Machine Learning Nanodegree

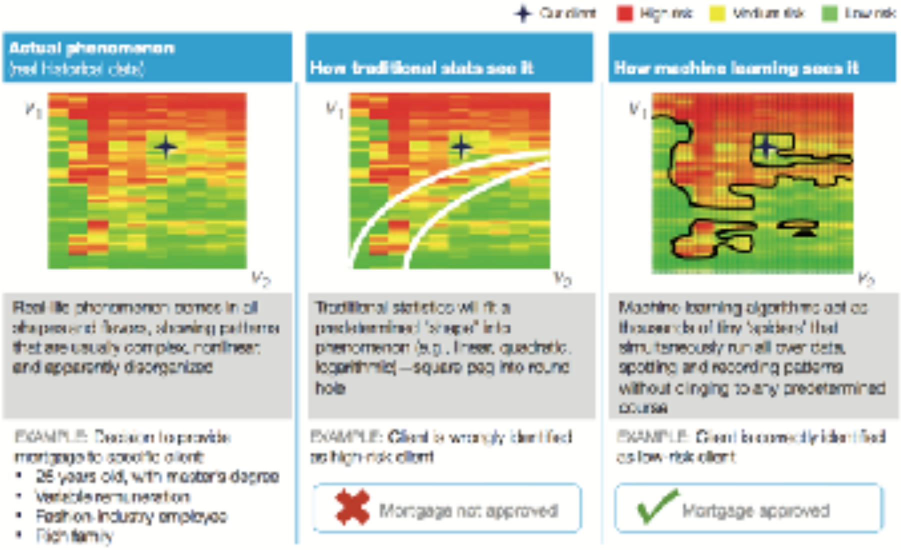
Capstone Project Proposal: Lending Club Dataset

Kenneth Gjaeringen
10th July 2017

Domain Background

In the consumer financing area being able to identify high risk customers during the origination process would benefit the originator by reducing foreclosure costs and loss severity. A consumer lending business face a large number of decisions when lending money to a consumer. Making sense of all these consumer variables makes it necessary to rely on models and algorithms rather than human discretion. Traditionally a lender base their lending business on standard underwriting policies (see middle chart below showing the traditional risk curve). These policies are quite rigid in that they classify the risk based on various scales, e.g. debt to income no more than 50%, and borrower characteristics such as being employed. In-house risk or rating agency models are then applied to access creditworthiness on consumers, but on a weighted average basis for the entire loan portfolio. None of these standard industry models are able to identify specific consumers who are at risk of default. This Capstone project aim to apply machine learning algorithms to better define consumers who might default. Being able to predict at the underwriting process whether a borrower is likely to be at high risk of default would give the lender further options whether to carry on with the loan application or increase the interest rate to reflect the additional risk. Furthermore, these borrowers could be tracked a lot more closely during their term to ensure a successful redemption of the loan.

In a paper published by McKinsey & Company[1] they compared the traditional approach, see middle chart below, to what a machine learning algorithm could do, right hand chart below. Machine learning offer the opportunity to give a lender a deeper insight into their data. The comparison below demonstrates the opportunity applying a machine learning model to borrower data. Using the available data (present and past) a machine learning algorithm should be able to look at a borrower and determine whether they are a creditworthy borrower or not. Whilst in a traditional approach an applicant might not be considered an eligible applicant due to being wrongly identified as high risk, but using a machine learning approach pockets of creditworthy borrowers (outside the lender credit policy middle chart) can be identified, as can be seen from the third chart below.



[1] McKinsey & Company The Future of Bank Risk Management working paper (http://www.mckinsey.com/business-functions/risk/our-insights/the-future-of-bank-risk-management)

Another area where machine learning can help is the categorisation of borrowers beyond the normal industry classification (stratification). Through feature engineering the machine learning model can find unique borrower clusters that can aid the future product development or targeted for cross selling purposes.

Obviously the above approach assumes there is a historic database to feed the machine learning models. However, for newer entrants into the financial market lack of data shouldn't be an issue as similar datasets can be acquired to test the machine learning models and verify whether the business model is viable or not.

Loan default classification would also be useful for accounting purposes, such as the accounting rule IFRS9, being able to predict based on past loan performance what is the likely outcome and loan provision that would be required for the accounts. However, in order to make this work one would also need a full transaction history along with the static borrower data. As detailed transaction history isn't available this wouldn't be possible to carry out.

For a business to be successful, credit decisions must be based on a model that is able to capture the customers risk profile in a timely and accurate manner.

My background is structured finance and data analytics within in the mortgage industry and this dataset is an opportunity to apply machine learning models to improve risk analytics, customer segmentation and loan pricing. Although this project will be focused on classifying a borrower during the application stage whether or not the applicant has a high likelihood of going into default. As the dataset already has the outcome for most of the Lending Club loans I will apply supervised learning models.

