Modelling Credit Card Defaults: A Comparison between Logistic Regression and Random Forests

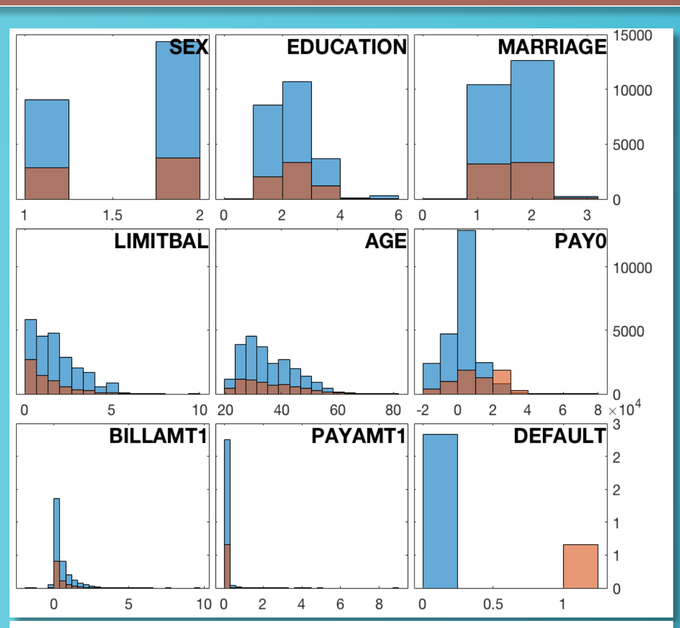
Yann Adjanor and Shawn Ban

Description and motivation

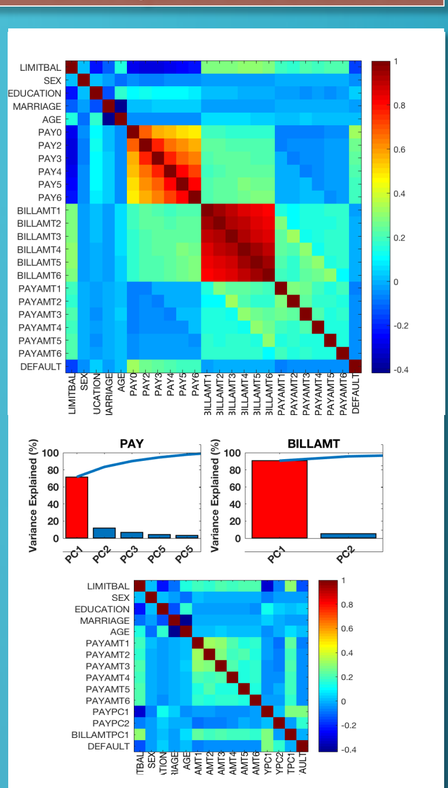
- Critical evaluation of two supervised machine learning classification algorithms: **Logistic Regression** and **Random Forests** (Bagged Trees ensembles).
- Domain of application: predict defaults of Taiwanese credit card customers in the next month.
- A previous study by Yeh & Lien (2009)1 aims to predict probability of default, but we treat this as a binary classification problem.

Exploratory data analysis

- Dataset: credit card defaults from UCI[2]. Contains 30,000 observations of default payments, demographics, credit data, history of payment and bill statements of credit card customers in Taiwan from April to September 2005.
- Dataset has 23 predictor variables (9 categorical, 12 continuous) and one response variable (default, 1=yes, 0 =no).
- Of particular interest are categorical variables tracking the repayment status over the previous 6 months.
- Account-specific data (Limit, Bill Amount, Payment Amount and Pay Records) are quite **right-skewed**.
- We **normalize** all the numerical data using a **z-score function** to preserve the observed variance of these skewed distributions.
- The correlation matrix shows that will most attributes are loosely correlated, 2 groups of data are very inter-correlated.
- We apply **principal component analysis** on those groups to reduce the overall amount of features to 14.
- We note the **class imbalance** in the dataset as defaults represent only 22.1% of the total; therefore choice of performance measure must be robust.



Main Features Observed Distributions



Features Transformation

Models Characteristics – Pros and Cons

Logistic Regression (LR)

- Special form of **regression analysis for non-continuous dependent variables**.
- In the case of binomial logistic regression, response variable assumed to be binary and fitted via link function (logit) through standard linear regression analysis; logistic function transforms binary variable into continuous variable.
- As in general linear models (GLM), different basis functions can be fitted to the model.
- Assumes binomially distributed errors with zero mean (normally distributed for large samples), and low levels of collinearity between predictor variables (Menard, 2010)[3].

Pros:

- **Simple and intuitive**, easy to explain to wider audience.
- **Deterministic** with a closed form solution (for OLS).
- **Fast**: *O(n)* for **evaluation** (*n*=number of features).

Cons:

- **Sensitive to noise and extremes** (outliers), particularly for least squares regression.
- Cannot capture non-linearities between predictors unless using more complex basis functions.
- Slow training for large number of features (around $O(n^3)$ complexity).
- Maximum likelihood estimation cannot yield a closed-formed solution as in GLM since errors not normally distributed; iterative process must be used and convergence may fail if model assumptions above not met.

