Udacity Capstone Project: Denver Crime and Weather

David Timm
https://github.com/dtimm/mlnd-denver-crime

Project Overview:

I hope to analyze a dataset documenting something typically considered unpredictable and chaotic: crime. Crime is a constant component of any society, and responding to and anticipating crimes is key to the function of modern society. My intent is to create a sort of crime forecast, using weather as the key. Weather also drives huge forces in society and economics, and it severely affects the moods and daily lives of the people the world over. I hope that, by modeling weather and crime in the city of Denver, to create a meaningful prediction of expected crimes and their geographic local in the city.

Problem Statement:

The first step will be sourcing meaningful crime data and weather data for the area under consideration: Denver has an open data site and publishes up-to-date crime information going back five years, and Weather Underground has both an API to access weather data and a large corpus of historical weather. My next step is to break the city into meaningful clusters based on the locations of crimes. After that, I will correlate weather to the crimes that have occurred in the past. There are many regression algorithms that might have good results on this data, and I intend to try the basic sampling from scikit-learn as well as multi-layer perceptrons in TensorFlow. I will be predicting crimes based on weather and location across a variety of crime type, assigning a prediction to each location, date, and crime type.

Metrics:

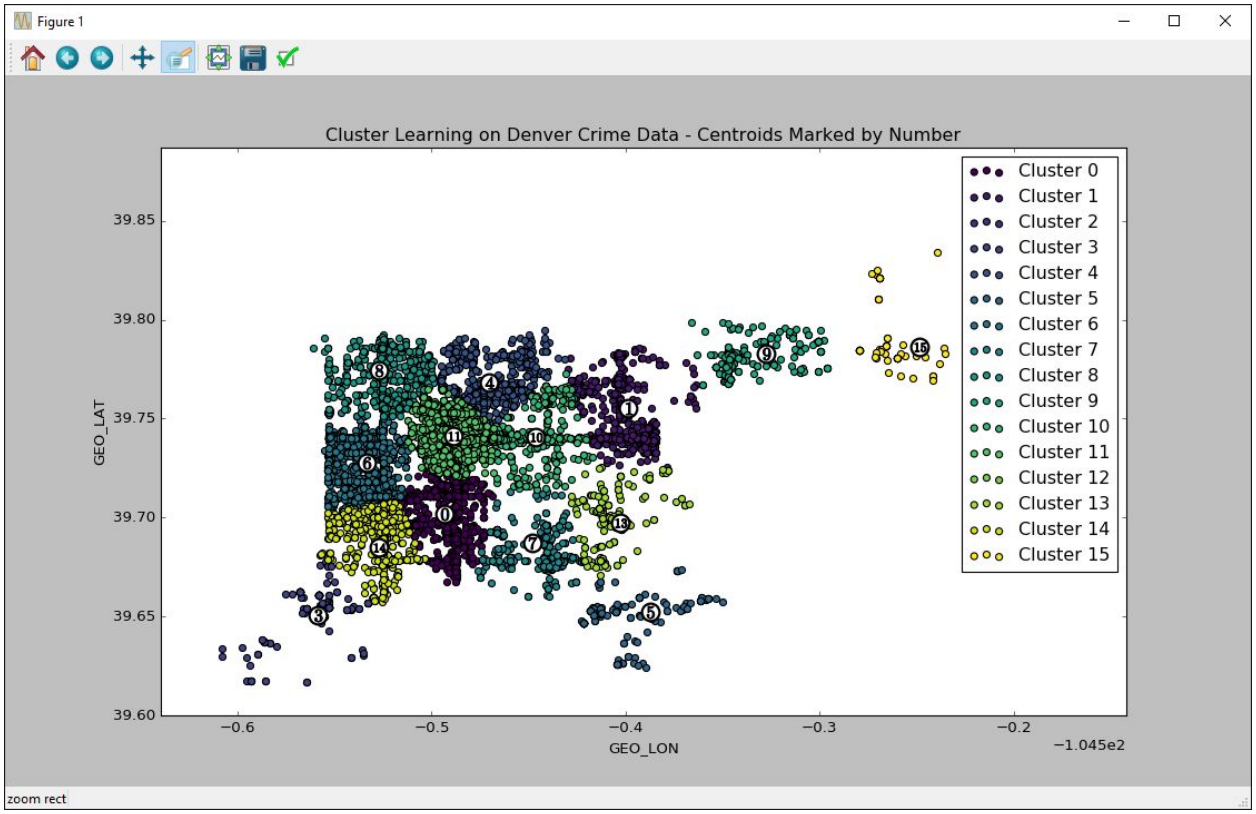
I will measure my success with R^2 score. I will measure my predictions for a set of dates against historical data not used to train my algorithm. I chose R^2 as my measure because other methods (such as average square error) are not easily comprehensible without a detailed knowledge of the data. My measurement will be against predictions of all the crime classes across a variety of dates (and thus weather patterns).

