A Comparative Study on Credit Card Client Default Prediction Using Multilayer Perceptron and Support Vector Machine

Kimon Iliopoulos
kimon.iliopoulos@city.ac.uk

Description and motivation of the problem

In the current market environment, credit card usage is one of the most widespread payment methods around. The more the people that use credit cards, the higher the chances of failing to pay their monthly debt. Thus, it leads to significant challenges for both financial institutions and obligors. It is very important for a financial institution to be able to accurately predict if a client is going to default or not.

The objective of this paper is to critically evaluate two models created to determine whether or not a client will be capable of paying it's credit card debt the coming month. Its primary focus is about predicting with high accuracy the actual default values. A binary variable indicating default on payment (Yes = 1, No = 0) is defined. We will refer to defaults as positive and non defaults as negative results. To address the Class Imbalance Problem (CIP), two main algorithms were used, Support Vector Machine (SVM) and feedforward Multilayer Perceptron (MLP), with different internal and external modifications.

CIP is a challenging Machine Learning research area and has received much attention in recent years. The primary motivation and goal of this study is to try to investigate how this matter can be resolved as several sectors face the same difficulties, e.g credit card fraud detection [13] and cancer classification [6].

Dataset

For the purpose of this project, the "default of credit card clients Data Set" from UCI Machine Learning Repository [20], is used. This dataset is consisted of 24 attributes and 30000 samples. Among others, it includes obligor's demographic characteristics and payment history. During the

preprocessing part undocumented values were deleted and values that did not make sense were aggregated to other categories. Additionally, 6 new features were created indicating how close a client's balance is to their limit, for a certain month. The relief algorithm [10] for feature

selection was applied and the 15 best features were kept for modeling, as the arrow in *Figure 1* shows. Furthermore, due to high imbalance of the target variable (*Figure 2*), 2 balancing methods were covered, adasyn [9] and borderline smote [8]. In this report, we will refer to the imbalanced dataset as original, and to the balanced ones by the technique's name used to achieve that.

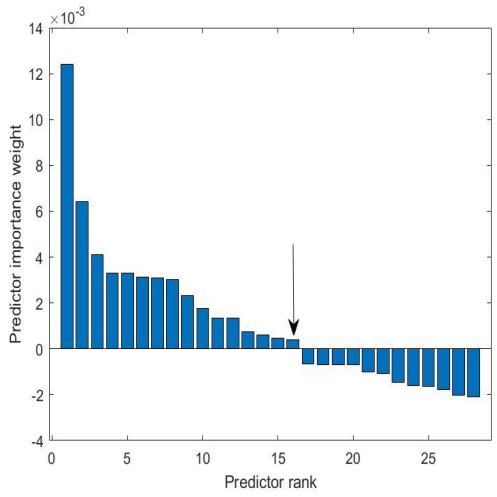

Figure 1: RELIEF algorithm results for feature selection

Figure 2: Bar plots of defaulting and non defaulting counts

Algorithms

- ### Support Vector Machine (SVM)

SVM is a supervised machine learning algorithm which can be used for both classification and regression problems. In the context of binary classification, the objective is to create a hyperplane that separates the two classes in a way that the classification error in novel data is minimized, by maximizing the margin [5]. The maximal margin is a geometrical optimization process, in which the distance between the separation hyperplane and the closest data points from each class is maximized [24].

SVM can be very effective in both linearly and non linearly decision boundaries by applying the kernel trick. Thus, it can solve a variety of tasks using linear, rbf or polynomial kernel. Additionally, SVM is never stuck on local minima, as it is guaranteed to produce a global minima solution [21]. It is also easy to train as it can use only a sample of a dataset and performs really well in small datasets. Contrarily, it requires a lot of computational power and this can be an issue when it comes to high dimensional datasets with lots of instances. Finally, SVM is not that effective in controlling noisy data with overlapping classes [11].

