs.cornell.edu/~caruana/ctp/ct.papers/caruana.icml06.pdf

This study compares the performance of eight machine learning algorithms namely, SVMs, neural nets, logistic regression, naïve bayes, memory based learning, random forests, decision trees, bagged trees, boosted trees, and boosted stumps. The performance metrics used are, accuracy, F-score, Lift, ROC Area, average precision, squared error and cross entropy. The study concludes that bagged trees, random forests and neural nets have the best average performance (prior to calibration) over all the metrics and over all the problems. When calibration is taken into account, the overall best performing algorithm is boosted decision trees (calibrated). In close second is Random forests, followed by bagged decision trees (uncalibrated).

Table 3. Normalized scores of each learning algorithm by problem (averaged over eight metrics)

MODEL	CAL	COVT	ADULT	LTR.P1	LTR.P2	MEDIS	SLAC	HS	MG	CALHOUS	COD	BACT	MEAN
BST-DT	PLT	**.938**	.857	**.959**	**.976**	.700	.869	**.933**	.855	**.974**	**.915**	.878*	**.896***
RF	PLT	.876	.930	.897	.941	**.810**	.907*	.884	.883	.937	.903*	.847	.892
BAG-DT	–	.878	.944*	.883	.911	.762	.898*	.856	**.898**	.948	.856	**.926**	.887*
BST-DT	ISO	.922*	.865	.901*	.969	.692*	.878	.927	.845	.965	.912*	.861	.885*
RF	–	.876	.946*	.883	.922	.785	.912*	.871	.891*	.941	.874	.824	.884
BAG-DT	PLT	.873	.931	.877	.920	.752	.885	.863	.884	.944	.865	.912*	.882
RF	ISO	.865	.934	.851	.935	.767*	**.920**	.877	.876	.933	.897*	.821	.880
BAG-DT	ISO	.867	.933	.840	.915	.749	.897	.856	.884	.940	.859	.907*	.877
SVM	PLT	.765	.886	.936	.962	.733	.866	.913*	.816	.897	.900*	.807	.862
ANN	–	.764	.884	.913	.901	.791*	.881	.932*	.859	.923	.667	.882	.854
SVM	ISO	.758	.882	.899	.954	.693*	.878	.907	.827	.897	.900*	.778	.852
ANN	PLT	.766	.872	.898	.894	.775	.871	.929*	.846	.919	.665	.871	.846
ANN	ISO	.767	.882	.821	.891	.785*	.895	.926*	.841	.915	.672	.862	.842
BST-DT	–	.874	.842	.875	.913	.523	.807	.860	.785	.933	.835	.858	.828
KNN	PLT	.819	.785	.920	.937	.626	.777	.803	.844	.827	.774	.855	.815
KNN	–	.807	.780	.912	.936	.598	.800	.801	.853	.827	.748	.852	.810
KNN	ISO	.814	.784	.879	.935	.633	.791	.794	.832	.824	.777	.833	.809
BST-STMP	PLT	.644	**.949**	.767	.688	.723	.806	.800	.862	.923	.622	.915*	.791
SVM	–	.696	.819	.731	.860	.600	.859	.788	.776	.833	.864	.763	.781
BST-STMP	ISO	.639	.941	.700	.681	.711	.807	.793	.862	.912	.632	.902*	.780
BST-STMP	–	.605	.865	.540	.615	.624	.779	.683	.799	.817	.581	.906*	.710
DT	ISO	.671	.869	.729	.760	.424	.777	.622	.815	.832	.415	.884	.709
DT	–	.652	.872	.723	.763	.449	.769	.609	.829	.831	.389	.899*	.708
DT	PLT	.661	.863	.734	.756	.416	.779	.607	.822	.826	.407	.890*	.706
LR	–	.625	.886	.195	.448	.777*	.852	.675	.849	.838	.647	.905*	.700
LR	ISO	.616	.881	.229	.440	.763*	.834	.659	.827	.833	.636	.889*	.692
LR	PLT	.610	.870	.185	.446	.738	.835	.667	.823	.832	.633	.895	.685
NB	ISO	.574	.904	.674	.557	.709	.724	.205	.687	.758	.633	.770	.654
NB	PLT	.572	.892	.648	.561	.694	.732	.213	.690	.755	.632	.756	.650
NB	–	.552	.843	.534	.556	.011	.714	-.654	.655	.759	.636	.688	.481

