Review and matlab implementation of the paper:

Using neural network ensembles for bankruptcy prediction and credit scoring

Filippo Tolin - 874631

Ca' Foscari University of Venice

March 28, 2024

Summary

1. Paper review
   - Goals
   - Tools
   - Advancements vs previous works
   - Study 1,2,3
   - Conclusions
2. A couple of remarks...
3. Matlab implementation
   - Methodology
   - Model
   - Single classifier
   - Multiple classifier
   - Diversified multiple classifier
   - Results of Study 1 and 2
   - Type 1 and 2 error

Paper review

Goals

- Observe the performance differences between different ANN ensemble approaches, namely single classifiers, multiple classifiers and diversified multiple classifiers, with regards to based on a set credit scoring and bankruptcy detection. The study is based on three of heterogeneous datasets;
- Evaluate the three classifier architectures performance with regards to Type 1 error and Type 2 error.

Tools

- Multilayer Perceptron feedforward Artificial Neural Network
  - One **hidden** layer
  - Five different values for the **hidden nodes** hyperparameter
  - Four different values for the **training epochs** hyperparameter
- Multiple classifiers with two techniques to compute them:
  1. Best n classifiers for every epoch
  2. Best n classifiers among all the epochs

Note

Technique 1 is only applicable to $n = 3 \wedge n = 5$, with $n \in [3, 5, 7, 9, 11, 13, 15]$. Multiple classifiers are based on **majority voting**.

Advancements vs previous works

- Employment of **multiple datasets** for system validation;
- Usage of **Type 1 and Type 2 errors** and not only average accuracy measures;
- Testing the classifiers performance on multiple classification tasks rather than a single one, specifically on **credit scoring** and **bankruptcy prediction**.

Brief remark

- **Type 1** error is associated with **false positives**;
- **Type 2** error is associated with **false negatives**.

Examples

- *Type 1*: the model classifies a credit-worthy client as a credit-risky one;
- *Type 2*: the model classifiers a credit-risky client as a credit-worthy one.

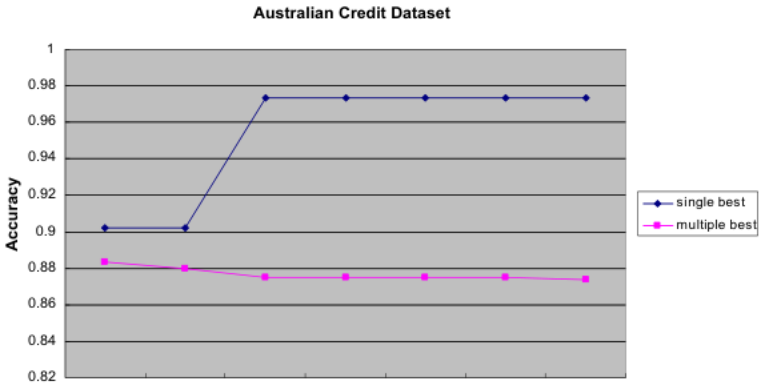
Study 1: single vs multiple classifiers

- Datasets are split into training (70%) and test (10%);
- For single classifiers the number of nodes is $nn \in [8, 12, 16, 24, 32]$ and learning epochs $[50, 100, 200, 300]$.
- Multiple classifers are build with the voting strategy combining the results of the top n classifers, with $n \in [3, 5, 7, 9, 11, 13, 15]$.

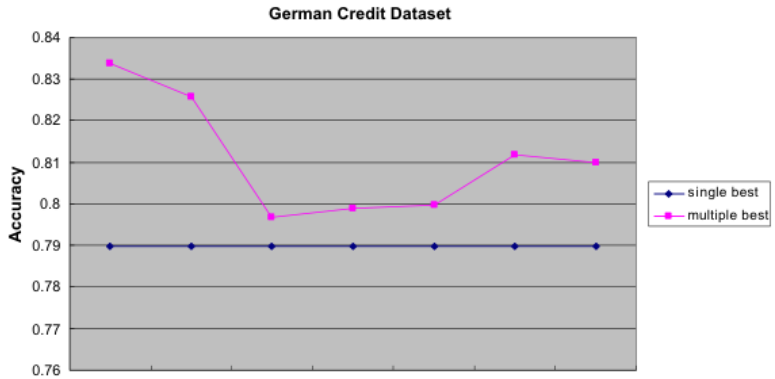
Takeout

On average, the single best classifer outperforms multiple classifers.

