A Relational Model of Data for Large Shared Data Banks

Rahul Shamdasani
Graduate Student : M.S. C.S.
Indiana University, Bloomington
rshamdas@iu.edu

Lars Thørväld
The Thørväld Group
Hekla, Iceland
larst@affiliation.org

Valerie Béranger
Inria Paris-Rocquencourt
Rocquencourt, France
vb@rocquencourt.com

ABSTRACT

In today's world, obtaining loans from a financial institution has become a common phenomenon, in this paper we will discuss the profitability of an institution from a business perspective and try to maximize the same, using various machine learning techniques. We will predict loan status based on applications and applicant's credit history. Loans taken by a borrower can be fully paid, default or charged off depending on a lot of factors out of which some are taken into account.

1 MOTIVATION

This project will help solve the real world problem by working on the real world data. After this project we will not only have learned about the algorithms needed, but will also understand the importance of data preprocessing and various ways used to do the same. LendingClub is the first peer-to-peer lender to register its offerings as securities with the Securities and Exchange Commission (SEC), and to offer loan trading on a secondary market.

LendingClub is the world's largest peer-to-peer lending platform. Lending club works as a mediator between investors and borrowers. It works in favour of investors and helps them identify potential borrowers for lending their money. Potential borrowers are the ones who are likely to pay their loans on time. These potential borrowers are identified based on the credit history of the borrower and other application specific factors like interest rates, term of loan etc. Our project will help solve this issue by using machine learning techniques.

2 RELATED WORK

Nulla placerat feugiat augue, id blandit urna pretium nec. Nulla velit sem, tempor vel mauris ut, porta commodo quam. Donec lectus erat, sodales eu mauris eu, fringilla vestibulum nisl. Morbi viverra tellus id lorem faucibus cursus. Quisque et orci in est faucibus semper vel a turpis. Vivamus posuere sed ligula et.

3 PERFORMANCE MATRIX

To maximize the profitability of the organization, the best matrix for evaluation was precision of the model. Inside the precision we are more concerned about reducing the number of false positive results.

- **True Positive(TP)**: These are the candidates which are predicted as potential borrowers by our model and are actually potential borrowers.
- **False Positive(FP)**: These are the candidates which are predicted as potential borrowers by our model and have actually defaulted their loans.
- **True Negative(TN)**: These are the candidates which are predicted as defaulters by our model and are actually potential borrowers.
- **False Negative(TN)**: These are the candidates which are predicted as defaulters by our model and are actually defaulters.

Our main aim while training this model is to minimize the number of false positive(FP) results, since these are the customers which cause heavy losses to lending club.

4 CLASSIFICATION

Our classification goal is to predict which class the loan belongs to: either Default or Fully Paid. In the following sections, we will share and discuss our experiments using Neutral Networks and Random Forest for classification problem. For metrics to evaluate classification performance, we use confusion matrix whose columns represent predicted values and rows represent true values. We also measure precision, recall, f1-score (the harmonic mean of precision and recall) and weighted average as defined below:

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - score = \frac{2TP}{2TP + FP + FN}$$

$$Support = the number of true instances for each label$$

$$Weighted - avg metric = metric weighted by support$$

5 MODELS

5.1 XG Boost

We are using extreme gradient boosting as one of the models for our data, XG boost is a tree based ensemble machine learning algorithms. We selected this algorithm since we have a large data set and data is a combination of categorical and numerical values, also we have only 77k samples for charges off case which is very low compared to 3.1M samples for fully paid data so XG boost observation weightage takes care of that part, as it make multiple trees of N random samples with replacement, after each iteration the observations are weighted so there might be some samples which get selected multiple times.

Hyper Parameters

- **Booster**:GBTree
- **Step Size Shrinkage:**: 0.3
- **Alpha Alias**: Learning Rate

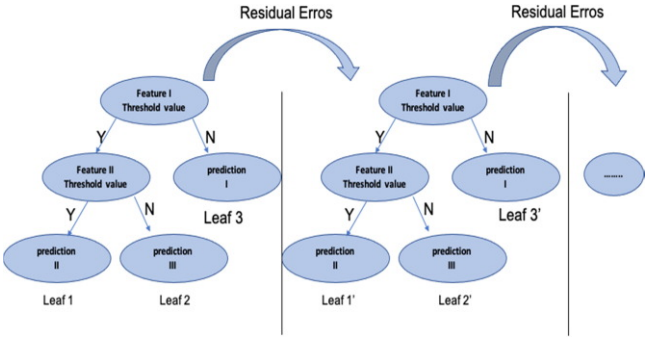

Figure 1: Operations in XG Boost

6 CITATIONS

Some examples of references. A paginated journal article [2], an enumerated journal article [7], a reference to an entire issue [6], a monograph (whole book) [15], a monograph/whole book in a series (see 2a in spec. document) [13], a divisible-book such as an anthology or compilation [10] followed by the same example, however we only output the series if the volume number is given [9] (so Editor00a's series should NOT be present since it has no vol. no.), a chapter in a divisible book [17], a chapter in a divisible book in a series [8], a multi-volume work as book [14], an article in a proceedings (of a conference, symposium, workshop for example) (paginated proceedings article) [3], a proceedings article with all possible elements [16], an example of an enumerated proceedings article [11], an informally published work [12], a doctoral dissertation [5], a master's thesis [4], an finally two online documents or world wide web resources [1, 18].