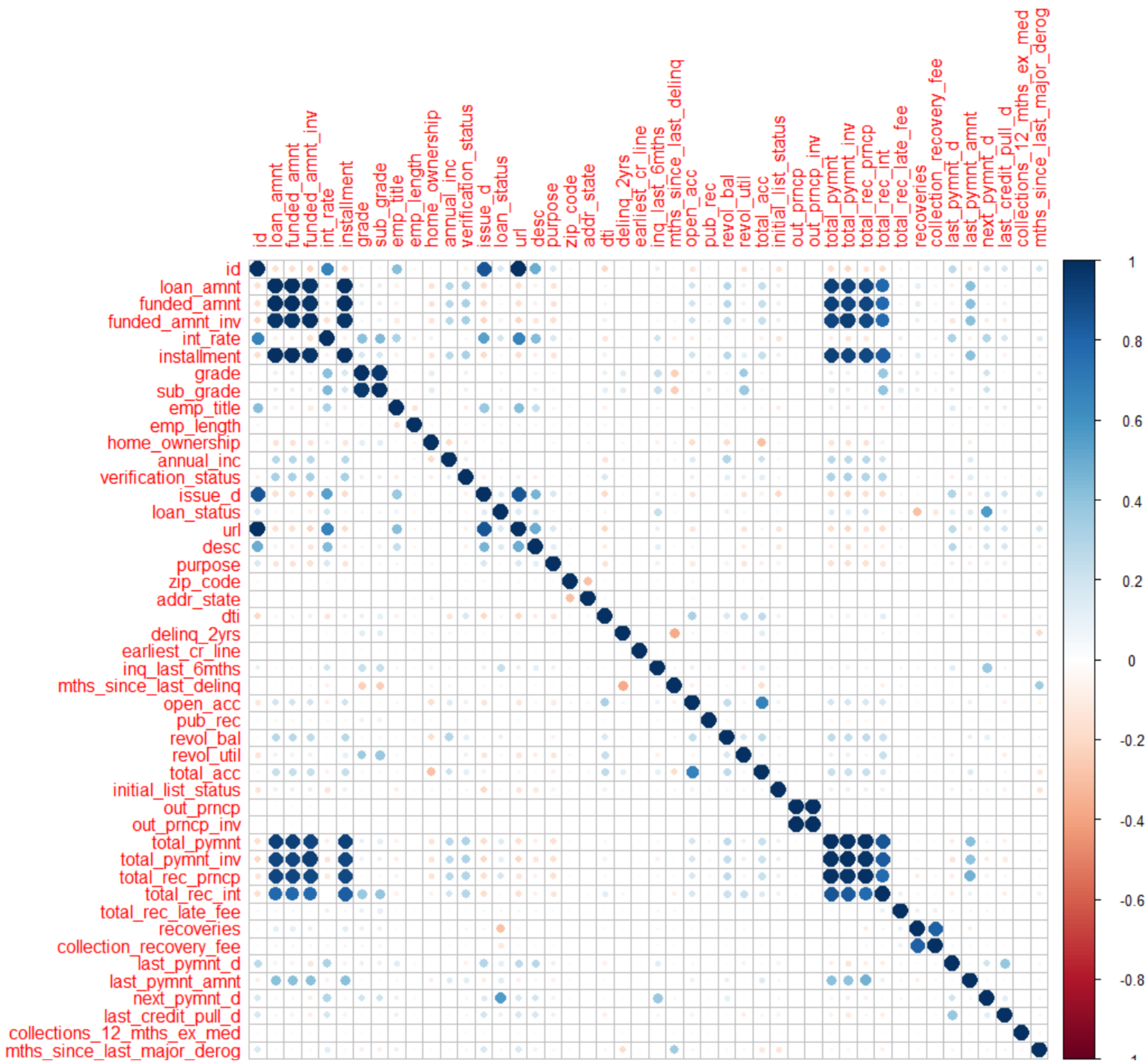
Conclusion

I was fascinated to discover that from both of the studies I researched, ensemble methods were generally the best performers. I was also pleasantly surprised to find that in both studies, the top three performing algorithms were almost identical. According to **Benchmarking state-of-the-art classification algorithms for credit scoring: A ten-year update**, the top three classifiers were Random Forests, Bagged (MLP) Neural Networks, and Bagged Decision Trees. Meanwhile according to **An Empirical Comparison of Supervised Learning Algorithms**, the top three are Boosted Decision Trees, Random Forests and Bagged Decision Trees. In both studies Random Forests and Bagged Decision Trees come out on top. It's important to note that **An Empirical Comparison of Supervised Learning Algorithms** did not include Bagged Neural Networks. Given the agreement among both studies I have decided to use Random Forests, Bagged (MLP) Neural Networks, and Bagged Decision Trees for this project. In addition to selecting classification algorithms, metrics for measuring the performance of said classifiers are required. Given the similarities of the goals of my project and the aims of **Benchmarking state-of-the-art classification algorithms for credit scoring: A ten-year update**, I have decided to use its performance measures. Specifically, the performance metrics for this project are Percentage Correctly Classified (PCC), the area under a ROC curve (AUC), H-measure, and Brier Score (BS).

Dataset

The dataset for this project is the Loan Data dataset from the Lending Club (https://www.lendingclub.com/info/download-data.action). Only 36 month term data from 2007 to February 2013 is being used because it is important that all loans examined are well past their due date. A borrower which defaults on a current loan may do so due to a temporary job loss or a myriad of other reasons. The borrower may then recover and return to good standing once again. Using 36 month term data up until February 2013, ensures that labelling potentially temporary defaults as permanent does not occur.

The original dataset has 111 features. The first step was to review the data and remove any features consisted of majority NULL values. Once those features were removed a correlation analysis was performed. Of each set of highly correlated features, one was removed. The image below displays the correlation plot before the removal of highly correlated features.



The next step was to determine which of the remaining features were significant. Two methods were employed: the Boruta algorithm and the Recursive Feature Elimination algorithm (RFE).

Boruta:

After running Boruta four separate times each on a unique set of random records from the dataset, the following results were found. Only features/attributes which were deemed important in 50% or more of the Boruta runs were ultimately selected as important.

