INFO 348 – Fall 2025

Final Project

Project due Monday, December 15th at 11:59pm

Bringing It All Together

For the final project, you will be conducting exploratory data analysis and writing up your findings. The goal for the final project is to give you an opportunity to apply some of the techniques we've covered over the course of the semester while asking and answering your own questions from the data. You may work individually, or collaborate with another student on your project.

The project is intended to be open-ended, in that it's up to you to decide where to focus your efforts. Your analysis should apply several of the techniques we studied over the course of the semester, with the goal of answering questions with your data. For instance, you could geocoding to investigate geospatial patterns. Alternatively, you could build a model to classify your data, or compare the results of multiple classifiers. The goal is to find a question (or set of questions) about the data that interest you, and attempt to answer them through your analysis.

Task Requirements

For full credit, you must complete a minimum number of the items from the following list of data and analysis tasks. If you're working alone, you must complete at least *three*; if you're working with a partner, you must complete at least *five.*

Furthermore, the tasks fall into two categories: *data tasks* and *analysis tasks*. Single-person projects must include at least one analysis task, while two-person projects must complete at least two. The tasks for each category are:

Data Tasks

- *Data Wrangling and Assembly*
  Create a unique data set by transforming and combining multiple existing sources of data into a cohesive whole. Must involve a non-trivial amount of code to manipulate data into something usable for analysis.

- *API Access*
  Construct a data set using an API. To receive full credit for this task, the data manipulation must be nontrivial: you must make calls to multiple different API endpoints and combine the results together into a cohesive data set (simply using a single API call to pull down a CSV is not enough).

- *Database Storage and Access*
  Programmatically store and query your data using a relational database. While we focused on reading from RDBMSs, you may elect to set up a SQLite database using `CREATE TABLE` statements and loading in your data. Once your data is loaded, you can demonstrate the use of queries to answer questions, summarize, or understand the data set.

Analysis Tasks

- *Statistical Summarization*
  Summarize and explain particular aspects of your data using summary statistics or other calculations. Simply reporting on means or modes is not enough, you must instead seek to identify and quantify interesting findings in your data set as a whole or particular subpopulations.

- *Association Analysis*
  Identify statistical associations found in your data, showing quantitative and visual proof of the relationships between variables. Discuss the meanings of these associations and how they give insight to your data.

- *Predictive Modeling*
  Build a classifier (or regressor) to predict some feature of your data using scikit-learn. Include a performance analysis, along with any explanation of modeling choices you made (e.g., model types, parameter values for the models, attributes included or excluded).

- *Visualization*
  Create a series of plots that bring some aspect of your data set into specific relief. The types of plots you use should be determined by what you want to convey, and what questions you hope to answer.

- *Geocoding*
  Perform some spatial analysis using a geocoder to calculate locations and distances between data instances or other points of interest. You might also use location data to plot on a map.

- *Clustering*
  Use the unsupervised learning functionality from scikit-learn to perform clustering on your data set. Be sure to explain what types of groupings you're hoping to capture, along with sample outputs and descriptions.

- *Natural Language Processing*
  Use NLTK to analyze unstructured text found in your data, by identifying interesting

language characteristics, learning predictive models, etc.

- *Other, with permission*
  Come up with your own analysis task. Just make sure you include it in your project proposal to get approval.

Data Sets

Your analyses should be based on one of the data sets from the list here:
https://bit.ly/348_data_f25

You are free (and encouraged) to augment that data with other publicly available data if useful. You can also use your own data set *with prior approval*. In general, the data must meet following requirements:

- Data must be public - nothing proprietary, illegally downloaded, etc.
- Data must be rich - to enable interesting analysis, data sets should comprise hundreds (if not thousands) of examples, with multiple associated attributes, entity types, etc.
- Data must be novel - basically, if it's been thoroughly analyzed and publicly reported on the internet, it will be disallowed (therefore, any data sets that are the subject of Kaggle competitions, or have been used as online case studies, etc. will be disallowed)

When picking a data set, you should seek to find an application area that interests you. Successful projects will answer questions about the data and the things it represents, and good questions are most often inspired by personal interest and curiosity.

Project Writeup

Your project submission should include a 2-4 page written document (3-6 pages if working with a partner) describing your work. How you structure the writeup is up to you, but in general you should include:

- The names of all team members, along with a brief overview of how each person contributed
- The goals of your project – what did you hope to learn from your analysis?
- A description of the data set, including any preprocessing you did to get the data into a usable format
- A short writeup for each task completed, summarizing the techniques you used, as well as any conclusions you were able to draw
- Description of challenges you encountered when working with the data, and how you were able to overcome them (or not!)

- Descriptions of any insights into the data or domain that you obtained through your work
- Ideas for future exploration of the data, including interesting questions raised by your analysis

Imagine that the audience for the writeup is a 348 student like yourself, and you are describing what you've done so they can continue the work. While not graded for style, writeups should be clear, well organized, cohesive, and readable (writeups with nothing but short bullet points will not be given full credit).

Project Code

All code used to import, manipulate, and analyze your data should be included in a single Jupyter notebook. Code should be reasonably clean, commented, and fully functioning – while your code doesn't have to be pristine, you should include comments that help a reader to understand how things work. Organize the cells in the notebook so they can be run sequentially to replicate your analysis results.

Et Cetera

- The level of effort required for your project should be equivalent to 2-3x a normal homework assignment. Additionally, if you're working with a partner, the scope of your work and writeup is expected to be proportionally more substantial than if you're working by yourself.
- You are encouraged to leverage existing tools (pandas, scikit-learn, etc.) wherever possible. You are free to base your code off of any of the examples covered in class or homework assignments.
- The use of large language models (e.g. ChatGPT, Copilot) for your code or writeup is prohibited, as is replicating previous analysis found elsewhere.
- Note that "negative results" are totally acceptable for this assignment (for example, "here's several things we tried along with a theory of why none of them worked").
- Your project will be graded holistically, taking into account effort, creativity, degree of difficulty, technical proficiency, quality of the writeup, etc.

What to Submit

You should submit single zip file containing:

- A single pdf containing your writeup named `<lastname>_writeup.pdf` or `<lastname1>_<lastname2>_writeup.pdf` if working with a partner (for example, `rattigan_scrooge.pdf` if I was working with my friend Ebenezer).
- A Jupyter notebook called `<lastname>_code.ipynb` containing the code used to generate results (`rattigan_scrooge.ipynb` for the example above)
- Data files used in the project (if they are > 5MB in size, you can include a text file called data.txt containing URLs linking to the data sets)