Random Forest (RF)

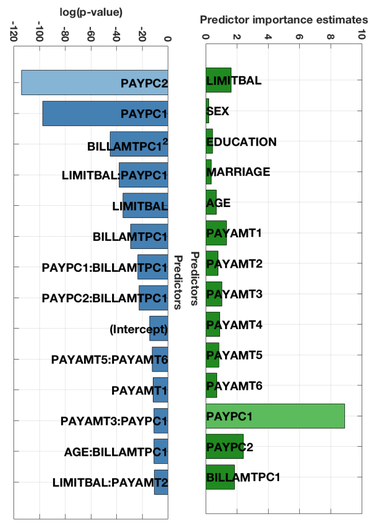
- Builds an **ensemble of decision trees**.
- We choose the **bagged trees** (bootstrap aggregating) method, which builds the forest by **sampling subsets with replacement** from the training data. Described in detail by Breiman (2001)[4].
- Each tree is built with a random subset of features to reduce correlation between trees.
- By default, trees are equally weighted.
- Generates unbiased estimate of generalization error as forest building progresses.

Pros:

- **Bootstrapping reduces variance** and makes Random Forest less prone to overfitting.
- Provides a ranking of features by importance, which can be informative.
- Easily handles categorical features.
- **Relatively robust to outliers** as they are isolated in small regions of feature space in each decision tree.

Cons:

- Not easy to interpret visually, as inspecting individual trees does not yield much information.
- Can be computationally expensive depending on tree and forest size, as requires storage of each tree in memory.



Features Importances

Hypothesis –Methodology and Choice of Evaluation Criteria

Hypothesis

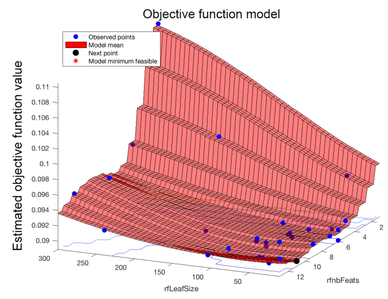
- We expect worthwhile results from both methods, but Random Forest to outperform Logistic Regression.
- Previous studies suggest that Logistic Regression performs better on smaller datasets, but Random Forest performs better on larger datasets[5 6]

Methodology

- Training set: 24,000 observations; test set: 6,000 observations, keeping roughly the same class proportions in each set via Matlab's `cvpartition`.
- For Logistic Regression, we look at **linear and quadratic basis functions** to capture non-linear interactions between predictor variables.
  To reduce dimensionality of the pure quadratic model (120 combined features), we use a **stepwise model construction** which incrementally adds quadratic terms to the base linear model if these terms improve the model (p-value below pre-defined threshold).
- For Random Forest, we train a regression tree with **maximum a posteriori** rule, i.e. assign to default if P(default) > 0.5.
  Random Forest hyper-parameters include number of trees, leaf size, tree depth; check for a stable out-of-bag error to select number of trees, and run a **Bayesian hyper-parameter optimization** to select leaf size, tree depth.

Choice of evaluation criteria

- Due to class imbalance, a rule always predicting 'no default' would be 78% accurate. Accuracy alone is therefore a poor model evaluation criteria.
- We report accuracy, precision, recall, F1 score and area under curve (AUC) on the test set for each model, but **prefer AUC measures** for model evaluation instead of traditional accuracy measures for classification, as noted in Yeh and Lien (2009)[1].



RF Bayesian Optimiser

RESULTS	Logstic Regression	Random Forest
Accuracy	0.803	0.823
Recall	0.254	0.396
Precision	0.636	0.668
F1 Score	0.363	0.497
AUC	75.20%	77.60%

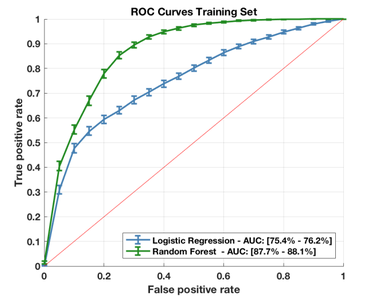
Performance Scores

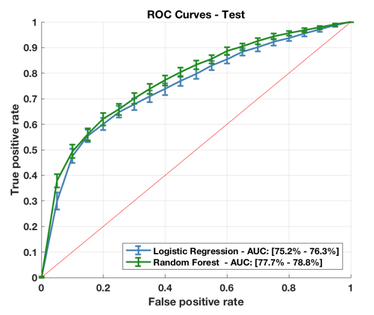
Choice of Parameters

Best model found comprises **14 linear terms** and an **additional 29 quadratic terms**.
Learning curves show that the model is not over-fitting and shows signs of under-fitting.

- Result **could be a local optimum** as solver reached maximum number of iterations on 9 features: system is under-determined and cannot be fully captured by a simple regression model, even with quadratic terms.