22 features/attributes deemed important

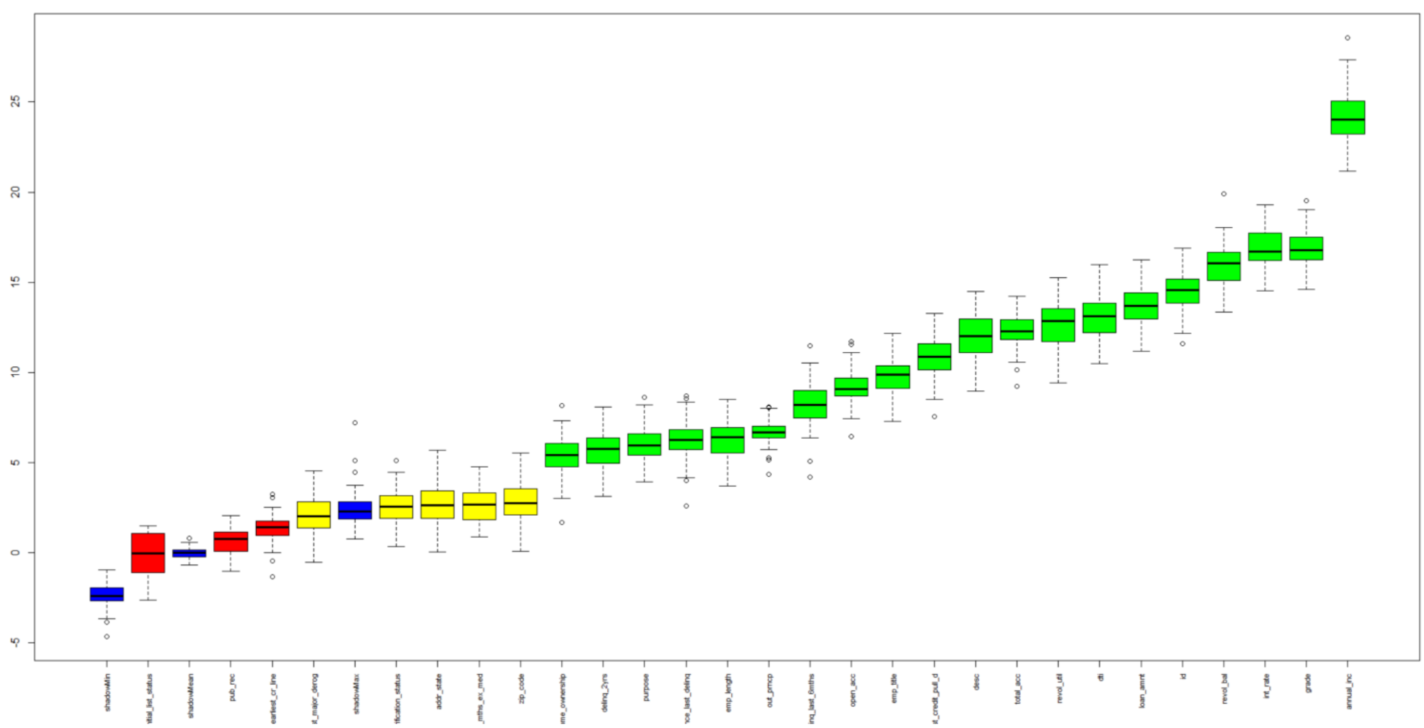
2 features/attributes deemed unimportant

4 features/attributes with tentative importance

The following features/attributes were found to be significantly important in 50% or more of the Boruta analyses:

1. id
2. loan_amnt
3. int_rate
4. grade
5. emp_title
6. emp_length
7. home_ownership
8. annual_inc
9. verification_status
10. desc
11. purpose
12. dti
13. zip_code
14. delinq_2yrs
15. inq_last_6mths
16. mths_since_last_delinq
17. open_acc
18. revol_bal
19. revol_util
20. total_acc
21. out_prncp
22. last_credit_pull_d

The graph below displays the importance of each of the features. It was taken from one of the four iterations of the algorithm and generally reflects the overall result. Green = Significant importance, Yellow = Tentative importance and Red = Insignificant importance



Recursive Feature Elimination (RFE):