- **Multilayer Perceptron (MLP)**

MLP, is a feedforward artificial neural network that contains more than one, non visible to the user, computational layers. This architecture is called feedforward because every layer's node is fed with information that flows forward through the network, from the input to the output layer [1]. The backpropagation algorithm [16] is the most important component of a feedforward neural network. It is a learning procedure that repeatedly adjusts the connection weights between the network's layers by penalizing them, according to a loss function calculated in the output layer, in order to minimize the difference between the network's output and the associated output [15].

MLPs are capable of detecting complex and non-linear relationships between the dependent and the independent features. In this case, the network adjusts its weights using non linear activation functions that are applied in every hidden and output layer [18]. Moreover, there are a lot of variations of the backpropagation algorithm [10] and these are applied in numerous research areas but less so in business. Apart from the inherent complexity of the algorithm, there is an additional element of difficulty in interpretation, mostly due to the "black box" computations made in the neural network's hidden layers [3]. Finally, while neural networks are prone to overfitting, they are also computationally expensive and require large amounts of data to operate effectively [18].

Methodology

❖ Balancing techniques

The credit default dataset suffers from a highly imbalanced target variable. The negative target values comprise of 20% of the samples. It is challenging for the algorithms to find patterns on this class due to limited occurances during training, which is making them prone to predicting the majority class exclusively. According to literature, external handling of imbalanced datasets includes oversampling the minority class, which is the most commonly used technique and achieves better results, and undersampling the majority class [12]. The methods used for balancing our target class are borderline smote and adasyn. Both of them were applied either to the original dataset or to the dataset acquired after the feature selection to the original.

❖ Feature selection

It was found that some studies applied feature selection [16] and some others did not [17]. For that reason, it was considered necessary to explore both scenarios. As proposed in [20], the Relief algorithm [10] is used for the feature selection and is applied to the original dataset exclusively.

❖ Models

In total, 9 SVM models were trained, tested and compared. At first, we trained 3 SVM models, one for linear, polynomial and rbf kernel to the original dataset, both with and without feature selection. The best performed one, was then trained to the adasyn dataset and the borderline smote dataset. All SVM models, were tested to the original dataset. The hyperparameters tuned were box constraint and kernel scale for every kernel distribution. The optimization process is made using bayesian inference. Lastly, the baseline model was also implemented with the default values given by the MATLAB function *fitcsvm*.

Due to the extremely high computational cost, it was decided to optimise SVMs in 5,000 instances. It was observed that above this number of rows there was no significant improvement in the model performance to novel data. The best arised model was then re-trained, this time to 80% of the dataset, 24,000 samples, and then was tested to the test set, with 6,000 samples. At the same time a 10 fold cross validation was applied to avoid overfitting.

As for the MLP, 9 models were trained, tested and compared to come up with the best one. Firstly, 3 different architectures were trained to the original dataset, for 1 hidden layer of 25 nodes, 2 hidden layers of 25 nodes each and 3 hidden layers of 25 nodes each. Secondly, 3 other architectures were trained to the 15 most important features of the original dataset (*Figure 1*). Nodes per layer were reduced to 15 for 1 hidden layer and 13 for 2 and 3 hidden layers. The best model was then trained to the adasyn and the borderline smote dataset to observe the changes in it's performance. All MLP models were tested to the original dataset. Lastly, the baseline model was implemented with the default values given by the MATLAB function *patternet* to compare the results with the optimised ones.

Every MLP model mentioned, was trained using a grid search optimisation process. The hyperparameters tuned were the activation functions of the hidden layers and the output layer for "tansig", "elliotsig" and "logsig" functions, the momentum, for 0.5, 0.7, 0.9, 0.95, 0.99 and the learning rate for 0.1, 0.01, 0.001, 0.0001 values. The number of epochs is set to 100 for tuning and to 300 for training the optimal model with the best hyperparameter values, while early stopping was applied using validation checks. The training set was composed of 80% of the dataset, 30% of which was used as validation set. The rest, 20%, was used for testing. It is important to mention that MLP's performance depends a lot on the initialization of the weights and *patternnet* function does it randomly. For reproducibility and comparative reasons, a random number generator of the same seed initialized the weights for every MLP model.

