Final Project Instructions
STAT 471

Overview

The goal of the final project is for you to apply the tools you've learned in class to a real-world data mining problem.

Project format. You will find and/or assemble a dataset relevant to a domain that interests you (finance, business, sports, healthcare, etc.), and formulate one or more analysis goals that can be addressed based on the data. In many cases, you will need to clean and/or featurize the data before it is suitable for analysis. Once you have a clean dataset, you will carry out the data mining tasks we learned in class: data exploration followed by building, interpreting, and evaluating predictive models. Projects must include a predictive modeling component and can optionally also include statistically formal association analyses. After the data mining is done, you will make a set of conclusions and recommendations based on the analysis.

Deliverables. The main deliverable will be a final report presenting the problem, data, analysis, and conclusions in a way that would be clear to a technically literate stakeholder (e.g. your manager at a tech company). Unlike the homeworks, this report will not contain any code; its main goal is to communicate your analysis and your results. In addition to the final report, you will separately submit the underlying data and code. It is crucial that data analyses be *reproducible*, so that other data scientists can understand their details as well as replicate and extend them. More details on these deliverables are provided below.

Logistics. The final project can be completed in groups of up to three. Only one submission per group is necessary. Final project reports will be submitted through Gradescope, while data and code will be submitted as zipped folders on Canvas. **All deliverables are due on May 2 at 11:59pm. Final projects will be accepted up to three days late, with a 5 point deduction per late day. Students with unused late passes for homework can use these on the final project instead.** The teaching staff are happy to provide guidance on any aspect of your final projects, so feel free to reach out to us through email, Piazza, or office hours.

Recommended workflows

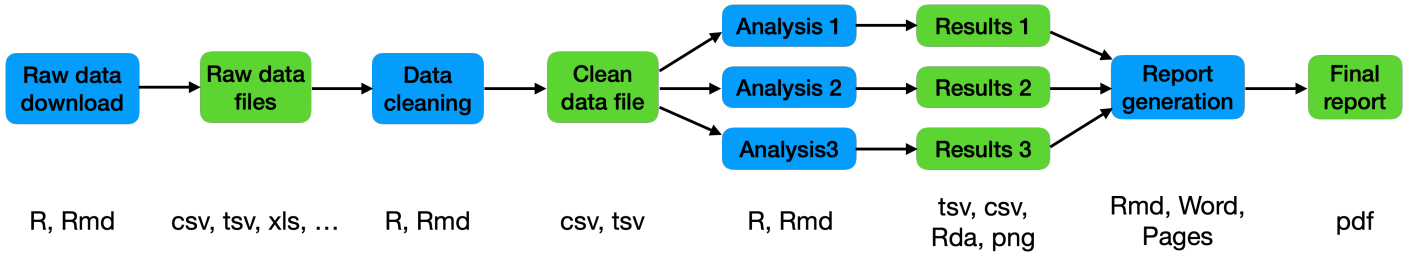
It is crucial to keep track of all data, code, and results in the process of a data science project. Depending on the project, it may be more convenient to accomplish this via an integrated workflow or a step-by-step workflow. Both workflows are described below. Regardless of the workflow, you should document exactly where the raw data came from. Ideally, you should have download links for each raw dataset, and R commands that download the raw data based on these links. This may not always be possible, in which case you should describe in as much detail as possible how you acquired the raw data.

Integrated workflow

An integrated workflow is one where all the data cleaning, analysis, and reporting is done in a single R Markdown file, like we have been doing for the homeworks. This workflow is convenient because it is familiar, easily reproducible, and there are fewer files to keep track of. The final report for an integrated workflow would be the compiled PDF (with echo = FALSE specified to omit code chunks) and the data/code deliverable would be the Rmd file together with any relevant data files. The drawback of this approach is that, for computationally heavier projects, recompiling the entire Rmd file can take too long. For such projects, it may be better to conduct the analysis using a series of scripts, each saving an intermediate output to be read in by the next. This is the step-by-step workflow.

Step-by-step workflow

In a step-by-step workflow, analysis tasks are modularized in their own R scripts or R Markdown files. The analysis scripts rely on each other's outputs, and the reporting is done in a separate R Markdown or regular word processor file. An example of such a workflow is given in the flowchart below: blue boxes indicate scripts and green boxes indicate output files. Analysis results can be stored in the form of R objects using the Rda format (e.g. a fit object from a random forest), tables using the tsv/csv format (e.g. a table of feature importance scores), or images using e.g. the png format (e.g. a CV plot). The final images and tables for the report can be produced by Rmd or by the analysis scripts; in the latter case even a usual word processor like Word or Pages can be used to put together the final report.



To keep track of all the files involved in a step-by-step workflow, it is recommended to set up a folder structure with separate folders for data, code, and results. It is recommended to keep raw data and processed data in separate subfolder; the raw data should be downloaded and then never modified.

Deliverables

Final report

While the final report must be submitted in PDF form, it can be generated from either R Markdown or from a word processing software. The final report must contain the following sections:

- **Executive summary.** A brief summary (one page or less) of the problem, data, analysis, and conclusions.
- **Introduction.** Background information, description of the analysis goals and why these questions are important to address in the context of the application.

- **Data description and exploration.**
  - Descriptions of where the data were obtained, how they were cleaned, how many observations there are, what information (features and response) they contain.
  - Allocation of data for training and testing.
  - Exploratory data analysis.
- **Model building, evaluation, and interpretation.** Description of the data mining procedures employed and their results. Any association analyses would go in this section as well.
- **Conclusions.** Overall conclusions and recommendations based on the analysis, comparison of method performance, takeaway messages for stakeholders, limitations of the current study and recommended follow-up analyses.

The final report should be submitted through Gradescope, with one submission per group.

Data and code

All data and code used to create the final report should zipped in a folder with an informative name (e.g. Katsevich-election-prediction.zip) and submitted to Canvas, with one submission per group. Submissions with several code, data, and/or results files (e.g. those based on the step-by-step workflow described above) should contain a README file describing the different files. The teaching staff should be able to reproduce your final project report based on the data and code provided.