भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Example dependent cost-sensitive classification using Deep Neural Net

Fraud Analytics: Assignment 4

Jatin Sharma
CS17BTECH11020

Anjani Kumar
CS17BTECH11002

Dhananjay Raut
CS17BTECH11014

Vijay Tadikamalla
CS17BTECH11040

(Teamsize of 4 with special permission from sir)

Objective

Several real-world classification problems are example-dependent cost-sensitive in nature, where the costs due to misclassification vary between examples – e.g. Credit Scoring. The objective of this assignment is to incorporate this requirement in real-world cost agnostic ML models and analyze the trade-off between classification accuracy and financial cost.

Introduction

Standard cost-insensitive binary classification algorithms, like Logistic Regression, Decision Trees, are often used in practice for real-world classification problems like Credit Scoring. The objective in credit Scoring is to classify when the target customer is likely to default a financial contract based on past financial experience. However, in the financial world, the cost associated with approving a potential defaulter varies differently and is quite different from falsely denying a good customer. Some authors have proposed methods that include the miss-classification cost [1]-[2], but assuming a constant misclassification cost is a major drawback.

We implement a framework where misclassification cost varies across examples, i.e. cost-sensitive example dependent classification. We then implement cost-sensitive logistic regression, by changing the objective function of the model to one that is cost-sensitive. We then evaluate the model with vanilla logistic regression and analyze the trade-off between classification accuracy and incurred financial cost. The results will show that the enhanced model will out-perform the base model and reduce the financial cost by a huge factor.

Neural Network

Neural networks or also known as Artificial Neural Networks (ANN) are networks that utilize complex mathematical models for information processing. They are based on the model of the functioning of neurons and synapses in the brain of human beings. Similar to the human brain, a neural network connects simple nodes, also known as neurons or units. And a collection of such nodes forms a network of nodes, hence the name "neural network."

A deep neural network (DNN) is an artificial neural network (ANN) with multiple layers between the input and output layers. The layers in the middle can be called hidden layers. compose computations performed by many layers. Denoting the output of hidden layers by h(l)(x), the computation for a network with L hidden layers is:

$$f(x) = f\,[\,a^{(L+1)}\,(h^{(L)}(a^{(L)} \ldots h^{(1)}(\,a^{(1)}(x)\,)\,)\,)\,]$$

$$\mathbf{a}^{(l)}(\mathbf{x}) = \mathbf{W}^{(l)}\mathbf{x} + \mathbf{b}^{(l)},$$
$$\mathbf{a}^{(l)}(\widehat{\mathbf{x}}) = \boldsymbol{\theta}^{(l)}\widehat{\mathbf{x}} \quad , l = 1$$
$$\mathbf{a}^{(l)}(\widehat{\mathbf{h}}^{(l-1)}) = \boldsymbol{\theta}^{(l)}\widehat{\mathbf{h}}^{(l-1)} \quad , l > 1$$

After passing through the network, the output value is calculated using the equation:

$$y_k = f\left[\sum_j w_i f\left(\sum_j (w_i x_i) + b_j\right) + b_k\right] \quad (1)$$

where;
$f(.)$ is the activation function
y_k is the kth output value
x_i is the ith input variable or feature
w is the weighting value used in the hidden and output layers and
b's are the network biases

The logistic (sigmoid) function, which is the most common and widely used activation function, was employed in this study. The equation for this function is given by:

$$f(x) = \frac{1}{\left[1 + \exp^{(-x)}\right]}$$

In our case, the network architecture and hyperparameters may or may not be optimized to the problem; instead, the network provides a basis for comparison for our target when, later, the training algorithm comes into action.

COST-SENSITIVE LOSS

We design a new cost function that is cost-sensitive and example dependent, by merging the various costs as follows

$$J^c(\theta) = \frac{1}{N}\sum_{i=1}^{N}\Big(y_i(h_\theta(\mathbf{x}_i)C_{TP_i} + (1 - h_\theta(\mathbf{x}_i))C_{FN_i})$$
$$+(1 - y_i)(h_\theta(\mathbf{x}_i)C_{FP_i} + (1 - h_\theta(\mathbf{x}_i))C_{TN_i})\Big).$$

Here we take into account example dependent costs associated, mainly C_{TP}, C_{TN}, C_{FP}, C_{FN} .