Udacity Capstone Project: Denver Crime and Weather

David Timm
https://github.com/dtimm/mlnd-denver-crime

Data:

I retrieved data from [Denver's open data site](). The dataset has nineteen features: INCIDENT_ID, OFFENSE_ID, OFFENSE_CODE, OFFENSE_CODE_EXTENSION, OFFENSE_TYPE_ID, OFFENSE_CATEGORY_ID, FIRST_OCCURRENCE_DATE, LAST_OCCURRENCE_DATE, REPORTED_DATE, INCIDENT_ADDRESS, GEO_X, GEO_Y, GEO_LON, GEO_LAT, DISTRICT_ID, PRECINCT_ID, NEIGHBORHOOD_ID, IS_CRIME, and IS_TRAFFIC. Of these features, I will only be using four: OFFENSE_CATEGORY_ID, FIRST_OCCURRENCE_DATE, GEO_LAT, and GEO_LON. The rest are either redundant, unnecessary, or both. I removed all entries that had no or incomprehensible location data. I then used K Means to determine if there were reasonable clusters for the crime data based on GEO_LAT and GEO_LON. The silhouette scores were best with 2, 3, and 16 clusters. This is the plot for drug and alcohol offenses created with the project 3 tools on 16 clusters:



There are several clear breaks as well as several dense areas split into multiple areas.

Udacity Capstone Project: Denver Crime and Weather

David Timm
https://github.com/dtimm/mlnd-denver-crime

I also calculated silhouette scores per crime to make sure the values weren't too skewed there:
Silhouette Scores:

all-other-crimes	0.454
murder	0.461
arson	0.415
auto-theft	0.398
theft-from-motor-vehicle	0.384
drug-alcohol	0.447
larceny	0.473
aggravated-assault	0.432
other-crimes-against-persons	0.433
robbery	0.455
burglary	0.368
white-collar-crime	0.429
public-disorder	0.413

The next step is to add time-series weather data for each centroid in the map. I pulled the centroid coordinates and correlated them to zip codes, unfortunately, it proved unviable to get weather data at the resolution of zip codes from any source I contacted, so I ended up using weather data from the KAPA station in Denver, CO for all of my results.

I grouped all crimes into counts for each cluster and date of first occurrence, then I added the weather data for each date into the dataset. I ran pandas' describe method on the data a final time to verify that the data I produced was sensible. There are quite a few columns, so it is included as data/crime_weather_stats.csv. The crimes reported all have a 25th percentile value of zero with the exception of traffic-accident. This is accurate to the reported data. The weather is also sensible. It's time to do some regression.

Algorithms:

I split the data into test and train sets, and ran each of the regressors available in scikit-learn against them. I narrowed the list of regressors to DecisionTreeRegressor and Perceptron. Next, I ran grid search to find the best set of parameters to maximize R^2 on my test data. The Perceptron was extremely sensitive to randomness in its initialization, which isn't good with a constantly expanding and changing dataset. I implemented a multi-layer perceptron with ReLu activation in TensorFlow to see if I could get more consistency (see this Git branch). The best R^2 I achieved was 0.100, and never matched the single layer Perceptron from scikit-learn. The DecisionTreeRegressor still gave the best and most generalizable results. I believe that the sparsity of the data (most day-category combinations are zero) gave the DTR an advantage

over other regression algorithms. After grid search, I settled on a DecisionTreeRegressor with a max depth of seven nodes and minimum sample split of 50. This minimized the variance and bias on the test dataset, resulting in an R^2 of 0.516.