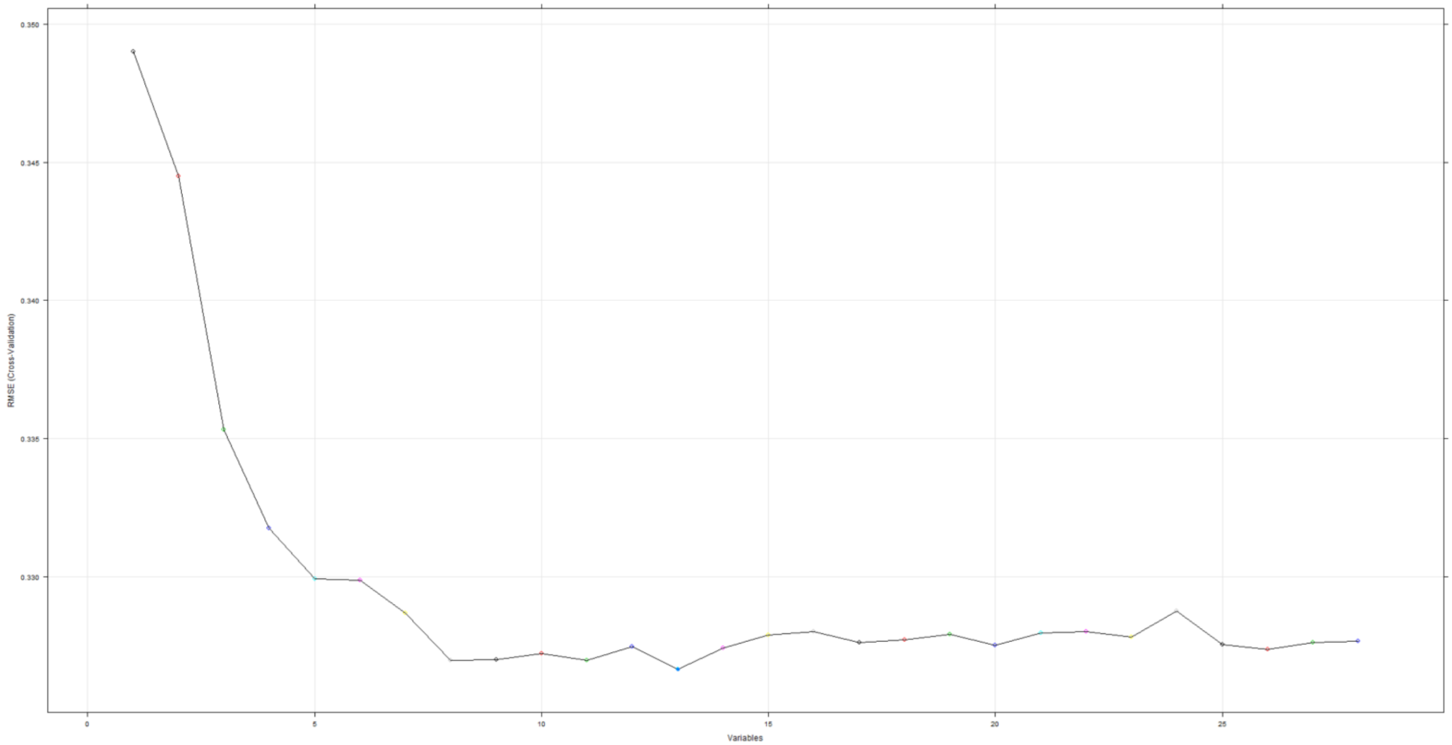
According to the RFE algorithm, there are 17 features/attributes of significant importance. These features are listed below along with their overall significance.

```
                      Overall
annual_inc          20.766586
revol_bal           13.414077
int_rate            12.425477
id                  12.326874
emp_title           11.834513
loan_amnt            9.727355
grade                9.581749
total_acc            7.796833
revol_util           6.757664
open_acc             6.715921
purpose              6.253235
dti                  6.137568
last_credit_pull_d   5.895808
emp_length           5.670145
inq_last_6mths       5.322751
home_ownership       4.888201
desc                 4.841160
```

It is important to note that all of the significant features selected by RFE were also deemed as significant by the Boruta algorithm. Of the 16 deemed important, the RFE algorithm selected an optimal subset of 13 features.

1. annual_inc
2. revol_bal
3. int_rate
4. id
5. emp_title
6. loan_amnt
7. grade
8. total_acc
9. revol_util
10. dti
11. open_acc
12. last_credit_pull_d
13. purpose

According to the RFE algorithm, the top feature/variable is **annual_inc**. This corresponds to the Boruta analysis. The diagram below shows the relationship between the number of variables/features to and Root Mean Square Error (RMSE).



In conclusion, both the Boruta and RFE algorithms found a similar set of significant set of features/variables. The main difference being that Boruta tended to find more features/variables significant than RFE. One advantage of RFE was the ability to determine the optimal subset of features/variables which would minimize the RMSE. As such, I will be using the set of 13 optimal features/variables as listed above.

Below is the Data Dictionary, describing each feature and whether or not it will be used for analysis. Note: Only those features labeled as '**Significant and part of optimal subset**' will be used for analysis.

