
Supervised classifiers for predicting teamwork effectiveness in software engineering projects using NNs

1 Description

An interesting application of machine learning methods is predicting teamwork effectiveness in software engineering projects [1, 2, 3]. This is a complex scenario since the features that describe the characteristics of projects and participants are generally noisy or subjective. Also, the number of instances or cases available for these problems is usually limited. Therefore, it is relevant to investigate the performance of different classifiers in this domain.

2 Objectives

The goal of the project is to apply a set of classifiers to predict, starting from different sets of features that describe the work of 74 students teams, (Problem 1: Whether the project has been above or below expectations; Problem 2: Whether the product has been above or below expectations). This means that two different classification problems have to be solved using the same set of features. A database will be used for the analysis ¹. Three supervised classifiers should be created to solve each classification task and the accuracies produced by all these classifiers should be computed.

The project should apply Neural Networks for: I) Find suitable feature representations for this problem that are very usable for other ML classifiers, OR, II) Implement NN-based classifiers for this problem, OR III) The combination of I and II (e.g., using an RBM to find the features and a Multi-layer Perceptron to classify the problem using the extracted features). In case II), the students are free to decide which feature representation is more appropriate for the data. In case I), they can use any classifier with the NN-based features.

The student should: 1) Design any preprocessing of the time series the dataset; 2) Define and learn the classifier for any of the two problems using the training data. 3) Design the validation method to evaluate the accuracy of the proposed classification approach. 4) Answer to the following questions in the report:

- What class of problems can be solved with the NN? (e.g., supervised vs unsupervised problems)
- What is the network architecture? (e.g., type and number of layers, parameters, connectivity, etc.).
- What is the rationale behind the conception of the NN?
- How is inference implemented? (e.g., How is the information extracted from the network?). Type of prediction or type of inference process.
- What are the learning methods used to learn the network ? Algorithms used for learning the network.

¹The “Data+for+Software+Engineering+Teamwork” dataset can be downloaded from <https://archive.ics.uci.edu/ml/datasets/Data+for+Software+Engineering+Teamwork+Assessment+in+Education+Setting>

As in other projects, a report should describe the characteristics of the design, implementation, and results. A Jupyter notebook should include calls to the implemented function that illustrate the way it works.

3 Suggestions

- Check the description of the data in README.txt the dataset file <https://archive.ics.uci.edu/ml/machine-learning-databases/00393/SETAP%20FINAL%20DATA.7z>.
- Read paper describing the original approach to this data [3].
- See example notebook for work projects.
- Implementations can use any Python library.
- If classes are not well balanced you may use performance measures different to the accuracy.

References

- [1] Shihong Huang, Dragutin Petkovic, Kazunori Okada, Marc Sosnick, Shenhaochen Zhu, and Rainer Todtenhoefer. Toward objective and quantitative assessment and prediction of teamwork effectiveness in software engineering courses. *ACM SIGSOFT Software Engineering Notes*, 38(1):7–9, 2013.
- [2] Máira Marques, Sergio F Ochoa, María Cecilia Bastarrica, and Francisco J Gutierrez. Enhancing the student learning experience in software engineering project courses. *IEEE Transactions on Education*, 2017.
- [3] Dragutin Petkovic, Marc Sosnick-Pérez, Kazunori Okada, Rainer Todtenhoefer, Shihong Huang, Nidhi Miglani, and Arthur Vigil. Using the random forest classifier to assess and predict student learning of software engineering teamwork. In *Frontiers in Education Conference (FIE), 2016 IEEE*, pages 1–7. IEEE, 2016.