- **Evaluation**

The choice of evaluation is very important given that having an imbalanced dataset can cause biased conclusions. In our case, it is very crucial to minimize the False Negative (FN) predictions, as it is very expensive for the card issuer. Accuracy would not be an objective performance metric. For that reason, F1 score and Cohen's kappa were calculated as well.

Literature review and hypothesis statement

There have been a lot of studies about the credit card default prediction. In [17], SVM and MLP were trained in the imbalanced dataset and they both achieved an accuracy of around 81%, whereas balancing the dataset with smote method dropped the SVM's accuracy to 73% and the MLP's remained the same while default prediction increased for both. According to [17], the

highest accuracy achieved for the SVM in unseen data is 79%. Finally, [22] accomplished an accuracy of 83% for the MLP.

To sum up, it is expected that SVM and MLP are going to have similar performance when trained to the original dataset. However, when trained on the balanced dataset we expect a substantial improvement in the minority class prediction and a drop in the total performance of the model. Finally, SVM will probably be much slower than MLP because as C.J. Burges states it is required to "solve a convex quadratic programming problem, since the objective function is itself convex, and those points which satisfy the constraints also form a convex set" [4].

Results, findings and evaluation

❖ Model selection

Table 1, shows the performance of SVM for all the kernels with and without feature selection. They were initially trained in the original dataset to come up with the best model. Linear kernel models achieve the best F1 score and kappa statistic. Also, feature selection improves the performance for the RBF kernel as it overfitted completely when it was trained in the whole dataset. Polynomial of degree 3 and linear kernels performed better without feature selection, but the linear outperformed the polynomial in terms of performance and computational cost.

Model	Dataset trained	Kernel	Feature selection	Accuracy	F1 score	Cohen's kappa	Training time (minutes)
SVM	Original	Linear	Yes	79.8%	46.89%	34.89%	25.2
SVM	Original	RBF	Yes	81.3%	44.22%	34.51%	65.8
SVM	Original	Polynomial	Yes	81.1%	42.03%	32.62%	176
SVM	Original	Linear	No	80.9%	49.87%	38.54%	85.9
SVM	Original	RBF	No	77.8%	0	0	196.5
SVM	Original	Polynomial	No	81.35%	43.63%	34.07%	267
SVM(baseline)	Original	Linear	No	42.9%	37.36%	0.05%	6

Table 1: Performance of different SVM kernels/models trained to the original dataset

Linear kernel model, trained in the most important features (yellow row, *Table 1*), was then trained to the adasyn and borderline smote datasets with the results shown in *Table 2*.

Model	Dataset trained	Kernel	Feature selection	Accuracy	F1 score	Cohen's kappa	Training time (minutes)
SVM	Original	Linear	No	80.9%	49.87%	38.54%	85.9
SVM	Borderline smote	Linear	No	76.78%	52.05%	36.86%	92.7
SVM	Adasyn	Linear	No	79.6%	52.97%	40.02%	62.5

Table 2: Performance of the best SVM kernel/model trained in different datasets

As expected from the literature review and declared in the hypothesis statement, F1 score is increased by 2% for the borderline smote model and 3% for the adasyn while Cohen's kappa is decreased for the borderline smote and increased for the adasyn. The total accuracy dropped by 4% for the smote and by 1% for the adasyn. To sum up, the adasyn model is the best observed one, for both F1 score and kappa statistic being at the same time the fastest of the three with 62.5 minutes required for training.

In Table 3, we calculated the performance metrics for three different MLP architectures with and without feature selection being trained, every time, in the original dataset. It was found that the best performance was achieved for 3 hidden layers with 13 nodes each and by training the model to the most important features.