Data Dictionary:

Feature	Description
acc_now_delinq	The number of accounts on which the borrower is now delinquent.
acc_open_past_24mths	Number of trades opened in past 24 months.
addr_state	The state provided by the borrower in the loan application
all_util	Balance to credit limit on all trades
annual_inc	The self-reported annual income provided by the borrower during registration.
annual_inc_joint	The combined self-reported annual income provided by the co-borrowers during registration
application_type	Indicates whether the loan is an individual application or a joint application with two co-borrowers
avg_cur_bal	Average current balance of all accounts
bc_open_to_buy	Total open to buy on revolving bankcards.
bc_util	Ratio of total current balance to high credit/credit limit for all bankcard accounts.
chargeoff_within_12_mths	Number of charge-offs within 12 months
collection_recovery_fee	post charge off collection fee
collections_12_mths_ex_med	Number of collections in 12 months excluding medical collections

delinq_2yrs	The number of 30+ days past-due incidences of delinquency in the borrower's credit file for the past 2 years
delinq_amnt	The past-due amount owed for the accounts on which the borrower is now delinquent.
Desc	Loan description provided by the borrower
Dti	A ratio calculated using the borrower’s total monthly debt payments on the total debt obligations, excluding mortgage and the requested LC loan, divided by the borrower’s self-reported monthly income.
dti_joint	A ratio calculated using the co-borrowers' total monthly payments on the total debt obligations, excluding mortgages and the requested LC loan, divided by the co-borrowers' combined self-reported monthly income
earliest_cr_line	The month the borrower's earliest reported credit line was opened
emp_length	Employment length in years. Possible values are between 0 and 10 where 0 means less than one year and 10 means ten or more years.
emp_title	The job title supplied by the Borrower when applying for the loan.*
fico_range_high	The upper boundary range the borrower’s FICO at loan origination belongs to.
fico_range_low	The lower boundary range the borrower’s FICO at loan origination belongs to.
funded_amnt	The total amount committed to that loan at that point in time.
funded_amnt_inv	The total amount committed by investors for that loan at that point in time.
Grade	LC assigned loan grade
home_ownership	The home ownership status provided by the borrower during registration. Our values are: RENT, OWN, MORTGAGE, OTHER.
Id	A unique LC assigned ID for the loan listing.
il_util	Ratio of total current balance to high credit/credit limit on all install acct
initial_list_status	The initial listing status of the loan. Possible values are – W, F
inq_fi	Number of personal finance inquiries
inq_last_12m	Number of credit inquiries in past 12 months
inq_last_6mths	The number of inquiries in past 6 months (excluding auto and mortgage inquiries)
Installment	The monthly payment owed by the borrower if the loan originates.
int_rate	Interest Rate on the loan
issue_d	The month which the loan was funded
last_credit_pull_d	The most recent month LC pulled credit for this loan
last_fico_range_high	The upper boundary range the borrower’s last FICO pulled belongs to.
last_fico_range_low	The lower boundary range the borrower’s last FICO pulled belongs to.
last_pymnt_amnt	Last total payment amount received
last_pymnt_d	Last month payment was received
loan_amnt	The listed amount of the loan applied for by the borrower. If at some point in time, the credit department reduces the loan amount, then it will be reflected in this value.
loan_status	Current status of the loan
max_bal_bc	Maximum current balance owed on all revolving accounts
member_id	A unique LC assigned Id for the borrower member.
mo_sin_old_il_acct	Months since oldest bank installment account opened
mo_sin_old_rev_tl_op	Months since oldest revolving account opened
mo_sin_rcnt_rev_tl_op	Months since most recent revolving account opened
mo_sin_rcnt_tl	Months since most recent account opened
mort_acc	Number of mortgage accounts.