Problem Statement

This Capstone project aim to predicting whether a borrower is likely to default or not using loan data available at origination (although restricted to data available in the Kaggle dataset). From the various inputs a borrower can either be classified as in default or not and is a binary problem or classification problem. In normal circumstances borrowers who tend to default usually have similar characteristics and hence the reason for going down the classification route. However, there will be borrowers with good credit features that will experience life changing events such as illness, divorce, death, unemployment, etc. which will have an impact on the loan performance. However, these factors are outside any model capability due to their random occurrence of these events (considered random due to lack of detailed personal data). I would expect cases like these would normally be outliers.

Using the approach defined by Mitchell (1997) the machine learning problem can be defined as follows:

Task: Classify applicants likely to default based on the training experience;

Performance measure: percentage classification accuracy using a number of measures detailed in the Evaluation Metrics section;

Training experience: Lending Club dataset with final loan status with which the algorithm will train itself;

Feature engineering: reducing the attributes to key variables that help the algorithm prediction will improve model performance and make it less complex[2];

Target function: using the various variables a target function will be generated based on the aim of optimising the loss function, i.e. adjusting the various input variable factors to minimise the incorrect classification.

Target function representation: Supervised learning classification algorithms, using a handful of algorithms to baseline the output to further tune the best performing algorithms;

[2] Guyon and Elisseeff "An Introduction to Variable and Feature Selection" http://www.jmlr.org/papers/volume3/guyon03a/guyon03a.pdf

Datasets and Inputs

The Lending Club dataset was obtained from kaggle.com and at the time of download contains 74 fields and 887,379 loans. The dataset contain completed loan data for all loans issued through the 2007-2015, including the current loan status (Current, Late, Fully Paid, etc.) and latest payment information. The file containing loan data through the "present" contains complete loan data for all loans issued through the previous completed calendar quarter. Additional features include number of finance inquiries, address including zip codes, and state, and collections among others.
In this dataset made available on kaggle.com consisting of loan level borrower details, summary transaction history and account-balance data for individual Lending Club consumers. The raw data was available as a CSV flat-file.

There is a data dictionary on the Kaggle website but due to version control issues this document does not reflect the current dataset and is only partially relevant. Only data features available at the time of origination of the loan will be considered during this project plus the final loan status. Any other features related to loan performance will be excluded and deemed irrelevant as it won't represent the data at the point of origination. The field called "loan_status"

To mention a few features from the dataset that would be available at the time of origination:

- Loan grades;
- Term;
- Interest rate;
- Debt to income (DTI);
- Employment Length;
- Home ownership;
- Income verification;

After importing the entire dataset features not required will be removed from the dataset, any remaining features string categories (such as Home ownership which contains "RENT", "MORTGAGE" and "OWN") will have its own column for each category with a boolean value, i.e. where there field value says "RENT" the "Home_Ownership_RENT" feature will have a "1" otherwise "0". Any value fields will be normalised. Features like Zip code which can be converted into a Region will be kept as it is as Region by itself shouldn't have an extensive impact whether the borrower will default or not. Normally the Region is of concern if you have a high concentration / exposure. Furthermore, borrower defaults in certain towns and regions may be related to a company closing five years from now, it's unlikely that a specific forecast or even a specific distribution of probabilities can be constructed that factor or for any individual factor (Chapter 12 in Davidson & Levin 2014).

Once the dataset has gone through the "ETL process" feature engineering can take place to ensure noise is reduced, improve predictability and reduce training time with less data. Techniques such as Principal Component Analysis and oversampling methods will be applied to the dataset to further understand the loan population and clustering, reduce any imbalance and reduce the dimensionality of the dataset.

Solution Statement

To tackle the above problem statement described above we will apply supervised learning algorithms to classify borrowers who are likely to default at the application stage. The challenge of supervised learning is to find a function that generalises beyond the training set, so that the resulting function also accurately maps out-of-sample inputs to out-of-sample outcomes.

However, before any machine learning algorithm can be applied a certain amount of data preprocessing needs to be applied to ensure better predictability. The Lending Club dataset contains more data points than is required for this project. In order to solve the default question at the origination stage and minimise look-ahead bias, I will exclude any loan performance data should be excluded as this won't be available during the loan origination process. For the remaining data features the next stage would be to normalise value fields and decompose categorical attributes, i.e. if there are 2 categories making up a Loan Type feature (containing repayment or interest only) an extra field will be generated for each category with "1" where the feature is true and "0" when it's not true.

