CSE 519 -- Data Science (Fall 2019)
Prof. Steven Skiena
Semester Projects

Proposal Due: Thursday, October 24, 2019
Progress Report Due: Thursday, November 14, 2019
Poster/Final Report Due: Thursday, December 5, 2019

The project will involve concentrated work in one research project related to data science. Certain aspects of the project will have to change to support the size of the class, but we are structuring the projects and grading in such a way as to provide meaningful feedback for large numbers of students.

The possible choices of projects will be constrained to those listed below, so there will be several groups working independently on each project. You will not be allowed to propose your own project topic, but my topics leave there is enough room to pursue distinct directions that I expect to see variety among the submissions. Each project will have a TA or other graduate student serving as "captain" for the project. Your captain is the first line of defense for individual discussions about your team's particular project. I will also be encouraging the use of Piazza for questions/discussions about your project, as well as occasional "town halls" about each project in or immediately after class.

While I anticipate that much of the work will be done as the deadline approaches, it is important to get started early enough to discover insurmountable roadblocks in data acquisition or problem definition before it is too late. The project proposal and progress reports have been instituted to ensure people get serious well in advance of the final deadline.

Each group will be responsible to turn in 3-5 page project proposals/literature search and progress reports as of the dates above. I will award almost half of the total grade for each project on the strength of the preliminary reports. This is to encourage starting early, and to make sure that you and I both know what you are to do before it is too late to avoid trouble.

My hope is that best of the submissions for several of these topics will lead to published work. This has been the case several times when I have taught such a course.

Rules of the Game

- From past experience, I have found that students tend to herd on a small subset of possible projects. Further that they are irresistibly drawn to poor choices: that offer the least available data, that have limited ways to evaluate success, or little freedom for creativity in modeling. Please consider projects which seem less popular -- likely they will prove better to work on.

- Take the time to investigate multiple options during the proposal phase. You should have gotten some preliminary results by then, enough to provide confidence you will be able to successfully complete the project.
- I reserve the right to reassign the groups with the weakest proposals to other topics for the second phase of the project, to rescue them from a bad situation and rebalance the class.
- A substantial part of the grading of the projects will be done by peers: each student will be charged with grading three other projects on the same topic. Part of your grade for the course will be a function of how seriously you take this, e.g the quality of your feedback. Peer review is how research papers are selected for publication, and I hope this will be revealing. I am hoping that this review will provide insights to improve your own projects as well.
- To preserve anonymity in submissions, it is important that the papers you submit online for grading **not** contain your names or ID numbers. **Indeed, the peer grading form/rubric will include a question if the grader can figure out whose paper they have, and if so we will take off points.**
- Expect to be asked to do a trial peer grading exercise to test our system soon. Please respond promptly.
- Group sizes should range from two to four students. The amount of work per group is expected to scale linearly with the effort involved. The projects are large enough that I do not think students working by themselves can achieve the critical mass to get something substantial done, and are strongly discouraged.
- Each student will be asked to proportion their group's effort among all the team members they are working with. So make sure you work with people who you like, trust, and respect.

I will quite possibly add another few projects later, so check back occasionally.

How much do people sleep?
Captain: **Shihao Zhou** **<shihao.zhou@stonybrook.edu>**

Social media analysis sheds considerable light on human behavior, gaining statistical strength from the scale of such interactions. In this project, we will analyze Twitter data to get insight into factors affecting how much sleep different populations receive.

Prof. Jason Jones of Sociology has used Twitter for studying sleep patterns, most notably his study of the performance of NBA basketball players after unusually late nights, as measured by late night tweets. A good way to start this project is to read his papers on this work, particularly

JJ Jones, GW Kirschen, S Kancharla, L Hale
Sleep health 5 (1), 68-71

Association between late-night tweeting and next-day game performance among professional basketball players

There are many interesting ways to break down this data. I am much more interested in projects which do a sophisticated and careful (statistically defensible) analysis of one of these questions than a hack programming job that does a lousy job tabulating all of them. Possibilities include:

- *Sunlight and seasonal effects* -- The time of sunrise varies with latitude and season. A reasonable hypothesis is that sunrise affects the time people awaken in the morning. Can you produce a satisfying sleep map video showing wakeup times by place for each (say) week of the year? Are bedtimes in synch with this, or do people get less sleep in summer and more sleep in winter? Are their periodic changes in the temperate regions, which have less fluctuation in sunrise/set times than more polar regions?
- *Weekend and holiday effects* -- How do holidays (where people are less likely to travel to work) change people's sleep schedules, and how long does it take for them to recover?
- *Time zone effects* -- Because time zones are broad, there is more light at 7AM on the eastern side of a time zone than the west. All presumably work the same conventional hours (traditionally 9AM-5PM). A reasonable hypothesis is that people on the east end of a time zone wake up earlier than the west -- is this true?
- *Daylight savings time effects* -- How do sleep times adjust to annual one-hour time changes in fall and spring? Do people gain/lose an hour of sleep, or does it come from other activities? How long does the change in sleep times last?
- *Job title effects* (*) -- The self-reported description string for each Twitter account gives insight into how people see themselves. Often it contains information as to career and interests. Are there differences in the sleep patterns of truckers and teachers? Prof. Jones has done some work on these strings, but the first task is to classify them into groups.
- *Age, gender and nationality effects* -- Do sleep schedules vary significantly among people of different demographic groups? Be careful: these effects must be separated from sampling and geographic biases.
- *Time course effects* -- As the years go by, are people getting more or less sleep? Be careful: these effects must be decoupled from changes in the popularity of Twitter and all the other factors discussed above.
- *Travel effects* -- Geolocation tags and self-reported tweets should identify people who had cross-country travels. How did this affect their sleep patterns, and for how long?

Large scale Twitter data sets are available on the IACS Seawulf cluster.

- Prof Jones has been collecting a 1% sample of all tweets from February 2015 through to the present. These tweets are stored as text files in JSON format. Each hour, the collector writes about 100,000 tweets to disk and the hour files are about 100 MB compressed.

- The files contain Tweet objects in JSON format. There is good documentation for these objects available through the Twitter API documentation. https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object.html
- Each Tweet object contains a User object. User objects contain information about the author, including their preferred display time zone, their self-description and a free-response location field.

One days worth of Prof. Jones' hourly data are available at:

https://drive.google.com/file/d/1A2p8V9O2HKR6KrzJME3ZlNZQyqILFsIp/view?usp=sharing
https://drive.google.com/file/d/1bYH3ooiAKZbg-fiZ8FHHYNiCSA4dYSBb/view?usp=sharing
https://drive.google.com/file/d/1yiotD05mRrZvkFPo8teG4TPuhws_qUVo/view?usp=sharing
https://drive.google.com/file/d/1ITWmOl-i5n8rk-MzNM8lm59nG-mbjD3T/view?usp=sharing
https://drive.google.com/file/d/1eOZlS7PO-ywqMcRptVS-FFYfQhCkTPiN/view?usp=sharing
https://drive.google.com/file/d/17XAk7_FNFlEo14YhYmD2_6SNB_y4MTfZ/view?usp=sharing
https://drive.google.com/file/d/1zo3lcTd2V6Ttt-ylJH4h-iMpZVd_CzF5/view?usp=sharing
https://drive.google.com/file/d/1UtGsytOW7vW0wPwt1cGmaPnCLQ_KMlvS/view?usp=sharing
https://drive.google.com/file/d/1Mbl-BJ1YFhpEJc2Zeefd-wsWQnzd7989/view?usp=sharing
https://drive.google.com/file/d/1LNk2L8blALDILpeexux557p6T2d876XE/view?usp=sharing
https://drive.google.com/file/d/1Sjrq2KEQTDFfVrvDTp_P1oipyULY1y3Q/view?usp=sharing
https://drive.google.com/file/d/1RLmu4JqumjdzbPz18xfmZHiKJu3lx8lF/view?usp=sharing
https://drive.google.com/file/d/10SSnH8Coq_x6YnIprzbpZPrrEKKSWVZI/view?usp=sharing
https://drive.google.com/file/d/1U-_NRGp5Uls9LJumTnIlTpQtAfY2oyR0/view?usp=sharing
https://drive.google.com/file/d/19ycs2tO9C74FeNA-wU2bMzwjgaFO6pQu/view?usp=sharing
https://drive.google.com/file/d/1-awGVaqo9Vr8lenIcowf-F9oZwUcA824/view?usp=sharing
https://drive.google.com/file/d/1xygxTkZZLQ3QBKll9sYvWDdlwHw1xjAy/view?usp=sharing
https://drive.google.com/file/d/10FTVg4E2hBj7KSLotNerKjD4uL3amkGT/view?usp=sharing
https://drive.google.com/file/d/1In4nUiPMiu4r_ttRv1Fbblr0DBptyP6E/view?usp=sharing
https://drive.google.com/file/d/1GhoTwSI30YflSAE85OCMM1SBXQqZd7-7/view?usp=sharing
https://drive.google.com/file/d/1PfJ0sB0dMFmcna6snDmVtn3A-t-TaPaZ/view?usp=sharing
https://drive.google.com/file/d/1nmrvS1u5e37n_qHb7OSfzunNuDMgVvsE/view?usp=sharing
https://drive.google.com/file/d/1ViJ1yICz2viKiFAbQf8ioHyhQSOublEw/view?usp=sharing
https://drive.google.com/file/d/1AQCgcvwSdlGO6cI35FcmYvlFXfEcZgQV/view?usp=sharing

Twitter data might also be obtainable by your own spidering/download efforts. For your final analysis, we can pursue access to the IACS cluster for serious teams.