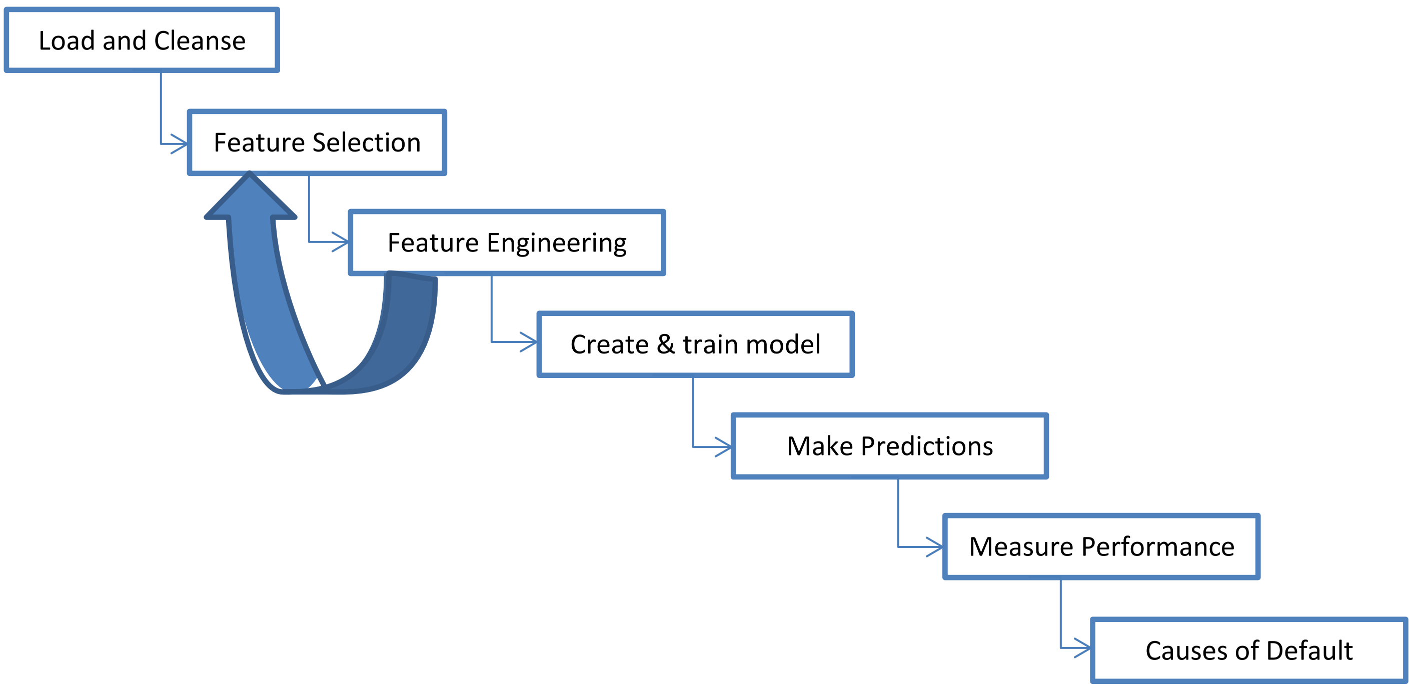
mths_since_last_delinq	The number of months since the borrower's last delinquency.
mths_since_last_major_derog	Months since most recent 90-day or worse rating
mths_since_last_record	The number of months since the last public record.
mths_since_rcnt_il	Months since most recent installment accounts opened
mths_since_recent_bc	Months since most recent bankcard account opened.
mths_since_recent_bc_dlq	Months since most recent bankcard delinquency
mths_since_recent_inq	Months since most recent inquiry.
mths_since_recent_revol_delinq	Months since most recent revolving delinquency.
next_pymnt_d	Next scheduled payment date
num_accts_ever_120_pd	Number of accounts ever 120 or more days past due
num_actv_bc_tl	Number of currently active bankcard accounts
num_actv_rev_tl	Number of currently active revolving trades
num_bc_sats	Number of satisfactory bankcard accounts
num_bc_tl	Number of bankcard accounts
num_il_tl	Number of installment accounts
num_op_rev_tl	Number of open revolving accounts
num_rev_accts	Number of revolving accounts
num_rev_tl_bal_gt_0	Number of revolving trades with balance >0
num_sats	Number of satisfactory accounts
num_tl_120dpd_2m	Number of accounts currently 120 days past due (updated in past 2 months)
num_tl_30dpd	Number of accounts currently 30 days past due (updated in past 2 months)
num_tl_90g_dpd_24m	Number of accounts 90 or more days past due in last 24 months
num_tl_op_past_12m	Number of accounts opened in past 12 months
open_acc	The number of open credit lines in the borrower's credit file.
open_acc_6m	Number of open trades in last 6 months
open_il_12m	Number of installment accounts opened in past 12 months
open_il_24m	Number of installment accounts opened in past 24 months
open_il_6m	Number of currently active installment trades
open_rv_12m	Number of revolving trades opened in past 12 months
open_rv_24m	Number of revolving trades opened in past 24 months
out_prncp	Remaining outstanding principal for total amount funded
out_prncp_inv	Remaining outstanding principal for portion of total amount funded by investors
pct_tl_nvr_dlq	Percent of trades never delinquent
percent_bc_gt_75	Percentage of all bankcard accounts > 75% of limit.
policy_code	publicly available policy_code=1 new products not publicly available policy_code=2
pub_rec	Number of derogatory public records
pub_rec_bankruptcies	Number of public record bankruptcies
Purpose	A category provided by the borrower for the loan request.
pymnt_plan	Indicates if a payment plan has been put in place for the loan
Recoveries	post charge off gross recovery
revol_bal	Total credit revolving balance
revol_util	Revolving line utilization rate, or the amount of credit the borrower is using relative to all available revolving credit.
sub_grade	LC assigned loan subgrade
tax_liens	Number of tax liens