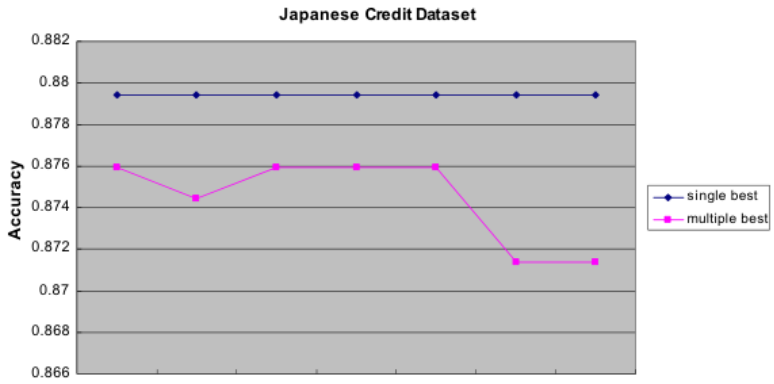
Study 1: single vs multiple classifiers



(a) Australian



(b) German



(c) Japanese

Figure: Comparison between single classifiers and multiple classifiers.

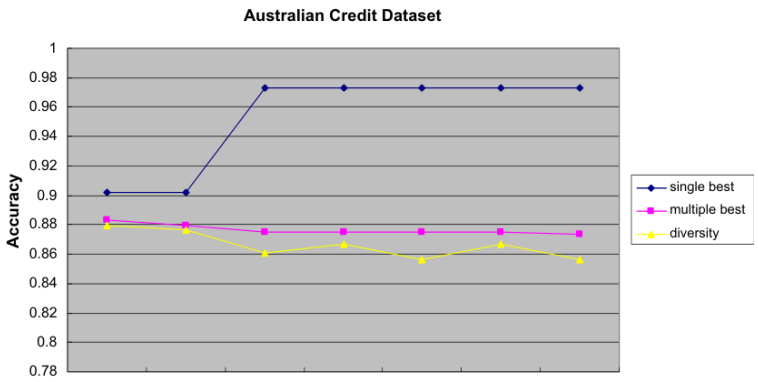
Study 2: single vs multiple vs diversified classifiers

- Train-test dataset generation is different for diversified multiple classifiers. Specifically, every model composing the classifier is trained on a fraction of the observations from the same dataset, then the majority voting is executed using a test dataset;
- The procedure aims at ensuring **diversity** between classifiers.

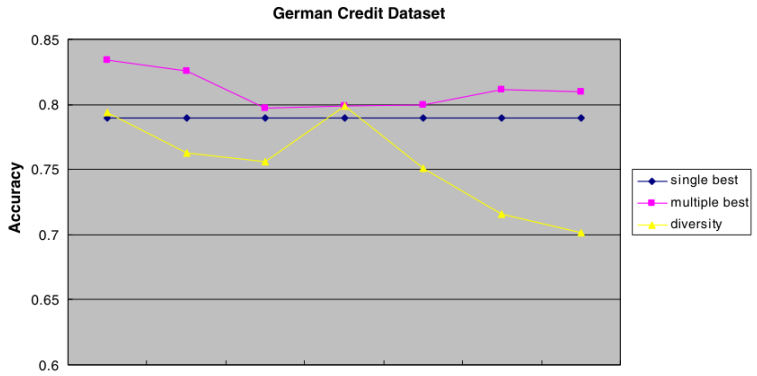
Takeout

The best single classifier is still, on average, a better classifier than the diversified multiple classifier (and the multiple classifier, as seen before).

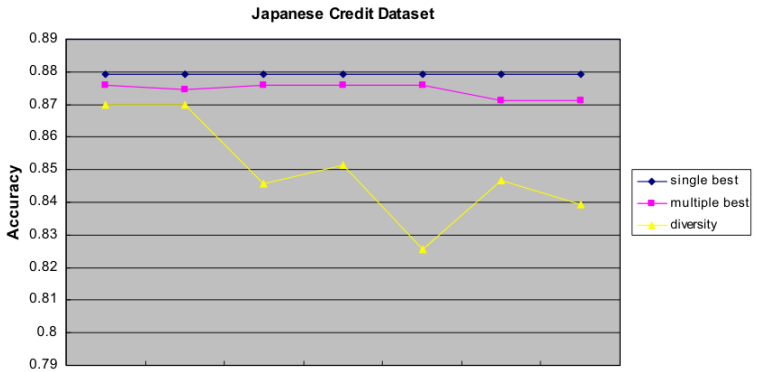
Study 2: single vs multiple vs diversified classifiers



(a) Australian



(b) German



(c) Japanese

Figure: Comparison between single, multiple and diversified classifiers.