The population of defaulted borrowers is a smaller segment of the total pool and therefore creates an imbalance of the classes in the dataset. Oversampling techniques such as Adaptive Synthetic (ADASYN) and Synthetic Minority Over Sampling Technique (SMOTE) will be applied to rebalance the dataset. Oversampling techniques uses key data points and creates synthetic instances of data points to rebase the imbalanced data class, i.e. defaulted borrowers. Oversampling randomly replicates minority instances to increase their population. Undersampling randomly downsamples the majority class[3]. Although oversampling techniques generates extra data undersampling can make the independent variable look like they have a higher variance than they do[3].

Benchmark Model

After the feature engineering process has been satisfactorily completed a set of baseline models run against the dataset to establish which model is likely to perform best. As it's difficult to determine which supervised learning model will perform best a set of models will be used:

- Ensemble methods
- Decision tree methods
- Nearest Neighbour methods
- Stochastic Gradient Descent method
- Support Vector Machine method
- Gaussian processes
- Naive Bayes
- Neural Network model

No tuning will take place during this model selection and will form part of the benchmarking of key models.

From this we will reduce the models to a handful of models to further tune and improve with the aim to end up with one machine learning algorithm that can be applied to the test set to predict borrowers who are likely to default. This output can be used in two ways: which borrowers is likely to default and what is the default rate of a portfolio/vintage/cohort.

Evaluation Metrics

In order to establish the classification performance of a machine learning model key performance metrics needs to be extracted for each benchmark and solution model. For this project the following performance metrics will be applied for all models:

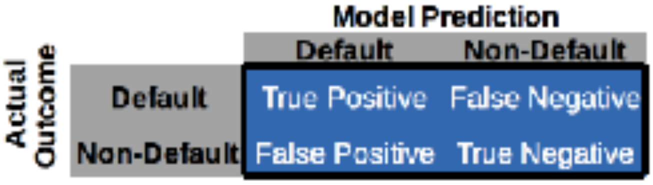
- Confusion Matrix
- Precision
- Recall
- Kappa statistic
- Mean Absolute Error
- Root Mean Squared Error
- F1-Score
- ROC Curves
- AUC

CONFUSION MATRIX

Confusion matrix is a method of describing the performance of the classification algorithms, typically for supervised learning models. The matrix also gives the end user an idea as to how well the model(s) is performing and what type of errors it is making.

The matrix consists of a 2 x 2 table where the table records the following outputs True Positive (TN) correctly predicted defaulted customers, True Negative (TN) correctly predicted non-default customers, False Positive (FP) incorrectly predicted default customers and False Negative (FN) incorrectly predicted non-default customers. The confusion matrix can be summarised as follows:

[3] Learning from Imbalanced Classes, Tom Fawcett. https://svds.com/learning-imbalanced-classes/



		Model Prediction	
		Default	Non-Default
Actual Outcome	Default	True Positive	False Negative
	Non-Default	False Positive	True Negative

PRECISION

Precision refers to the model's accuracy in instances that the model classified an account as a default account and expressed as follows:

$$Precision = TP/(TP + FP)$$

RECALL

Recall refers to the number of defaulted accounts identified by the model divided by the actual number of defaulted accounts (a measure of a classifier completeness) and expressed as:

$$Recall = TP/(TP + FN)$$

KAPPA STATISTIC

Cohen's Kappa is an evaluation statistic that takes into account how much agreement would be expected by chance. The Kappa statistic can be expressed as follows:

$$k = (Po - Pe)/(1 - Pe)$$

Where Po is the relative observed agreement among raters (accuracy), and Pe is the hypothetical probability of chance agreement, using the observed data to calculate the probabilities of each observer randomly seeing each category[4].

MEAN ABSOLUTE ERROR (MAE)

A metric to consider the error in the predicted values as compared to the expected values.

$$MAE = SUM(ABS(Prediction - Actaul))/TotalPredictions$$

ROOT MEAN SQUARED ERROR (RMSE)

Another way to calculate the error in a set of predictions is to use the Root Mean Squared Error. By applying the square root to the MSE calculation forces the values to be positive, and the square root of the mean squared error returns the error metric back to the original units for comparison.

$$RMSE = SQRT(SUM(Prediction - Actual) * *2)/TotalPredictions)$$

F1 SCORE

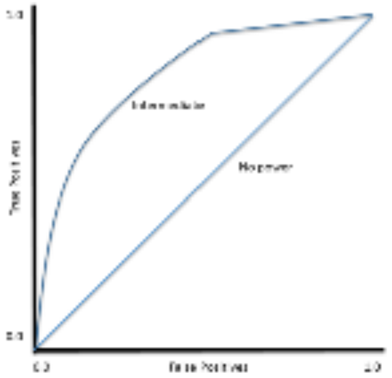
F1 Score is the weighted average of precision and recall. Expressed as:

$$F1 - Score = 2((Precision * Recall))/(Precision + Recall))$$

[4] Cohen's Kappa; https://en.wikipedia.org/wiki/Cohen's_kappa

RECEIVER OPERATING CHARACTERISTIC (ROC) CURVE

ROC Curves is a standard technique for summarising classifier performance over a range of tradeoffs between true positive and false positive error rates[5].