ACKNOWLEDGMENTS

This work was supported by the [...] Research Fund of [...] (Number [...]). Additional funding was provided by [...] and [...]. We also thank [...] for contributing [...].

REFERENCES

[1] Rafal Ablamowicz and Bertfried Fauser. 2007. *CLIFFORD: a Maple 11 Package for Clifford Algebra Computations, version 11.* Tennessee Technological University. Retrieved February 28, 2008 from http://math.tntech.edu/rafal/cliff11/index.html

[2] Patricia S. Abril and Robert Plant. 2007. The patent holder's dilemma: Buy, sell, or troll? *Commun. ACM* 50, 1 (Jan. 2007), 36–44. https://doi.org/10.1145/1188913.1188915

[3] Sten Andler. 1979. Predicate Path expressions. In *Proceedings of the 6th. ACM SIGACT-SIGPLAN symposium on Principles of Programming Languages (POPL '79).* ACM Press, New York, NY, 226–236. https://doi.org/10.1145/567752.567774

[4] David A. Anisi. 2003. *Optimal Motion Control of a Ground Vehicle.* Master's thesis. Royal Institute of Technology (KTH), Stockholm, Sweden.

[5] Kenneth L. Clarkson. 1985. *Algorithms for Closest-Point Problems (Computational Geometry).* Ph.D. Dissertation. Stanford University, Palo Alto, CA. UMI Order Number: AAT 8506171.

[6] Jacques Cohen (Ed.). 1996. Special issue: Digital Libraries. *Commun. ACM* 39, 11 (Nov. 1996).

[7] Sarah Cohen, Werner Nutt, and Yehoshua Sagic. 2007. Deciding equivalances among conjunctive aggregate queries. *J. ACM* 54, 2, Article 5 (April 2007), 50 pages. https://doi.org/10.1145/1219092.1219093

[8] Bruce P. Douglass, David Harel, and Mark B. Trakhtenbrot. 1998. Statecarts in use: structured analysis and object-orientation. In *Lectures on Embedded Systems*, Grzegorz Rozenberg and Frits W. Vaandrager (Eds.). Lecture Notes in Computer Science, Vol. 1494. Springer-Verlag, London, 368–394. https://doi.org/10.1007/3-540-65193-4_29

[9] Ian Editor (Ed.). 2008. *The title of book two* (2nd. ed.). University of Chicago Press, Chicago, Chapter 100, 201–213. https://doi.org/10.1007/3-540-09237-4

[10] Peter Eston. 1993. *The title of the work* (3 ed.). 5, Vol. 4. The name of the publisher, The address of the publisher, Chapter 8, 201–213. https://doi.org/10.1007/3-540-09237-4 An optional note.

[11] Matthew Van Gundy, Davide Balzarotti, and Giovanni Vigna. 2007. Catch me, if you can: Evading network signatures with web-based polymorphic worms. In *Proceedings of the first USENIX workshop on Offensive Technologies (WOOT '07).* USENIX Association, Berkley, CA, Article 7, 9 pages.

[12] David Harel. 1978. *LOGICS of Programs: AXIOMATICS and DESCRIPTIVE POWER.* MIT Research Lab Technical Report TR-200. Massachusetts Institute of Technology, Cambridge, MA.

[13] David Harel. 1979. *First-Order Dynamic Logic.* Lecture Notes in Computer Science, Vol. 68. Springer-Verlag, New York, NY. https://doi.org/10.1007/3-540-09237-4

[14] Donald E. Knuth. 1997. *The Art of Computer Programming, Vol. 1: Fundamental Algorithms (3rd. ed.).* Addison Wesley Longman Publishing Co., Inc., USA.

[15] David Kosiur. 2001. *Understanding Policy-Based Networking* (2. ed.). Wiley, USA.

[16] Stan W. Smith. 2010. An experiment in bibliographic mark-up: Parsing metadata for XML export. In *Proceedings of the 3rd. annual workshop on Librarians and Computers (LAC '10)*, Reginald N. Smythe and Alexander Noble (Eds.), Vol. 3. Paparazzi Press, Milan Italy, 422–431. https://doi.org/10.1038/nphys1170

[17] Asad Z. Spector. 1990. Achieving application requirements. In *Distributed Systems* (2nd. ed.), Sape Mullender (Ed.). ACM Press, New York, NY, 19–33. https://doi.org/10.1145/90417.90738

[18] Harry Thornburg. 2001. *Introduction to Bayesian Statistics.* Stanford University. Retrieved March 2, 2005 from http://ccrma.stanford.edu/~jos/bayes/bayes.html