Study 3: Type 1 and Type 2 errors

- In Study 1 and Study 2 the results of classifiers are compared based on the **accuracy** of the classifiers. Study 3 compares the models performance with regards to Type 1 and Type 2 errors.

Takeout

This study highlights how single classifiers do not totally outperform multiple or diversified classifiers.

Study 3: Type 1 and Type 2 errors

Average error rate of Type I and Type II errors

	Australian credit			German credit			Japanese credit		
	S	M	D	S	M	D	S	M	D
Type I error	*12.16*	12.85	14.28	*44.27*	45.25	59.82	15.02	15.07	*14.42*
Type II error	12.97	12.14	*11.55*	9.48	*8.46*	8.67	10.79	*10.00*	14.06

Figure: Type 1 and Type 2 error across datasets and classifier architectures.

Conclusions

- If the performance is measured with **accuracy**, **single best neural network classifier is more suitable** for bankruptcy prediction and credit scoring tasks, if compared with multiple or diversified multiple neural network classfiers;
- If performance is measured with **type 1 and 2 errors**, there seems to be no **clear winner** among the model architectures analysed.

A couple of remarks. . .

A couple of remarks...

- The methodology used to build the datasets for diversified multiple classifers could be associated with the poor performance by the classifiers. In fact the higher the n the smaller the train-sets used for each models' training. This could lead to a lot of results variability;
- It's not clear why the authors used two methodologies to build the multiple classifiers even though one of them is applicable only to $n = 3$ and $n = 5$ classifiers.

Matlab implementation

Methodology

- **Pre-processing**. The datasets used for the matlab implementation are (presumably) the same datasets used by the authors of the paper. The datasets pre-processing, specifically:
  - **feature normalization**. In order to apply ANN, all the continuous features need min-max normalization;
  - **one-hot-encoding**. In order to apply ANN, all the categorical features need a different encoding, specifically one-hot-encoding.
- **Datasets**. The three datasets were retrieved from the UC Irvine Machine Learning Repository.
  - Australian (690 x 15);
  - German (1000 x 20);
  - Japanese (690 x 16).

Model

```
epochs = [50 100 200 300];
hidden_nodes = [8 12 16 24 32];

net = fitcnet(X_train, Y_train,...
   'Activations','sigmoid',...
   'IterationLimit',epoch,...
   'LayerBiasesInitializer','ones');
   'LayerSizes', node,...
testAccuracy = 1 - loss(net,X_test,Y_test,...
"LossFun","classiferror");
```

Single classifier

Methodology

For the single neural network classifier, the dataset is divided into test (70%) and train (30%); then the implementation execute a loop that tests all the possible combinations of hidden nodes and epochs.

Multiple classifiers

Note

Since there seem to not exist a `MATLAB` function for the multiple classifier for `fitcnet`, the implementation has been done by hand.

- For every n and dataset, n models are trained based on the hyperparameters of best n single classifiers;
- Then predictions are formulated by every model with the test dataset;
- The results of the predictions from n models undergo the majority voting process, where the resulting vector is the vector of the n multiple classifier predictions.

