List of Project Topics

The following is a list of suggested project topics for you to choose. The list is currently being updated. Please check back again as new topics will be added.

A. Urban Crime Rate Prediction (Type = group project)

Crime prediction is an important application of big data for law enforcement especially in large cities. For this project, your goal is to develop an effective predictive modeling framework using diverse types of data available (e.g., historical data, weather data, traffic data, demographic data, etc). Specifically, given a location (latitude/longitude or a county/zipcode), time of day (say, late evening), and day of week (say, weekday), predict the likelihood that a crime will occur. If successful, the prediction model can then be applied to all locations in your study region for a future time period. You can display the predicted crime rate on a map for visualization purposes. The specific tasks to be performed include:

1. Collect the crime data. Due to the open government initiative, the many major cities (e.g., Chicago and New York City) have provided APIs to download the historical crime data. Most of the data sets have the following information: Location (latitude/longitude), Time, and Crime type. You need to be careful when processing the data to check whether the location field indicates the actual location of the crime or the precinct where the crime is occurring. Similarly, does the timestamp represent the time when the crime occurs or when the police report is lodged?
2. Collect other auxiliary data. This could be location information (e.g., residential, business, or entertainment district), demographic information (e.g., population density, poverty level, etc), weather data, social media data, or any information that can help improve the prediction model.
3. Data preprocessing. One of the first tasks is to partition the study region into smaller subregions (e.g., districts/counties/zipcodes) so that the prediction can be performed for each subregion rather than for a specific location (latitude/longitude). Next, compute the crime rate as well as other auxiliary features for each subregion at each time period. The preprocessed data should have the following format:

   ```
   (subregionID, timePeriod, timeOfDay, dayOfWeek, crimeRate, auxFeature1, auxFeature2,  ...)
   ```

   where auxFeature1, auxFeature2, etc., are the attributes/features computed from the auxiliary data. CrimeRate is a ratio between the number of crimes that occur in the subRegion at the given time period to, say, the total number of crimes that occur on the given day for the entire study region. An example of the data may look like this:

   ```
   region7,01-15-2018:08:14:20, morning, weekday, 0.0005, weather=snowing, traffic=high, â€¦)
   region7,01-15-2018:14:21:30, afternoon, weekday, 0.010, weather=clear, traffic=low,â€¦)
   region8,01-20-2018:20:48:15, evening, weekend, 0.025, weather=clear, traffic=low, â€¦)
   ```

4. Model building and evaluation. You will need to divide the data into training, validation, and test sets. You can use any regression technique to train the model. In addition to calculating the prediction accuracy of your model on the test data, it would be useful to also identify which feature is important for making the prediction.
5. Model deployment (for groups with 3 students). Deploy the model of the entire study region to predict location of crime for a future time period. You can build a web-based interface or a smart phone app to display a map that shows the crime rate prediction generated by your model for certain time period over the entire study region.

References:

- Crime Rate Inference with Big Data, In Proceedings of ACM SIGKDD International Conference on Data Mining (2016).
- Automatic Crime Prediction using Events Extracted from Twitter Posts, Proc of 5th International Conference on Social Compyting, Behavioral-Cultural Modeling and Prediction (2012).

B. Flight Delay Prediction (Type = individual)

This is an important big data problem for commercial aviation as delays can disrupt airports and cause significant stress and discomfort to the passengers. The goal of this project is to develop a predictive modeling framework to predict whether a flight will be delayed, cancelled, or diverted using data from the US Department of Transportation. Specifically, given a flight number from an origin airport to a destination airport on a given day, make a binary prediction whether the flight will be delayed or not. You can also convert this into a multi-class problem by predicting the reason for the flight delay (no delay, delay due to weather, delay due to mechanical failure, etc). The specific tasks to be performed in this project are:

1. Data collection. You can use flight data from the Kaggle machine learning competition (available from here) for the original source data. However, you must complement the data with additional data sources such as passenger traffic and cargo data, weather data, or others.
2. Data preprocessing. One of the key preprocessing tasks will be to merge the data from different sources. As an example, for the passenger traffic and cargo data, you can use the data to determine whether an airport is large, medium, or small by discretizing the number of passengers or cargos who use the airport (you should

also try to use more than 3 discrete intervals). You may also use clustering to do the discretization. For weather data, you need to determine the weather condition at the departure and destination locations of each flight on the given day of the flight departure.
3. Data exploration. For each departure airport, calculate its percentage of flight delays/cancellations and average delay time. Display the result on a map (which shows which airport has the most frequent cancellations/delays and longest average delay in its departure time).
4. Model building and evaluation. You will need to divide the data into training, validation, and test sets. You can apply any classification technique to train the model. In addition to calculating the prediction accuracy of your model on the test data, it would be useful to also identify which feature is important for making the prediction.