Experimentation

Dataset Summary

The dataset consists of customers' records such as illegal/legal transactions, whether the customer has filed within time limits, average tax per month, etc, and costs associated with each example.

Statistics

The dataset contains about 140000 samples, each one with 12 features and the class label.
The proportion of default or positive examples in 29.85%

Data partitioning

About ⅓ of the dataset was used for testing and the rest was used for training purposes, sampled based on class label distribution, to avoid undersampling in training examples.

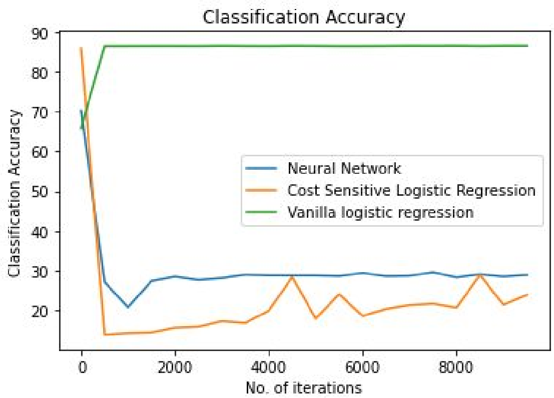
Results

We compare our results with the base logistic regression model which uses a cost-insensitive loss function and with a cost-sensitive example dependent neural network and analyze the trade-offs.

Cost independent logistic loss

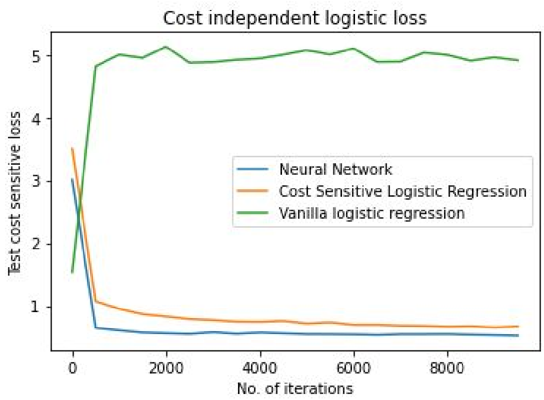
As we can see from the above plot, cost-sensitive models perform worse than the cost agnostic logistic regression. This is because of the fact that logistic regression is minimizing the negative log-likelihood loss, while other models are minimizing the cost dependent loss function.

Classification Accuracy



We will see that even though cost-sensitive models compromises the classification accuracy, the cost dependent loss minimization far exceeds the base logistic regression.

Cost-Sensitive Loss



The above plot shows the tradeoff between classification accuracy and financial loss. Even though the cost-sensitive models compromise the classification accuracy, they outperform the base model by a huge factor(~10) in minimizing the cost-sensitive loss, which is the actual goal. The Neural Networks model is able to outperform both the other models because of its ability to better estimate complex decision functions.

Conclusion

The cost-sensitive example dependent models, clearly, outperforms other models when the goal is to minimize the real-world loss. We also saw the tradeoff between the classification accuracy and loss, i.e. even though a model's classification accuracy is very high, it can still lead to a huge loss in real-world scenarios.

References

- https://towardsdatascience.com/fraud-detection-with-cost-sensitive-machine-learning-24b8760d35d9
- A. C. Bahnsen, D. Aouada and B. Ottersten, "Example-Dependent Cost-Sensitive Logistic Regression for Credit Scoring," 2014 13th International Conference on Machine Learning and Applications, Detroit, MI, 2014, pp. 263-269, doi: 10.1109/ICMLA.2014.48.
- C. M. Bishop, Pattern Recognition and Machine Learning, ser. Information science and statistics. Springer, 2006, vol. 4, no. 4.
- https://towardsdatascience.com/understanding-neural-networks-19020b758230