Reproducibility review of: A machine learning based approach for predicting usage efficiency of shared e-scooters using vehicle availability data

Carlos Granell

2022-06-10

This report is part of the reproducibility review at the AGILE conference. For more information see https://reproducible-agile.github.io/. This document is published on OSF at https://osf.io/djfc2/. To cite the report use

> *Granell, C. (2022, May 10). Reproducilbility review of: A machine learning based approach for predicting usage efficiency of shared e-scooters using vehicle availability data. https://doi.org/10.17605/OSF.IO/DJFC2*

Reviewed paper

> *Zhao, P., Li, A., Pilesjö, P., and Mansourian, A.: A machine learning based approach for predicting usage efficiency of shared e-scooters using vehicle availability data, AGILE GIScience Ser., 3, 20, https://doi.org/10.5194/agile-giss-3-20-2022*

Summary

The provided workflow was **partially reproduced**. The authors provided a detailed description of the data sets in a Data and Software Availability section. Access to a processed data set of the e-scooter sharing vehicle data Service in Stockholm, Sweden, is provided along with Python scripts to run three machine learning methods: Logistic Regression (LR), Artificial Neural Network: Multilayer Perceptron (MLP), and Random Forest (RF). The implementation of these ML methods is based on the Python library `Scikit-learn`.

The reproduction described in this report uses the Python code provided in a Github repo. The results reported here refer to Figure 5, which is a bar chart comparing the performance evaluation metrics (accuracy, F1, precision and recall) of the three ML methods. Nevertheless, no code is provided to visually recreate the figure, but the scripts produce the required data to create that figure. For the rest of figures and tables, no code is provided.

Reproducibility reviewer notes

The original paper submission did not provide a link to a code/data repo, but it was added after contacting the authors. Starting with the repo https://github.com/micromobility-research/usage_efficiency_prediction, I downloaded the dataset and went to the steps below

```
git clone https://github.com/micromobility-research/usage_efficiency_prediction

mkvirtualenv agile2022-003

# Download dataset 'Stockholm_data_for_training.pkl' in agile2022-003

pip install pandas geopandas sklearn scipy matplotlib seaborn eli5
```

Execution of the RL model with the provided dataset (blue bar in Figure 5)

```
# in ./agile2022-003
python3 LR.py

X1: accuracy using user specific features = 63.17%
[[ 19599  63539]
 [ 14036 113468]]
F1:
0.7452472981271614
precision:
0.6410367951549939
recall:
0.889917179068892
OK
```

Execution of the ANN model with the provided dataset (orange bar in Figure 5)

```
# in ./agile2022-003
python3 ANN.py

X1: accuracy = 69.18%
[[ 39744  43394]
 [ 21525 105979]]
F1:
0.7655312647854462
precision:
0.7094923446673763
recall:
0.8311817668465303
OK
```

Execution of the RF model with the provided dataset (green bar in Figure 5)

```
# in ./agile2022-003
python3 RF.py

X1: accuracy using test = 71.23%
[[ 43227  39911]
 [ 20696 106808]]
[0.0795066  0.06362379 0.10772698 0.11532842 0.04677267 0.11871834
 0.11085419 0.11827179 0.11544051 0.1237567 ]
F1:
0.7789864453382832
precision:
0.7279766083465673
recall:
0.8376835236541599
```

The results obtained, the performance evaluation metrics per each model, are consistent with the values charted in Figure 5. I did not recreate Figure 5 because no code is provided, even though I could code a similar barchart as the one in Figure 5. Yet, the `RF.py` script was killed and did not end properly. The second task of the script, which is aimed to create a boxplot to represent the feature importance of RF model (I guess, Figure 6) failed.

Comments to the authors

- The provided scripts only generate the required data (printed in the console) for Figure 5. *I recommend adding a script to generate the actual bar chart depicted in Figure 5.*
- Some scripts take some time. *I recommend adding estimated execution time to each script, as the ANN and RF scripts took a long time to execute.*
- The README file of the github repo is minimal, limited to the abstract and a link to the dataset on a Drive folder. *I recommend adding usage information to the README file to associate each script to the expected output in the paper, and to add a* `requirement.txt` *file or similar to indicate the required packages and their versions to run the analysis.*