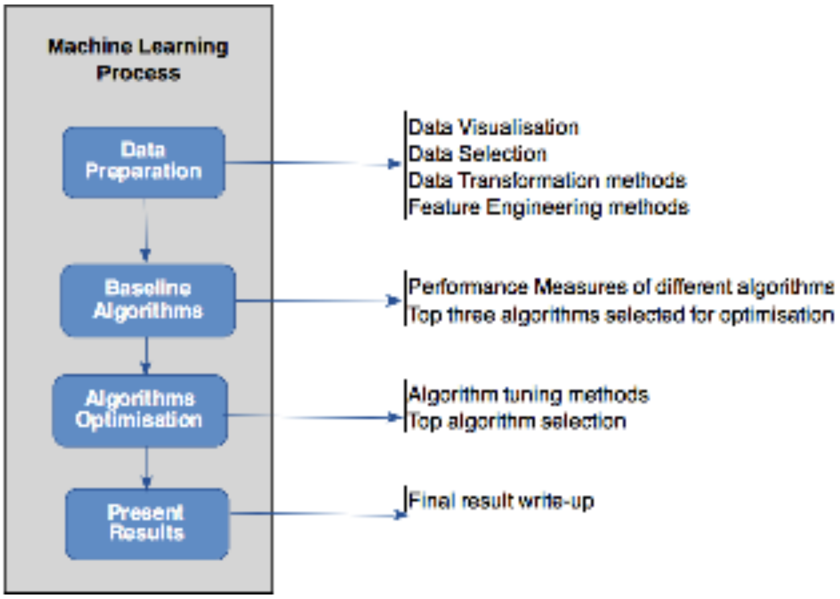
For a perfect classifier the ROC curve will go straight up the True Positive axis and then along the False Positive axis. A classifier with no power will sit on the diagonal. Furthermore, by adding several machine learning algorithms to this ROC curve will show the optimal algorithm to use going forward.

THE AREA UNDER THE ROC CURVE (AUC)

The AUC It is equal to the probability that a random positive example will be ranked above a random negative example.

Project Design

In order to ensure the best result from a dataset answering a specific problem statement using machine learning (ML) algorithms a rigid workflow needs to be followed otherwise the machine learning algorithms won't be fully optimised and might even cause algorithms to be discarded due to poor workflow process. Below is a summarised view of the proposed workflow which can be applied to any ML problem. The workflow below assumes the problem statement that will be answered by the ML algorithm has been defined in advance.



[5] Data Mining For Imbalanced Datasets: An Overview, Nitesh V. Chawla; http://www3.nd.edu/~dial/publications/chawla2005data.pdf

1. Data Preparation

Most of the project time will be spent in this stage to better understand the data, exclude any data that is not relevant and make the dataset ready for a machine learning algorithm.

The following data preparation steps will be applied:

- Data preprocessing, i.e. clean data, data formatting, data rebalancing, outlier analysis and removal, dimensionality reduction;
- Data transformation, i.e. data normalisation and discretise data to handle nominal values;
- Data summarisation, i.e. create various plots to unravel any obvious relationships in the data.

2. Baseline Algorithms

At the outset of a ML project it might not be obvious which ML algorithm that might perform best. As described in the Benchmark Model section I will apply a number of supervised learning algorithms without any tuning to analyse which model performs the best.

The ML models will be evaluated against a standard set of metrics to further shortlist ML algorithms to optimise (see Evaluation Metrics section for further details).

3. Algorithms Optimisation

From the baseline algorithm process a handful (top 3 models) ML algorithms will be taken forward to be optimised by exploring each algorithm variables. As in the baseline algorithm stage the same evaluation metrics will be applied. The aim of this workflow process is to arrive at a model that generalises enough to be applied in a production environment and used as part of the loan underwriting process.

4. Present Results

A detailed report outlining the workflow process and document steps taken to achieve the output after algorithm optimisation.

Furthermore, the final document will detail operationalisation of the predictive model created, elaborating the process of data gathering and data quality, and how to use the predictive model.

References

A. Davidson and A. Levin, Mortgage Valuation Models Embedded Options, Risk and Uncertainty. Oxford University Press, 2014.

T. M. Mitchell, Machine Learning. McGraw Hill, International Edition 1997.