Data Science Workshop 2018/19: Project Guidelines

Project phases:

1. Choose a dataset(s) from either World Data Bank or Kaggle
2. Analyse the dataset to understand its nature and properties
3. Define a data science prediction problem
4. Sanitize and organise your dataset: Missing data, data integration.
5. Perform feature engineering and extraction.
6. Implement/Choose appropriate ML algorithms and methods over the data model to output results.
7. Evaluate your model statistically and use your results to improve the feature selection and parameter tuning.
8. Present the important and interesting parts of your workflow and results in a user friendly human-readable Jupyter Notebook/ R-notebook.
9. Derive interesting insights and/or applications from your analysis.

Project administrations:

1. Students must form teams of 3-4 members. It is the students responsibility to form the teams. Teams must send a registration email to the course staff by no later than 18/11/2018. Please include students names, emails, and IDs.
2. Teams must obtain a prior approval for the dataset and problem from the course staff, as well as to re-approve any changes in the data/problem that are made. If eligible, approvals will be granted at the first students presentation meeting. If not, teams must change their dataset and problem as instructed by the course staff, and re-obtain an approval by mail or in an office-hour appointment.
3. Teams will present their progress in class in the designated times appointed by the course staff. (15% of final grade)

Project Description:

1. The project must be a complete, end to end data science solution, including all phases described above.
2. As there are many exiting related solutions and notebook, specifically for the Kaggle datasets, teams are required to review existing solutions and state in what ways their project is differ.
3. As some of the data science workflows may be fairly simple, your advanced project must have at least one **point of focus**, namely a nontrivial issue that the team will handle. Students will be required to explain the particular difficulty, and to review related, practical/academic previous work. Examples are:
    1. Non trivial features extraction
    2. Complex problem setting such as time-series analysis
    3. Development of a complex ML model (i.e. a non-standard neural net, or implementation of a new model based on a recent academic paper
    4. Handling big data (in terms of volume or velocity)
    5. Performing weak-supervision
4. Data gathering is *out of scope*, i.e. no points will be granted for complex data gathering techniques such as web crawling etc., and it can NOT eligible be a point of focus.

Submission:

1. A link to the Github repository that includes:
    1. *all* of your code
    2. The final Jupyter/R Notebook, containing all phases of the project - from cleaning to evaluation. The notebook should be fully reproducible on a standard Linux machine with preinstalled Anaconda. All external libraries should be installed automatically so the notebook works instantly and seamlessly. The notebook should be comprehensive yet interesting to follow and read. It should contain the important and interesting aspects of each phase of the project.
    3. Your data (if it is too big, store it elsewhere yet make sure its accessible to the notebook when we reproduce it on our servers)
2. A documentation file (8 pages maximum, 1.5space, font size 12, PDF Only) describing your work.
    - Dataset description
    - Dataset analysis summary - characteristics and the nature of the datasets.
    - Problem formulation
    - Explicitly declare the point of focus of your project
    - Description of the solution w.r.t. each step in the data science process
    - Findings and statistical evaluation
    - Insights and applications
    - Related work - what else has been done in the practical (i.e. Kaggle competitions) or in the academic (papers) regarding your dataset and/or the specific problem you deal with.
    - Citations: Any external sources must be cited: That includes: Kaggle notebooks, previous solutions, papers, python libraries, code repositories (e.g. Github, Bitbucket) and significant forum posts that you relied on.

Important notes and tips:

1. Make sure you carefully read and address each of the points mentioned in this document
2. Note the weight of each aspect in the grading sheet and make sure you give it an appropriate consideration and effort.
3. In accordance, your submitted notebook and documentation file are of very high importance. They should be very well presented in order for you to get full marks.
4. Make sure that your problem of choice, as well as the solution are not too simple (Should be much more complex than the one shown in the hands-on class).
5. Consult with the course staff as much as possible and implement any feedback you get.