References:

- A Review on Flight Delay Prediction (2017).
- A Comparative Analysis of Models for Predicting Delays in Air Traffic Networks (2017)

C. Sentiment Analysis of Twitter data for Predictive Modeling Application (Type = Individual)

Social media platforms such as Twitter have become a popular data source for many big data applications, from event monitoring (e.g., detecting earthquakes) to forecasting applications (e.g., predicting disease outbreak). The goal of this project is to leverage the social media data for a predictive modeling application such as stock market prediction. You can choose any application you want but Twitter data must be one of the sources used. The following example illustrates the scope of such project for the stock market prediction problem. If you would like to apply the twitter data for other applications, please consult with the instructor first. Given a collection of real-time tweets, the goal is to provide a price forecast for a particular stock based on users sentiment expressed in their tweets. The specific tasks to be performed in this project include:

- Data collection. Identify a set of target stocks you are interested in predicting and download their historical data from Yahoo finance website. You will also be provided with 1 year of Twitter geotagged data (the size of the raw data is more than 1 TB) and is available from the CSE servers (e.g., arctic or black). Per Twitter data policy, make sure you do not store the raw Twitter data on GitHub or any other publicly available sites. The dataset should be used only for the class project. The data provided contains tweets with geolocation in United States collected from April 2015 to April 2016. You will need to identify a set of keywords that can be used to find the tweets related to each stock. You should store only the processed data (i.e., ignore all the tweets do not fit your keyword search criteria) to reduce space requirement.
- Data preprocessing. For each tweet, you need to perform sentiment analysis on the tweet to determine its polarity (say, positive, negative, or neutral). See, for example, OpinionFinder. You can also build your own sentiment analysis software using a list of publicly available positive and negative sentiment words (see for example, here or here) or train your own classifier to do so. You will also need to compute the daily changes of the stock price since the prediction task is to predict how much the stock price will change in a given day.
- Model building and evaluation. Build a regression model to predict the changes in the stock price as a function of changes in its previous k days as well as its twitter sentiment for the past k days ((k is a parameter you can tune, could be 1 day, 2 days, and so on). You will need to divide the data into training, validation, and test sets. You can use any regression technique to train the model.

References:

- Twitter mood predicts the stock market(2010).
- Exploiting Topic based Twitter Sentiment for Stock Prediction (2013)

D. Product Recommendation System (Type = group)

Recommender systems are widely used to address the information bottleneck problem. For example, in e-commerce websites such as Amazon, where there are millions of items/products sold, finding the right item can be a challenging task. In this project, you will design and develop a product recommendation system based on review and product data provided by Amazon. You will need to apply different product recommendation algorithms and evaluate their effectiveness on the data. The specific tasks to be performed are as follows:

- Data collection and preprocessing. Download the Amazon product data from the one of the following two websites:http://jmcauley.ucsd.edu/data/amazon/ or https://snap.stanford.edu/data/web-Amazon.html. To reduce the size of data you need to process, you can limit the scope to one of the following product categories: books, electronics, or home and kitchen. Preprocess the data to extract the userIDs, product information, and user reviews/ratings. You can limit the analysis to users who have rated sufficiently many products in the dataset (the number of reviewed products is a threshold that you need to figure out when analyzing the data).
- Model building and evaluation. Apply different algorithms (such as item- or user-based nearest neighbor, principal component analysis, matrix factorization, etc) to do the recommendation. The recommendation can be based on items/products the user has previously rated (using both the review summary and review score fields) as well as the product description. You may need to use Hadoop especially given the size of data to be processed. Compare the performance of the different algorithms.

- Develop a web-based interface for your recommender system. Specifically, the system will allow a user to login, enter reviews for some products (chosen from those that are in the dataset that you have). Using these reviews and information about the products, make a ranked list of top-3 recommendation to the user.

Instead of Amazon product recommendation, you can also develop a movie recommender system using data from MovieLens and IMDB data websites. Your project must include a data collection and preprocessing, model building and evaluation, as well as a web-based prototype system to deploy your recommendation results. Another possibility is to develop a music recommendation system based on the Yahoo music ratingdatasets.

References:

- Item-based collaborative filtering recommendation algorithms (2001).
- Matrix factorization techniques for recommender systems (2009)

E. Sports Analytics (type = individual/group)

Sports analytics is an emerging area that has been widely adopted by nearly all professional sports teams to gain an advantage over their competitors. In this project, you will collect, preprocess, and analyze data from a chosen sport to predict a match winner. Specifically, given a pair of teams--a home team and a visiting team--your task is to predict who will emerge as the winner of the match. The specific tasks to be performed for this project are as follows:

1. Data collection and preprocessing. Download the historical data about previous matches, which should include statistics about the games and the teams that played against each other. Create a feature vector for each match which includes the following information
    - Statistics from the last k games (e.g., last 5 home games and last 5 away games) for the home and visiting teams (e.g., average number of points scored, average number of points allowed, average number of penalties committed, and other statistics about their offense and defense). Make sure you use only the statistics from previous matches and not the match you're trying to predict. You may also want to include the opponent strength when calculating the statistics from the last k games.
    - For group project, you need to include other information beyond the match statistics. For example, add some statistics about the players who played in the previous matches. Another possibility is to include information about the venue of the matches (e.g., what was the crowd size in the previous k home games---is it sold out, less than Â3/4 sold out, etc). Another possibility is to add betting information about the matches (e.g., which team is the favorite or underdog).

   You should make sure you have at least created the feature vectors for at least 300 matches in order to train a good prediction model. Do not restrict yourself to datasets from a single year alone.
2. Model building and evaluation. You may model the match prediction either as a classification or a regression problem. For classification, the task is to predict the winning team, whereas for regression, the task is to predict the final score of the match (or the margin of the score between the home and visiting teams). You are free to apply any classification or regression techniques to the data. You will need to divide the data into training, validation, and test sets before doing your experiments.
3. For group project, you will need to develop a prototype system that will take a list of upcoming matches and make a prediction. The system should also display their previous prediction results and the actual outcomes of the previous games (so the accuracy of the prediction model can be assessed).

You can also read the following references below for NCAA March madness prediction and try to implement their approach (or your own approach) to predict the winners for this year's tournament.

References:

- March Madness Prediction: A Matrix Completion Approach (2015)
- Building an NCAA men's basketball predictive model and quantifying its success (2014)