DSCI 552: Machine Learning for Data Science

Problem Set 2

Instructor: Dr. Marcin Abram – email: mjabram@usc.edu

Deadline: Thursday, February 18, 2020 at 10 am PDT

You can submit the solutions (pdfs) on Blackboard. The deadline to submit solutions is always the second Thursday at 10 am (PDT). As long as the problem set is open, you will be able to send multiple answers (the last submitted answer will matter).

Task (20 point)

You are a new hire in a mid-size company. You have just completed your first task. Today morning, you handed your report in and now you circle around the office. You feel exhausted and a bit nervous. You don't really know what to expect next...

After the lunch, you were approached by your technical manager. "Our boss wants to see you – we have to discuss the next task for you", he communicated.

Following your manager, you enter the meeting room. The CEO and the senior developer already wait inside.

Your CEO said: "You did an excellent job! We are truly lucky, that you decided to join our firm. Our clients were very please with the model that you designed for them. The report was very helpful – they highlighted, that it was important for them to correctly understand the limitations of the model and to know the expected performance after it is deployed in a real world.

However, now we have another task for you. We want to start a trial with a major hospital. I want you to prepare a proof-of-concept, so we can convince them, that a partnership with our firm can be beneficial for them.

You will get a historical medical data. I want you to design a model, that can classify, if a certain treatment is recommended for the patient or not. Additionally, there are 5 additional features (denoted as `MeasureA`, `TestB`, `GeneA`, `GeneB`, and `GeneC` in your dataset) that we can use. However, they are really expensive and difficult to collect. I want you to assess, how useful they are.

We meet with the hospital in two weeks. I want a detailed report describing your main findings, on my desk, on Thursday, February 18, at 10 am."

Hint: Your boss called your task "proof-of-concept" but in fact, the nature of that assignment is the same as the last time. You are asked to train a classification model and you must measure how good that model is. Additionally, you must give recommendations which features are important to collect. You should look at all variables, but at minimum, you should test the importance of `MeasureA`, `TestB`, `GeneA`, `GeneB`, *and* `GeneC`.

Your Technical Manager said: "This time it really matters, that your model has a good performance. If we can show, that our model makes less mistakes than a human doctor, it would be a big deal. Describe exactly how you tested your model. They are really going to look at that section. Additionally, similar to the last time, the interpretability of the model is very important. You should restrict yourself to logistic regression."

Hint: Remember, that accuracy alone, is not a good measure. We care both, about accuracy and precission. Report also false positive and false negative. To chose a right model, you can use for example the AUC score. It is ok (it's even expected) that you will do some feature engineering. You can also try to add regularization to your logistic regression[1] *and test if it helps you or not. To show that the model can be interpreted, you can indentify and explain the most important relations between the variables and the expected outcome (e.g., how the probability that the treatment is recommended changes with age? Or gender?)."*

The senior developer took you aside and said: "My task will be to prepare a technical demo based on your work. To do this, I need your code. Remember, that I'm not a Data Scientist like you – so you have to be very careful when you are writing your code. Write comments and try to explain any nontrivial section – so I'm not confused when I read your code."

Hint: In the ideal case, people should be able to take your code, run it, and recreate all your results. In a less-ideal case, it should be a demonstration of a "typical run". The code should demonstrate your approach end-to-end. People should just specify the path to the dataset, run it, and see the final results.

Learning Objective: You will be able to train and evaluate logistic regression. You will be able to create a technical report. You will be able to communicate your findings with several people that represent different archetypal roles.

[1] Check the `penalty` parameter in `sklearn` LogisticRegression function. Read more on `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html`

Data

First, enter the competition by accepting the terms at `https://www.kaggle.com/t/8d1413df8de447afa5e649549d05770a`. Then, you will be able to find the dataset at `https://www.kaggle.com/c/usc-dsci552-section-32415d-spring-2021-ps2/data`. There are two files. The first file, `ps2_available_dataset.csv` should be used to complete this assignment. The second file, `ps2_kaggle_predict.csv` can be used for the (optional; non-obligatory) in-class Kaggle competition.

Report

To help you, I prepared a template for you, see `https://www.overleaf.com/read/qcbcsdscvcdb`. You are encouraged to use the template, but if you prefer to use another editor – or to made some modifications – this is acceptable. Just remember to always export your report to the pdf format at the end.

Report Submission

Submit your report (a pdf file) using Blackboard (check the "Assignments" section on Blackboard page of our class).

Code Submission

To accept the assignment, use the secret link `https://classroom.github.com/a/oPbHKV6k`.

Upload your code to a private GitHub repository linked to our GitHub Classroom, `https://classroom.github.com/classrooms/77759207-usc-dsci552-32415d-spring2021-classroom`.

Please, remember that your GitHub name is linked to your student profile in our GitHub Classroom. If you use Python, you can commit `.py` or `.ipynb` file(s). If you use other programming languages, like `R`, `C++`, `Java`, `Julia`, etc., prepare a description that would explain how to compile (if necessary) and run your code. If you submit more than one file (or one Jupyter Notebook), include `Readme.md` file, where you would explain in which order we should read the files.

Grading Rules

In order to grade your work, we will role-play the following situation. We will assume, that you are a new-hire in our company. You were asked to

provide a comprehensive technical report, that illustrates your findings. We will evaluate it from the perspective of three people.

- Your CEO (she would like to hear the high-level stuff; She will probably read only the abstract and the conclusions, maybe she will also look at the main figure or the main plot). *(4 points for the report)*
- Your manager (he would like to see a detailed report; he might also look at some parts of the code). *(6 points for the report and 2 points for the code)*
- A senior developer (they would like to see your code; they might not read the report at all). *(8 points for the code)*

Your final score is the sum of scores given by each person. Cumulatively, there are 10 points for the report and 10 points for the code.

Each person (the CEO, the manager, the developer) will use the following grade rubrics.

Grade Component	Meets Expectations	Approaches Expectations	Needs Improvement
Completeness			
Clarity and Support			
Validity			

By "Completeness" we mean, that all parts of the question are addressed. By "Clarity" we mean, that the text and the code is written in accessible way. By "Support", we mean that you provide sufficient evidences to back-up your statements. You must cite any source you use (even if you happened to copy or adapt a snippet of a code from the internet – you should still treat it as a citation. You should clearly mark how large is that snipped and provide adequate references).

Optional Challenge

We also created an optional challenge for you. There is no additional credits for participating in it[2]. However, we encourage you to try. To try, go to `https://www.kaggle.com/t/8d1413df8de447afa5e649549d05770a`. You will find there a special file, called `ps2_kaggle_predict.csv` – it is a special test dataset, where I removed the "treatment" column. Train a model (you are not restricted to linear models anymore – you can use whatever you want)

[2]Grading students based on the in-class competition is a very bad educational technique! We try to avoid it in this class. However, creating a small competition, and making the participation optional – this could be fun!

and make your predictions. On Thursday, February 18, during our lecture, I will present the leaderboard.

Have fun!

Don't Panic

Don't panic. We understand, that this is a large, open-ended task. We also understand, that this might be your second technical report that you were tasked to write. We are dedicated to help you do the best work – and while we keep high standards for you (and for us) – in the same time, we acknowledge, that you have limited time and limited resources to complete your task. This report doesn't have to be perfect to give you 100% score.

If you don't know where to start, read the *third* chapter of "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow", 2nd Edition by Aurélien Géron. Check also the *Appendix B. Machine Learning Project Checklist* from that book.

If something is not clear, ask questions on Piazza.