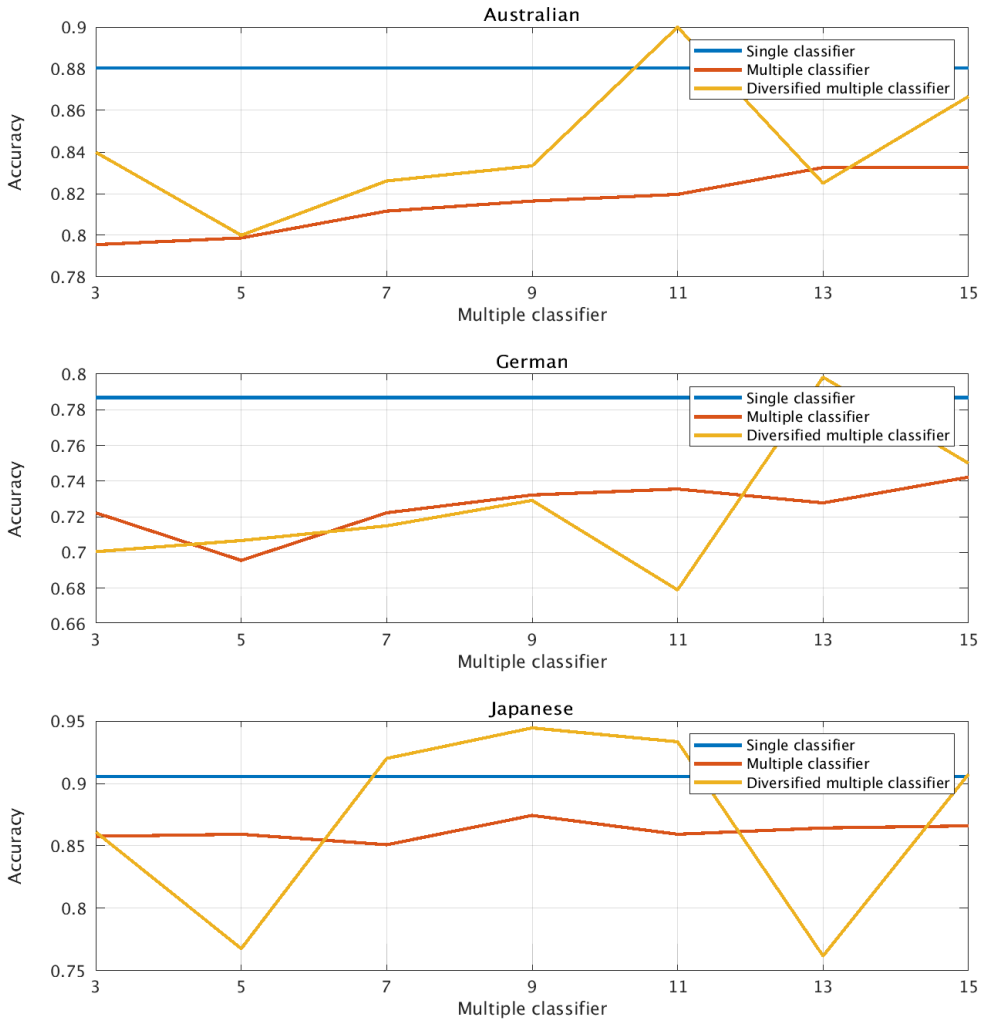
Diversified multiple classifier

- Same structure of multiple classifier, but every model is trained on a different subset of observations from the dataset;
- The algorithm for diversified dataset creation is the following:

Dataset splitting

$split = \frac{nrows(dataset)}{2 \times n + 1}$, then for every classifier the train dataset is $2 \times split$. Test dataset is of size $split$.

Results of Study 1 and 2



Results of Study 1 and 2

Takeout

From this results can be confirmed that the best single classifier ANN, on average, performs better than the multiple and diversified multiple classifiers across the three datasets and the different classification tasks.

Note

The exception of multiple classifiers outperforming the single classifiers happens for the japanese dataset, whereas on the paper implementation happens with the german dataset.

Type 1 and 2 error: setup

- **Single classifier**. For every combination of hyperparameters, the confusion matrix is computed and thus derived the type 1 and 2 errors;
- **Multiple classifier**. The same procedure is executed for every classifier.

Type 1 and 2 error: results

Table: Single classifier

Error type	Australian	German	Japanese
1	0.2107	0.1787	0.2309
2	0.2248	0.5384	0.1675

Table: Multiple classifier

Error type	Australian	German	Japanese
1	0.0900	0.1605	0.1771
2	0.1977	0.5373	0.1045

Type 1 and 2 errors: results

Takeout

As according to the paper authors and findings, the performance measured with type 1 and 2 error leaves more doubts about a clear winner model architecture.

References

Chih-Fong Tsai, Jhen-Wei Wu (2008)

Using neural network ensembles for bankruptcy prediction and credit scoring

Expert Systems with Applications 34(4), 2639 – 2649.