Model	Dataset trained	Architecture	Feature selection	Accuracy	F1 score	Cohen's kappa	Training time (minutes)
MLP	Original	1 hidden, 15 nodes	Yes	81.3%	44.59%	34.76%	4.1
MLP	Original	2 hidden, 13 nodes	Yes	80.5%	41.74%	31.65%	4.3
MLP	Original	3 hidden, 13 nodes	Yes	81.21%	46.71%	36.3%	9
MLP	Original	1 hidden, 25 nodes	No	80.6%	42.34%	32.19%	5.5
MLP	Original	2 hidden, 25 nodes	No	80.88%	46.43%	35.69%	9.2
MLP	Original	3 hidden, 25 nodes	No	80.3%	40.22%	30.23%	12.1
MLP(baseline)	Original	1 hidden, 10 nodes	No	80.31%	40.44%	30.38%	0.03

Table 3: Performance of different MLP architectures/models trained to the original dataset

As previously, Table 4 shows the performance of the best architecture (yellow row, *Table 3*) this time trained in the adasyn and borderline smote dataset. It is shown that the total accuracy of the model is decreased, same pattern as the counterpart step of the SVM models. Contrarily, F1 score demonstrated a substantial improvement while Cohen's kappa was slightly decreased for both balancing techniques. F1 score takes into account FN values, that means predicting non default for a client that will in reality default. We want FN to be as small as possible as it is

considered to be the most important aspect of the study because it has the highest negative effect for the credit card issuers. Hence the borderline smote MLP (yellow row, *Table 4*) model is determined to be the best one.

Model	Dataset trained	Architecture	Feature selection	Accuracy	F1 score	Cohen's kappa	Training time (minutes)
MLP	Original	3 hidden, 13 nodes	Yes	81.21%	46.71%	36.3%	9
MLP	Borderline Smote	3 hidden, 13 nodes	Yes	74.8%	51.52%	34.95%	7.5
MLP	Adasyn	3 hidden, 13 nodes	Yes	75.38%	50.78%	34.61%	7.1

Table 4: Performance of the best MLP architecture/model trained in different datasets

❖ Evaluation and findings

Now we have ended up with two models, the best MLP, shown in yellow color in *Table 4* and the best SVM shown in yellow in *Table 2*. In *Table 5*, their results are presented adding AUC score, accuracy of default and non default prediction.

Figures 2 and *3* show the confusion matrices and the ROC curves of each model respectively. It is clear that both of them can better predict TN than TP. From the 4,658 non default values, SVM predicted correctly 4,116 and MLP 3,686. The predictive accuracy of the non default class for SVM and MLP is 88.36% and 79.19% respectively (*Table 5*). As for the default class, from the 1,342 instances, SVM predicted correctly 669 and MLP 803, leading to a predictive accuracy of 49.85% and 59.83%. On the one hand, SVM learned to better predict non-defaulters in the next month. On the other hand, MLP achieved 10% better performance in the default prediction with a drawback of 10% in the accuracy of the non-defaulters. Additionally, overall accuracy, F1 score and kappa statistic are higher for the SVM whereas AUC is better for the MLP. Lastly, it is important to state that SVM's training time is 62.5 minutes while MLP's is 7.5 minutes (*Table 5*). Computational power is an important part of every machine learning task and should be taken into account when choosing a model.

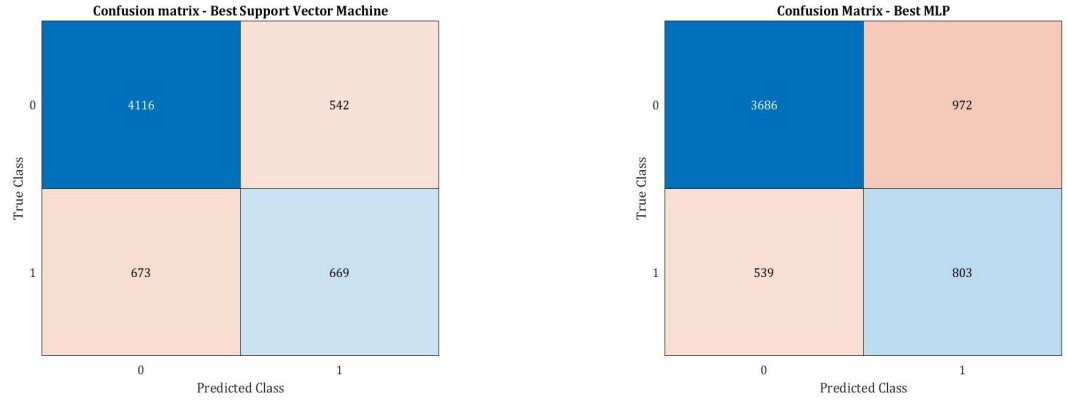

Figure 2: Confusion matrix for the best SVM (left) and the best MLP (right) models