There are many effective ways to mess up your analysis. Some of these include:

- *Bots and corporate sources* -- Mechanical sources do not sleep according to a human schedule. Companies are most active during the business day regardless of location or season. One must try to identify and eliminate such sources.

- *Sampling effects* -- You are only seeing 1% of all posted tweets, which means you are unlikely to see the first and last tweet of the day for people. An important orthogonal analysis might be to analyze the full profile of a smaller number of people, with limitations imposed by Twitter APIs.
- *Statistical sharpness* - You need the right statistic to pick up the subtle phenomenon of first and last tweet of the day. Perhaps it is over or under representation in each time window over a baseline, but you need to do some clear thinking to get something meaningful.

How Good is a Chess Player?
Captain: Allen Kim

In games such as chess, it is often not too difficult to tell whether one is playing a complete novice or a seasoned player. This project concerns predicting the chess rating score (ELO) of both players in a game, given a record of the match. Chess makes for an interesting and appropriate data science challenge, for there are records of millions of games available with ratings of both players. Can one write a model to predict the ratings of both players given the moves that they make?

Lichess (https://lichess.org/) is a website (started 2013) called Lichess that is completely free (no advertisements and running purely off of donations) and open-source with all of their game data publicly available. The database of all games is available at https://database.lichess.org/ There are now over 800,000,000 games in the database, each tagged with ratings of both players and the speed (blitz, rapid, classical).

To get a sense of how many games there are at every level, we can examine their opening analysis board that sorts by rating: https://lichess.org/analysis. They group by the following ELOs: 1600, 1800, 2000, 2200, 2500. There are over 15 million games for 1600, more than 30 million for 1800 and 2000, more than 12 million for 2200, and more than 750,000 for 2500. This only accounts for less than 100 million games. Given that there are over 800,000,000 games, there will be a diverse set of games from lower rated players as well as well as grandmasters.

Specific challenges here include:

- *Rating prediction from game transcript* -- Given the flow of play in a particular game, how accurately can you predict the ELO rating of the players? A good first task to try to distinguish beginner games (say below 1500) from masters (say above 2000) before seeking finer distinctions.

- *Game type prediction from transcript* -- Classic, Rapid, and Blitz chess are three variants which differ in the amount of time they allow each player. Can you guess the time limit on a given game as a function of the observed play?
- *Subtranscript-level analysis* -- Can you tell the quality of play from just the final board position, instead of the full record of moves on each side? What does clock usage time tell about the level of play?
- *Computer vs. human?* -- can you identify which player is human and which machine in a given match? Are certain computer programs easier to identify than other ones?
- *Characterizing upsets in chess* -- Strong players sometimes lose to weaker ones. Can one identify games where the weaker player wins?

Do some preliminary exploration with this data before you start trying to build a model. Initial experiments should definitely be done before submitting the proposal.

Political Polarization and Marriage

Captain: **Shihao Zhou** <shihao.zhou@stonybrook.edu>

Voter registration data records contain names, addresses, dates of births, and party affiliations. Although the regulations differ from state to state, many states (like Florida, apparently archived here) make their voter registration data part of the public record. Prof. Jason J. Jones of Sociology has been obtaining longitudinal voter registration data from several states for research purposes.

That there has been increasing political polarization in the United States has been well documented. The impact of this in the social sphere has been well documented. For example, Thanksgiving dinners in 2016 were apparently shortened to minimize unpleasant political discussions. We conjecture that similar pressures will take their toll on people who have different political beliefs than their spouses.

Using voter registration data, it should be possible to identify cohabitating couples (two people at the same address that are likely not brother/sister), and track them over each electoral snapshot to see whether they are still cohabitating. In particular, did couples with alternate party affiliations (one Democrat and one Republican) cease cohabitating at a greater than expected rate after Trump was elected in 2016?