- The out-of-bag error converges to 13.6% between 90-100 trees; we set 100 as our number of trees.
- We run a Bayesian optimizer (`bayesopt`) which searches for hyper-parameters to minimize the out-of-bag error.
- Treats objective function as a random prior, evaluates it and updates posterior distribution to determine next query point.
- Bayesian optimization yields an optimal leaf size of 46 and optimal tree depth as 5





ROC Curves

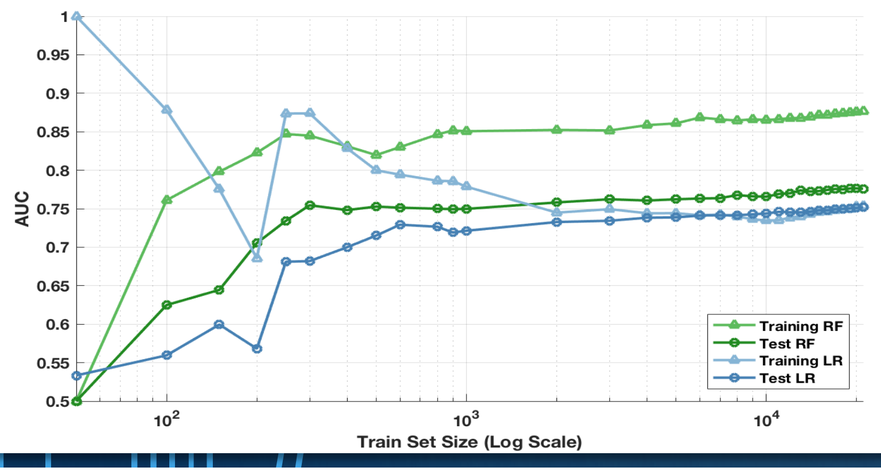
Experimental Results – Critical Analysis and Evaluation

Experimental results

- Overall, **Random Forest outperforms Logistic Regression slightly** on various performance measures (accuracy, precision, AUC), but performs much better on recall and hence F1 score

Critical analysis and evaluation

- Logistic Regression model is remarkably **robust** as noted by Perlich, Provost, and Simonoff (2003)[5] as different variants do not significantly improve or worsen performance, and training and test performance converge quite quickly, but at the expense of low variance and higher bias.
- Higher bias of the Logistic Regression model hints that the **underlying model is too simple**: adding quadratic terms is not enough and we could look at other basis functions such as a **Gaussian kernel function**, which allows a non-linear mapping to an infinite-feature space. Risk will then be of overfitting the data but this could be controlled by cross-validation.
- For Random Forest, AUC drops from 87% on training set to 78% on test set, error rises from 13.6% out-of-bag to 17.7% on the test set, indicating **low bias but high variance**, so the model seems to be overfitting the data.
- One solution for Random Forest is to reduce tree depth and the number of features, as parameters are optimized for out-of-bag error on the training set; this would increase bias but may lower variance.
- **Feature importance**: PAYPC2 is most significant feature for Logistic Regression, while PAYPC1 is most important for Random Forest.
- **Learning curves are instructive** as they seem to confirm that the Random Forest model may benefit from more data, but not the Logistic Regression model.
- **Matches our hypothesis**, but we had expected difference to be more pronounced.
- Logistic Regression takes 0.5 seconds to run, while Random Forest takes 11 seconds, but stepwise construction and parameter optimization are computationally expensive.



Model Learning Curves

Lessons Learned and Future Work

- Another option for Logistic Regression is to look at **non-parametric methods**: Frölich (2009)[7] shows that local likelihood logit regression can outperform its parametric version. Instead of a global parametrisation, local logit allows the parametrisation to change according to the instance's position in the features-space.
- For Random Forest, Chen and Breiman (2004)[8] suggest a modification to handle class imbalance via a **Weighted Random Forest** (WRF) algorithm.
  WRF places a higher weight on the minority class, thus imposing a penalty on trees that misclassify the minority; the class weights add another possible model parameter to be varied.

References

[1] Yeh, I. & Lien, C. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert Systems with Applications*, vol. 36, no. 2, pp. 2473-2480.
[2] UCI Machine Learning repository: https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
[3] Menard, S. (2010). Logistic Regression: From Introductory to Advanced Concepts and Applications. *SAGE Publications*.
[4] Breiman L. (2001). Random Forests. *Machine Learning*, vol. 45, no. 1, pp. 5-32.
[5] Caruana, R. & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. *ICML '06 Proceedings of the 23rd international conference on machine learning*, pp. 161-168.
[6] Perlich, C., Provost F., & Simonoff, J. (2003). Tree Induction vs. Logistic Regression: A Learning-Curve Analysis. *Journal of Machine Learning Research*, vol. 4, pp. 211-255.
[7] Frölich, M. (2006) 'Non-parametric regression for binary dependent variables', *Econometrics Journal*. Blackwell Publishing Ltd, 9(3), pp. 511–540.
[8] Chen, C. & Breiman, L. (2004). Using Random Forest to Learn Imbalanced Data. University of California, Berkeley.