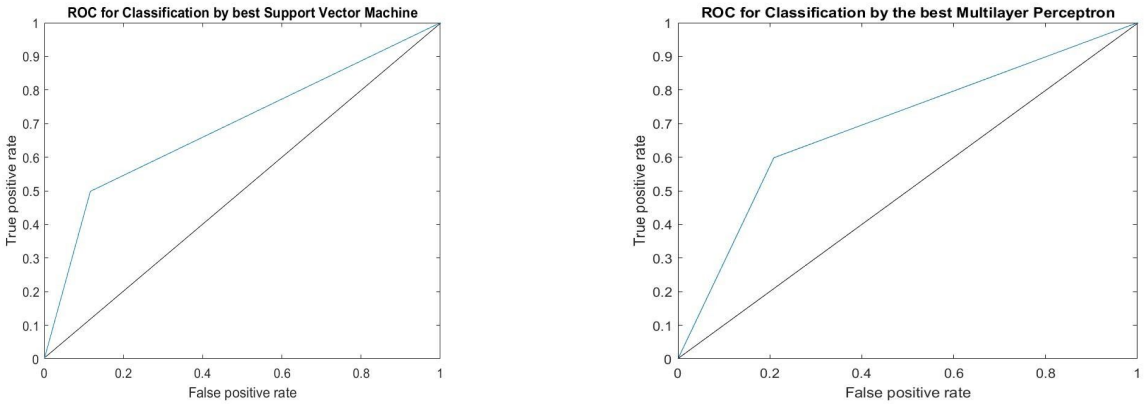

Figure 3: ROC curves of the best SVM (left) and MLP (right) models

In general, SVM models tend to produce suboptimal results when it comes to imbalanced datasets. As [2] states, SVM, apart from the objective of maximizing the margin, is also minimizing the C parameter (box constraint) which plays the role of a misclassification penalty and thus it's goal is to reduce the actual misclassifications. It causes the algorithm to create a hyperplane skewed over the minority class in order to have more correct classifications by predicting the majority class that will lead to better total accuracy performance [2]. Adasyn method has proved worst performance in predicting the minority class of an imbalanced binary classification task compared to smote in various tasks [7]. This might be the reason why SVM has a better overall performance but failed to prove its capabilities in predicting the minority class with the same frequency as MLP. Additionally, Adasyn technique finds the most difficult predictable observations and generates more clones of them [9]. Most of the time, these observations are located close around the borders of the two classes.

MLP has in general a better performance in imbalanced datasets compared to SVM, in predicting the minority class [14][17], even without any balancing applied to the original dataset. This can be explained because the back-propagation algorithms in neural networks repeatedly adjust the weights according to the occurrences of each class during training, amongst other reasons [18].

	Best SVM	Best MLP
Accuracy	79.6%	74.8%
F1 score	52.97%	51.52%
Cohen's kappa	40.02%	34.95%
AUC	69.11%	69.48%
Default accuracy	49.85%	59.83%
Non default accuracy	88.36%	79.19%
Training time	62.5	7.5

Table 5: Performance of the best SVM and MLP models

Conclusion

This study primarily focuses on the CIP, which is considered a challenging ML problem and has received extensive interest by the research community in recent years. Imbalance or even extreme imbalance is prevalent in multiple sectors, such as credit score, fraud detection, cancer and tumor prediction, therefore there is plenty of scope and urgency for further research to improve detection of rare events.

Predicting the default of a credit card client for the next month is considered a very challenging task with significant research around it. In total, 16 models were constructed and the best 2 were analysed deeper. SVM trained in adasyn dataset achieved better performance overall but the MLP trained in borderline smote dataset had 10% higher accuracy in predicting default credit cards. From the credit card issuer's perspective, a wrong non-default prediction (false negative) is more expensive than a wrong default prediction (false positive). The second scenario can be addressed with further modeling while the first one will cause a serious damage for every misclassified instance.