Prof. Jones has collected snapshots of all New York State voter registrations from 2012, 2016, 2017 and 2019. Each snapshot is a csv file which contains one record for each registered voter. In 2019, there were about 12 million registered voters in New York State. Multiple years of historical data are also available for the states of Ohio (about 8 million registrations) and Washington (about 4 million). Prof. Jones can share these files (each file is a few GBs) through Google Drive. For teams seeking more records or cross-state validation of their methods, 2012

records are available for California, Florida, Kansas, Missouri and Oklahoma; the teams could request current records from these states.

Note: there is legitimate sensitivity concerning analysis of data about real people. Under no conditions post or distribute data including the names and identities of individuals -- only aggregate level analysis, and of course do not communicate with or pry on (e.g. Google) any individuals you find.

Which Movies Endure and Why?

Captain: Yunxiang Wan <yunxiang.wan@stonybrook.edu>

Any given old movie will have had a certain popularity at its time of release, perhaps measured by its box office gross. To what extent is a film's contemporary (current) popularity a function of variables associated with its production (popularity, production budget, genre, awards, etc.) as opposed to subsequent historical factors (business release/distribution decisions, subsequent actor careers, technological factors like black-and-white to color transitions, etc.).

The Internet Movie Database (IMDB) (IMDB) is a rich resource providing data on essentially every motion picture ever released, including cast, genre, budget and other financial data. Ratings (both the number of people who have rated and their average score) provide a meaningful measure of current popularity. Streaming data may also be revealing. Box office gross (meaningfully adjusted to reflect year of release) provides a measure of its original popularity.

We need to develop an accurate model to predict the current popularity of a film as a function of variables known at its time of release. We need this to work over time scales since the beginnings of the sound picture era (1927) and ideally even before that.

After this, you have a tool which can quantify each film for how much more/less popular it is than it "should be". You must learn enough about popular genres of different time periods (each decade from the 1930's through the present day to be able to perform a sniff test of whether you are capturing real phenomena. Read some histories, and identify what the major films from each period are. **Last year's teams on a related task concerning popular music did a miserable job on this project because they refused to do sniff tests of their computational results**. As an example of films whose fate has diverged with time, consider *It's a Wonderful Life* and *Casablanca*.

After you have an effective model to identify over and underperforming movies, there are a variety of issues to explore to try to explain why, including:

- For each time period report the 10 films with the dramatically highest/lowest performance. Look for properties that explain what you see -- make hypothesis, do statistical tests, and draw conclusions.
- For star actors powering many successful films, what is the trend over their career. Do their early/late films tend to over/under perform? Do films without important stars retain more/less popularity than expected?
- Are their time periods or technical changes which mark abrupt shifts in the durability of particular films? Was there a transition on sound-to-silent, or black/white to color? What about Imax/3D?
- Consider the effects of genre: how have different genres performed? Can you demonstrate that war movies and westerns have gone out of style? What about musicals?
- How much do Academy Awards (Oscars) make? My guess is that Best Picture and Best Actor winning films endure better than Oscar nominated films which did not win.
- Consider other data sources, like song lyrics and the Million song data. Does a successful movie theme song explain the endurance of certain films? Can you make better explanations of why certain films endured or failed?

There is room for similar studies of other forms of entertainment media, like best selling books and Bollywood films/music -- but I need you to develop your models first for American movies before moving on.

Retail Sales Data Analysis
Captain: Anagh Agrawal <aaagrawal@cs.stonybrook.edu>

A small, local chain of retail stores has graciously made its market basket data available to us for analysis. This data consists of records of several million sales, over several years, of tens of thousands of different products. The primary goal of this project is to provide the company with insights that will help them with their business, like (1) which products to promote, (2) which products to place near other products, (3) which products to stock, or (4) which to recommend to specific customers.

This is a rich dataset, which offers many directions for possible investigation. A tiny example of the file is available here. Fields include:

Date,Transaction Time,Customer Number,Sales Header,STR,Item Number,Item Description,Net Sales Units,Net Sales,Cost,Gross Margin,Gross Margin %,Class Code,Class Name,Department Code,Department Name

Teams are encouraged to think creatively about what directions to pursue, although some ideas are given below:

- *Product embeddings* -- Build distributed representations of products (embeddings), perhaps using category data or the network of co-purchased items (or both). Do these embeddings provide interesting insights into which products are similar or related?
- *Promotional recommendations* -- Can you identify products to push to a given purchaser given the basket of items they bought?
- *Time series modeling of sales* -- Can you predict how much you will you sell of a given item what when/where?
- *Weather and Holidays* -- How does weather and holidays affect sales. Which items are differentially bought then, and why?
- *Location Analysis* -- Which stores have differential sales by location in interesting ways?

