Quick intro about the TFM

Master in Data Science, streaming edition 04

Goals of the TFM

- Students will show their capacity to work as data scientists
- Starting from raw data, all the way up to solving a problem or research question with data.
- TFM will include the following components:
  - Data acquisition
  - Data cleansing and preparation
  - Analysis
  - Frontend / visualization
- The question can be refined after several iterations working with the data

Deliverables

- Repository with all the code and ready to be executed
  - **Please provide instructions** to install any package or dependency. Easiest to use a dependency/package manager like conda and an `environment.yml` file.
  - The data necessary to fully replicate the work. It should not be included in the repo- repos are for code. It can be provided via Drive or similar file sharing service, with clear instructions in the repo on how to download it and where to place it. **Important**: be aware that when I mount a Drive in colab, *I'm mounting mine*. Make sure to give access to daniel@mateos.io and igor.arambasic@gmail.com to the data.
  - The notebooks should include all the cells executed and their results so that even if you don't want to replicate the code you can see the outcame.
  - If the repo is private, give us access to it: GitHub usernames danimateos & IgorAramb.
- Document explaining the TFM: a *memoria*
  - Any data scientist with the document and the repository should be able to replicate the TFM
- Interactive Frontend.
  - What the customer/user would interact with.
- TFM defense: in person presentation covering the TFM. Explain data, internal structure, design decisions, results and conclusions

Suggested structure of the document

- Introduction: what, why, why is it relevant, any previous related work/state of the art
- Raw data description: include a data dictionary explaining the semantics of the fields used.
- Methodology: machine learning techniques used, statistical methodologies.
  - Explain what you did and why you did it. You don’t need to include every dead end that you explored but didn’t work- that can be explained in the notebooks.
- Summary of main results
  - Detailed results will be available by running the code in the repo
- Conclusions
  - Not a summary of the work. The problem was relevant, now with your work, what can you say about how the problem is solved?
- User manual for the frontend

Technical restrictions

- None
- You can use any technology
  - As long as the evaluators can get access to that technology too
- The evaluators need to have access to the data, too.
  - NDAs have been signed in the past for sensitive data and projects.

Phases during the master

- First tutoring session: short presentation of the planned project
  - Not graded, serves to help you stay on track and decide on a plan.
  - By this point you should have at least had a look at the data and evaluated the viability of several options.
  - July 19/20, 2022
- Write message in Online Campus specifying topic, link to repo and group
  - **Message topic should start with [TFM]**
  - **1 or 2 people per group**
  - **Deadline: July 31st, 2022**
  - PLEASE DON'T WRITE ME AN EMAIL, post in Online Campus
- Second tutoring session: approximately two thirds of the way through the master
  - The objective is to solve roadblocks that you are facing.
  - By this point you should be on full TFM mode.
  - **November 16/17th, 2022**
- Deliver repo and main document
  - Include the document in the repo
  - **Deadline: February 10th, 2023**
- Defense: an in person presentation covering the TFM. Explain data, internal structure, design decisions.
  - **February 21st, 2023**

Recommendations with the repo:

- Public repo for greater visibility
  - Better future opportunities
- **Please don’t upload private data to the repo**
- Clean repo, with proper README.md
  - It’s best if you keep the *memoria* and README separate.
  - The README should be intended for anyone that arrives at the repo, not only the reviewers:
    - One paragraph describing the project.
    - Objective.
    - One or two graphs or figures if it makes sense.
    - Instructions to get the data and run the code.
- Preferably in English
  - But not mandatory

Evaluation criteria

Six components of the grade:

- Clarity of the documentation (*memoria*)
- Replicability
- Complexity: the minimum required grade in this aspect to pass is a 3
- Clarity and correctness of source code
- Relevance and fit-to-purpose of the chosen analytical methods
- UX and usability of the frontend
    - We will evaluate as the “consumer” of the data product, with zero knowledge of data science

All scored from 0 to 10 points and worth equally.
We evaluate the last repo snapshot committed prior to the end of deadline. If we need to evaluate the commit after the deadline the penalization is 0.1 points per hour of delay.

Replicability
Clarity in the documentation
Clarity and correctness of source code

- Memory and notebooks (or other kind of source) are complementary. They can’t both be light on details.
- It’s useful to have comments in the source. They should explain clearly the *why* the choices are made.
- The memory should be clear and concise, but complete. The target audience is a fellow data scientist.
    - Imagine that you are handing over a project at work. The code and memory should be enough for a reasonably skilled data scientist to take over the project without needing to consult you.

Complexity

- In the data and the analysis methodologies
- More complex is better, as long as it makes sense
- It should reflect a reasonable amount of effort (and results!).
- Minimum complexity required is 3: If we consider it's under 3, this attempt will be failed and you will have to do a later second (and last!) attempt at a TFM.
- Some example for you to get an idea of the criterion:
  - A notebook with complexity around 1
  - A notebook with complexity between 1 and 2
  - A notebook with complexity around 3
  - A notebook with complexity around 4
  - A notebook with complexity around 7

Relevance and fit-to-purpose of the chosen analytical methods

- Do the choices make sense?
- Are the metrics evaluated appropriately?
- Is the separation between training and testing data well maintained?
- Can the method or algorithm chosen reasonably do what was expected of it?

Frontend

- An interactive data product
- Don’t worry, it’s very easy with streamlit or Shiny!
- The target audience is the intended end user, which is different for different projects.
  - A member of the public?
  - A particular role at a company?
  - No data science knowledge.