Future work

Future work proposals are focusing on how to further improve the performance of both models in the FN predictions. For the SVM it could be achieved through external or internal learning balancing methods. External methods could include Safe-Level SMOTE, which is considered amongst the best oversampling methods to be applied alongside SVM models in various classification tasks [7]. Internal learning could involve balancing methods via adjustments on the importance of the box constraint parameter in the misclassification penalty for minority and majority classes [2]. Other variations of loss functions that would adjust the weights of an ANN to take into account the CIP by increasing the penalty for misclassifying the minority class [14]. Finally, instead of initializing the weights of the MLP randomly, the method proposed in [23] could be used to achieve better results.

References

1. Aggarwal CC. Neural Networks and Deep Learning. 2018. Epub ahead of print 2018. DOI: 10.1007/978-3-319-94463-0.
2. Batuwita R, Palade V. Class Imbalance Learning Methods for Support Vector Machines. *Imbalanced Learning*; 83–99.
3. Benitez J, Castro J, Requena I. Are artificial neural networks black boxes? *IEEE Transactions on Neural Networks*; 8: 1156–1164.
4. Burges CJ. . *Data Mining and Knowledge Discovery*; 2: 121–167.
5. Campbell C, Ying Y. Learning with Support Vector Machines. *Synthesis Lectures on Artificial Intelligence and Machine Learning*; 5: 1–95.
6. Golub TR, Slonim DK, Tamayo P, et al. Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. *Science*; 286: 531–537.
7. Gosain A, Sardana S. Handling class imbalance problem using oversampling techniques: A review. *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* 2017. Epub ahead of print 2017. DOI: 10.1109/icacci.2017.8125820.
8. Han H, Wang W-Y, Mao B-H. Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. *Lecture Notes in Computer Science Advances in Intelligent Computing*; 878–887.
9. He H, Bai Y, Garcia EA, et al. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)* 2008. Epub ahead of print 2008. DOI: 10.1109/ijcnn.2008.4633969.
10. Kira K, Rendell LA. A Practical Approach to Feature Selection. *Machine Learning Proceedings 1992*; 249–256.
11. Leonard J, Kramer M. Improvement of the backpropagation algorithm for training neural networks. *Computers & Chemical Engineering*; 14: 337–341.
12. Lin C-F, Wang S-D. Training algorithms for fuzzy support vector machines with noisy data. *2003 IEEE XIII Workshop on Neural Networks for Signal Processing (IEEE Cat No03TH8718)*. DOI: 10.1109/nnsp.2003.1318051.
13. Marqués AI, García V, Sánchez JS. On the suitability of resampling techniques for the class imbalance problem in credit scoring. *Journal of the Operational Research Society*; 64: 1060–1070.
14. Ngai E, Hu Y, Wong Y, et al. The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems*; 50: 559–569.
15. Oh S-H. Error back-propagation algorithm for classification of imbalanced data. *Neurocomputing*; 74: 1058–1061.
16. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. *Nature*; 323: 533–536.
17. Soui M, Smiti S, Bribech S, et al. Credit Card Default Prediction as a Classification Problem. *Lecture Notes in Computer Science Recent Trends and Future Technology in Applied Intelligence*; 88–100.

18. Subasi A, Cankurt S. Prediction of default payment of credit card clients using Data Mining Techniques. *2019 International Engineering Conference (IEC)* 2019. Epub ahead of print 2019. DOI: 10.1109/iec47844.2019.8950597.
19. Tu JV. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology*; 49: 1225–1231.
20. *UCI Machine Learning Repository: default of credit card clients Data Set* https://archive.ics.uci.edu/ml/datasets/default of credit card clients (accessed March 30, 2020).
21. Ullah MA, Alam MM, Sultana S, et al. Predicting Default Payment of Credit Card Users: Applying Data Mining Techniques. *2018 International Conference on Innovations in Science, Engineering and Technology (ICISET)* 2018. Epub ahead of print 2018. DOI: 10.1109/iciset.2018.8745571.
22. Xu P, Ding Z, Pan M. A hybrid interpretable credit card users default prediction model based on RIPPER. *Concurrency and Computation: Practice and Experience* 2018; 30. Epub ahead of print December 2018. DOI: 10.1002/cpe.4445.
23. Yam JY, Chow TW. A weight initialization method for improving training speed in feedforward neural network. *Neurocomputing*; 30: 219–232.
24. Yang J-B, Ong C-J. Determination of Global Minima of Some Common Validation Functions in Support Vector Machine. *IEEE Transactions on Neural Networks*; 22: 654–659.