This data is not for public distribution. Any student who takes on this project will have to sign an NDA, and pledge to keep the data private.

All student reports will be made available to the company for review, which may suggest specific directions for future work.

Monopoly Playing Programs
Captain: **Sanjay Mathew Thomas** <sanjay.mathewthomas@stonybrook.edu>

There have been widely-publicized recent successes in computer programs for playing chess, go, and poker, building on advances in machine learning and game theory. The board game Monopoly is generally not considered a game of sophisticated strategy, but I believe that is largely because people do not play enough games against each other to really establish the difference between players. But computer programs can play hundreds of games against each other to fairly establish superiority. Can we produce a world champion Monopoly program, as per http://monopoly-championship-history.wikia.com/?

In last year's class, Sanjay Mathew Thomas developed an adjudicator program available at http://www.monopoly-ai.com/. This defines APIs so that two or more teams building adjudication programs can play against each other to establish which is better. This supports the Official Rules of Monopoly and defines a standard for proposing trades with other teams. **You are to use this adjudicator, and evaluate against other programs.** I expect that this year's programs will be substantially stronger than last year's entrants.

Your efforts should start by building a baseline player (say always buys anything they can afford) and evaluate more sophisticated models against it in statistically significant tests. I expect that the best programs will take advantage of some of the following ideas:

- Simulation: what happens how likely at what point in the game.
- Scoring functions: how do we evaluate the value of a property or trade.
- Reinforcement learning: can we play two programs against each other and identify which were good/bad decisions to train better models
- Lookahead: is now a good time to do something based on other player's positions?

Your grades will reflect effort, discipline, and creativity, not just the success of your program in the tournament.

Evaluating Dimensionality Reduction Methods

Captain: Yunxiang Wan <yunxiang.wan@stonybrook.edu>

Dimensionality reduction is an important operation with large datasets. But what is the "best" dimensionality reduction method? In this project, you will get familiar with popular methods, find different kinds of datasets, and compare these methods on them. The goal is to design novel qualitative and quantitative methods to compare and evaluate different dimensionality reduction algorithms. Use your designed methods to find the "best" dimensionality reduction algorithm(s). Note that we do not want to see just a collection of plots of m algorithms running on n datasets.

Dimensionality reduction is important to data science because we often encounter high-dimensional data. These methods help us in two major ways:

- Many machine learning algorithms do not work well on high dimensional data. For example, clustering algorithm HDBSCAN often fail to cluster large number of points on high dimensional data. Performing dimensionality reduction on your data before feeding them to your machine learning algorithm may help on improving accuracy and running time.
- Dimensionality reduction into 2 or 3 dimensional helps visualization. With the help of it in, we can visually understand the performance of, for example, classification algorithms.

This project was inspired by this *Nature Biotechnology* paper, "Dimensionality reduction for visualizing single-cell data using UMAP" (https://www.nature.com/articles/nbt.4314) which compares UMAP with other state-of-the-art dimensionality algorithms in many qualitative and quantitative ways.

Some possible directions and suggestions for this project:

- Try many different algorithms, pick 4, 5 “best” ones to include in your comparisons and select the top 2 for very detailed comparison.
- Dataset is very important. Use dataset that can differentiate the capability of different algorithms. Try many and pick 2 or 3 in your report.
- You should understand the algorithm and the math of methods you pick. You should give your reasonings for your algorithms’ performance in each task.
- Spend thoughts on your plots. Not all of them need to be bar charts. Also do not just show the 2D reduction images. You should be able to tell a story with your plots.
- Only you know your dataset, so you need to explain clearly what each plot shows.
- Many algorithms do not work well on high dimensional data. Compare the improved performance after using dimensionality reduction.
- You can qualitatively analyze and compare 2 different algorithms by showing the 2D reduction results side by side. What information is lost from reduction?
- Design metrics that give quantitatively results of algorithms’ performance.
- If your group does not have access to good computing resources, try to find small yet interesting dataset and start with fast algorithms.

A few datasets to start with include:

- The MNIST DATABASE: http://yann.lecun.com/exdb/mnist/
- Fashion MNIST: https://github.com/zalandoresearch/fashion-mnist
- COIL-20: http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php

Dimensionality reduction algorithms and packages to start with include:

- Multidimensional scaling (MDS):
- https://scikit-learn.org/stable/modules/generated/sklearn.manifold.MDS.html#sklearn.manifold.
- t-distributed Stochastic Neighbor Embedding (t-SNE): https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html
- Autoencoder: https://github.com/L1aoXingyu/pytorch-beginner/tree/master/08-AutoEncoder
- UMAP https://github.com/lmcinnes/umap