Defense

- It’s your opportunity to show off your shiny new project and to respond to criticism.
- Not graded itself but can influence your overall grade.
  - For example, we might understand better the complexity of the project, or the reasons for decisions taken.
  - We study the projects before the defense. Most presentations lead us to reevaluate one or more aspects of the project.
- You will explain your project and present a quick demo of the frontend.
- Then we will ask questions or ask you to explain decisions taken.
- Focus on the project- your personal experience can be commented on and adds color, but should never be the main focus.
- Think about the time you have and what you want to use it for. Rehearse the presentation!

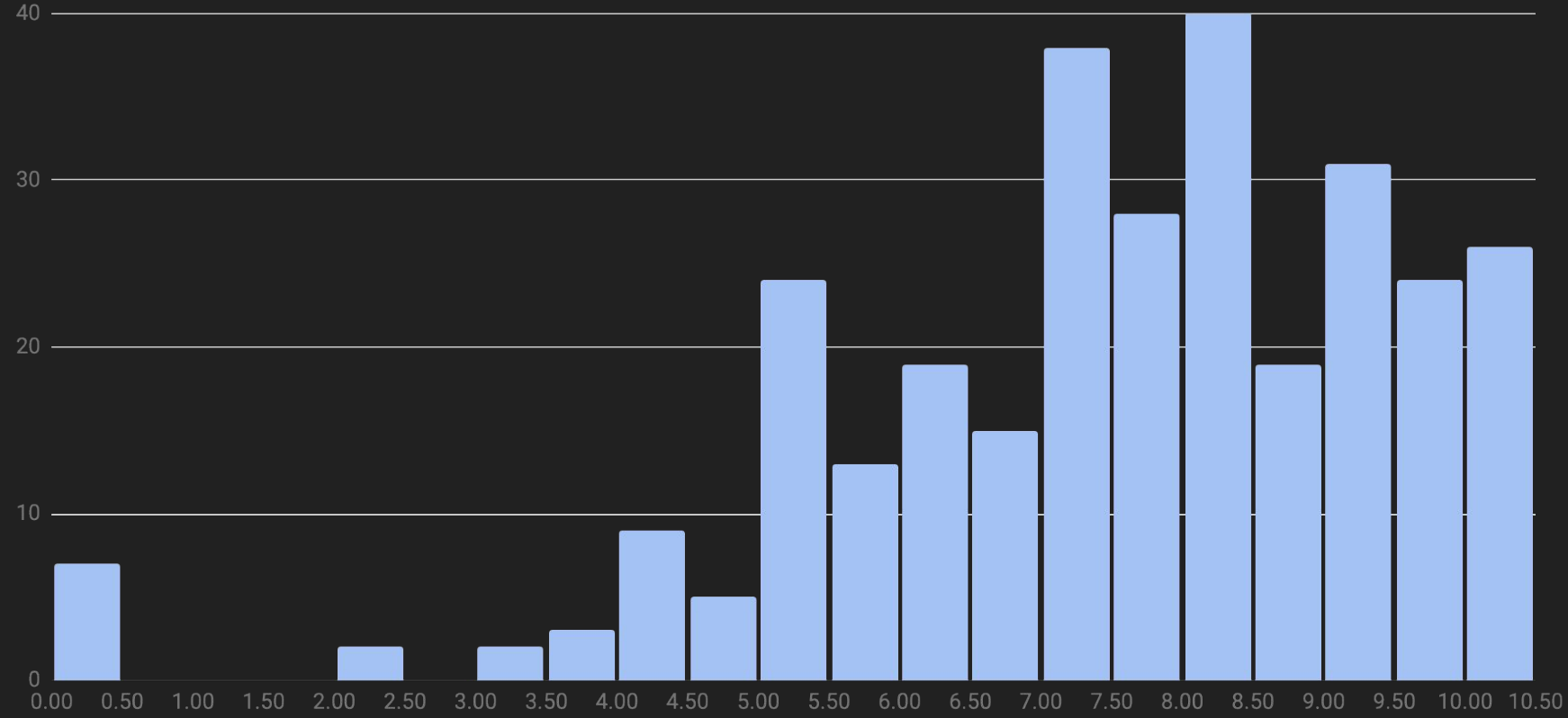
Links

- Introduction to Conda for (Data) Scientists
- Some links in the KSchool web
  - http://kschool.com/blog/data-science/martacamarads/
  - http://kschool.com/blog/big-data/jmvaldeolmillos/
  - http://kschool.com/blog/formacion/gonzalo-sanchez-tfm-data-science/
  - http://kschool.com/blog/data-science/jose-manuel-vera-data-scientist/
  - http://kschool.com/blog/data-science/tfm-data-science-describiendo-tendencias-busquedas-google-utilizando-tweets-relacionados/
  - http://kschool.com/blog/data-science/tfm-data-science-manuel-maestre-estimacion-precios-del-alquiler/
  - http://kschool.com/blog/data-science/tfm-data-science-banca/

Links to past TFMs

- All TFMs from previous editions, and some examples of very good ones:
  - https://github.com/huanlui/chord-suggester
  - https://github.com/SilviaMartinezQue sada/master-data-science
  - https://github.com/huanlui/chord-suggester
  - https://github.com/jonatancisneros/TransferLearningTFM
  - https://github.com/daniel-isidro/hot_n_pop_song_machine
  - https://github.com/pipe11/TFM_fake_news_detector
  - https://github.com/angelrps/MasterDataScience_FinalProject
  - https://github.com/AngelArcones/Data_Science_TFM
  - https://github.com/Aturt2/music-sheet-generator
  - https://github.com/nopaixx/kschool-tfm
  - https://github.com/diegoalvzfdez/master-data-science
  - https://github.com/macomino/TFM

TFM grades from previous editions



Questions?

- Can I repeat the same topic of a previous TFM?
  - Yes
- More questions :)