Term	The number of payments on the loan. Values are in months and can be either 36 or 60.
Title	The loan title provided by the borrower
tot_coll_amt	Total collection amounts ever owed
tot_cur_bal	Total current balance of all accounts
tot_hi_cred_lim	Total high credit/credit limit
total_acc	The total number of credit lines currently in the borrower's credit file
total_bal_ex_mort	Total credit balance excluding mortgagemp_lengthe
total_bal_il	Total current balance of all installment accounts
total_bc_limit	Total bankcard high credit/credit limit
total_cu_tl	Number of finance trades
total_il_high_credit_limit	Total installment high credit/credit limit
total_pymnt	Payments received to date for total amount funded
total_pymnt_inv	Payments received to date for portion of total amount funded by investors
total_rec_int	Interest received to date
total_rec_late_fee	Late fees received to date
total_rec_prncp	Principal received to date
total_rev_hi_lim	Total revolving high credit/credit limit
url	URL for the LC page with listing data.
verification_status	Indicates if income was verified by LC, not verified, or if the income source was verified
verified_status_joint	Indicates if the co-borrowers' joint income was verified by LC, not verified, or if the income source was verified
zip_code	The first 3 numbers of the zip code provided by the borrower in the loan application.

LEGEND

	Removed due to high correlation
	Insufficient values in data set
	Removed due to irrelevance
	Removed because data acquired after default
	Significant relevance
	Not found in dataset
	Label source
	Tentative relevance
	Significant and part of optimal subset

Approach



Step 1: Load and Cleanse

Load dataset and perform initial cleansing. Cleansing will entail removing records with little or no information other than NULL values and potentially populating empty cells with imputed values.

Step 2: Feature Selection

Determine which features are significant and which will be used in the predictive models. There are three sub-steps:

1. Manually remove all features which mainly consist of NULL values.
2. Using correlation analysis, remove highly correlated features from the dataset.
3. Using the Boruta algorithm and Recursive Feature Elimination, determine which features are significant. Note: The significant features will be used in the predictive models.

Step 3: Feature Engineering

Perform further analysis of the data and determine whether features can be modified, combined or split to provide even more useful information. Please note that this may involve going back to feature selection before moving to the next step.

Step 4: Create and Train Model

Using the pruned dataset consisting of features determined from steps 2 and 3, create a test and training dataset. Create predictive models on the training dataset for the following algorithms: Random Forests, Bagged (MLP) Neural Networks, and Bagged Decision Trees. Note: Each model is created using the same training dataset.

Step 5: Make Predictions

Use the models on the test dataset and record the results.

Step 6: Measure Performance

This is a sub-step of step 5. For each result set in step 5, measure the Percentage Correctly Classified (PCC), the area under a ROC curve (AUC), H-measure, and Brier Score (BS) and compare the performance of each of the classification algorithms.

Step 7: Causes of Default

Using the knowledge gained from all of the previous steps, determine the causes of loan default.

Code

https://github.com/ribeiros/lending_club_default_prediction