Appendix - Glossary

Activation function In artificial neural networks, the activation function of a node defines the output of that node given an input or set of inputs.
Area Under Curve When using normalized units, the area under the curve is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one.
Bayesian Inference Bayesian inference is a method of statistical inference in which Bayes' theorem is used to update the probability for a hypothesis as more evidence or information becomes available.
Binary classification Binary or binomial classification is the task of classifying the elements of a given set into two groups (predicting which group each one belongs to) on the basis of a classification rule.
Black box A neural network is a black box in the sense that while it can approximate any function, studying its structure won't give you any insights on the structure of the function being approximated.
Box constraint A Matlab parameter that controls the maximum penalty imposed on margin-violating observations, and aids in preventing overfitting (regularization).
Class Imbalance Problem (CIP) CIP is when each class does not make up an equal portion of your dataset.
Cohen's kappa Cohen's kappa coefficient (κ) is a statistic that is used to measure inter-rater reliability (and also Intra-rater reliability) for qualitative (categorical) items.
Confusion matrix A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.
Credit balance A credit card balance is the total amount of money you owe to your credit card company.
Default credit Default is the failure to repay a debt on a loan or security.
Elliotsig Elliot symmetric sigmoid transfer function.
F1 Score The F score, also called the F1 score or F measure, is a measure of a test's accuracy. The F score is defined as the weighted harmonic mean of the test's precision and recall.
False Negative A result that appears negative when it should not.
Fitcsvm fitcsvm trains or cross-validates a support vector machine (SVM) model for one-class and two-class (binary) classification on a low-dimensional or moderate-dimensional predictor data set.
Global minima A global minimum, also known as an absolute minimum, is the smallest overall value of a set, function, etc., over its entire range.
Hyperparameters In statistics, hyperparameter is a parameter from a prior distribution; it captures the prior belief before data is observed. In any machine learning algorithm, these parameters need to be initialized before training a model.
Hyperplane In geometry, a hyperplane is a subspace whose dimension is one less than that of its ambient space.
Kernel scale Kernel scale or gamma parameter defines how far the influence of a single training example reaches, with low values meaning 'far' and high values meaning 'close'.

Kernel The kernel function is what is applied on each data instance to map the original non-linear observations into a higher-dimensional space in which they become separable.
Learning rate The learning rate hyperparameter controls the rate or speed at which the model learns.
Linearly separable In Euclidean geometry, linear separability is a property of two sets of points visualized in 2 dimensions.
Local minima A local minimum, also called a relative minimum, is a minimum within some neighborhood that need not be (but may be) a global minimum.
Logsig Log-sigmoid transfer function.
Momentum Momentum is group of tricks and techniques designed to speed up convergence of first order optimization methods like gradient descent (and its many variants).
Noisy data Noisy data is data with a large amount of additional meaningless information in it called noise.
Obligors A person who owes or undertakes an obligation to another by contract or other legal procedure.
Optimization Optimization is a process that minimizes the loss function.
Overfitting Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data.
Overlapping class In overlapping classes, data samples appear as valid instances of more than one class which may be responsible for the presence of noise in data sets.
Oversampling Oversampling is a technique of generating new observations of the minority class to balance an imbalanced dataset.
Patternnet Patternet trains the generic feedforward neural network feedforwardnet to map each input vector into its corresponding target vector.
Softmax Function that takes as input a vector of K real numbers, and normalizes it into a probability distribution consisting of K probabilities proportional to the exponentials of the input numbers.
Tansig Hyperbolic tangent sigmoid transfer function.
Traingdx Gradient descent with momentum and adaptive learning rate backpropagation.