|  |  |  |
| --- | --- | --- |
|  | Anthony Olivieri  Syracuse iSchool  Applied Data Science  SUID: 370774094  aolivier@syr.edu | |
|  | |  |

Portfolio

# PROJECTS

## Data Visualization

The main goal of this project was to convey meaningful insights through visual communication. Data as raw numbers or text requires the audience to stop, focus, concentrate, and possibly calculate values to derive meaning. To express that value quicker and in a more memorable fashion the data can be arranged in symbols and colors. Humans can evaluate colors and symbols –sizes, positions, patterns—in a preattentive manner, meaning communication is instant and more pleasing to the audience.

**Technologies**: R, ggplot, Illustrator

**Techniques**: data mining, descriptive analytics, grouping and aggregation, plotting, data transformation, illustration, color theory

**Ethical consideration**: The charts on this poster can only convey insights into the data as it is: raw arrest records from Pittsburgh City Police. There is no way to display the effects of generational poverty or policing patterns.

**Project document**: [PDF Poster](https://github.com/oxenfree/Syracuse-iSchool-Data-Science-Portfolio/blob/master/Projects/Data-Vizualization/Pittsburgh-Crime-Stats-Poster.pdf)

## National Real Estate Analysis

This project aims to solve a financial investment problem. Given an investment opportunity in real estate, what three zip codes would yield the best return on that investment. This is obviously a prediction problem and because the data is median home values per zip code reported each month, it’s also a time-series problem. Time-series analysis offers interesting statistical challenges such transforming data to achieve stationarity and testing stationarity using the Dicky-Fuller test as well as testing for auto-correlation. The standard forecasting model used for time-series, ARIMA, was not used in this project in favor of a new library by Facebook called Prophet.

**Technologies**: python, jupyter notebook, Facebook Prophet, pyplot, seaborn, basemap

**Techniques**: data mining, plotting, data transformation, predictive analytics, machine learning, time-series, test statistics

**Ethical consideration**: Real estate investment using market indicators or demographic data may worsen economic conditions for residents of that area.

**Project Document**: [Jupyter Notebook](https://github.com/oxenfree/Syracuse-iSchool-Data-Science-Portfolio/blob/master/Projects/National-Real-Estate-Analysis/main.ipynb)

## Pittsburgh Real Estate Analysis

Using the lessons learned from the National Real Estate Analysis project, I decided to evaluate all zip codes around Pittsburgh with the same goal in mind: find an area worth investing in. After the specific zip codes which would garner high returns were identified, I then worked to use local home value, demographic, and home attribute data to create a price predictor for individual properties. The time-series analysis gave way to a regression prediction given individual attributes of a neighborhood and home.

Home sales records, and demographic data were gathered from local government records as well as the Census Bureau. Individual home attributes were gathered from Zillow’s API. I contacted Zillow to have my API call limit extended from 1,000 to 5,000 per day and ran an API client program for six days.

The data attributes were evaluated for correlation, scaled, added to an sklearn pipeline column transformer, imputer, and grid-search cross validation then fed into machine learning libraries. The results were evaluated for root mean squared error.

**Technologies**: python, jupyter notebook, Facebook Prophet, pyplot, seaborn, basemap, sklearn, Random Forest Regressor, Linear Regressor, Grid Search Cross-Validation

**Techniques**: data gathering, data mining, plotting, data transformation, descriptive analytics, predictive analytics, machine learning, time-series, test statistics, data visualization

**Ethical consideration**: Real estate investment using market indicators or demographic data may worsen economic conditions for residents of that area.

**Project Document**: [Jupyter Notebook](https://github.com/oxenfree/Syracuse-iSchool-Data-Science-Portfolio/blob/master/Projects/Pittsburgh-Real-Estate-Analysis/main.ipynb), [Zillow API Client](https://github.com/oxenfree/Syracuse-iSchool-Data-Science-Portfolio/blob/master/Projects/Pittsburgh-Real-Estate-Analysis/zillow_api_call.py)

## Political Tweets Sentiment Analysis

Twitter popularity as a stand-in for polling might be useful as another data point for predicting political race outcomes. This project attempts to analyze tweet data such as number of retweets and sentiment of tweets concerning political campaigns. Data from Twitter was gathered for several days for four upcoming political campaigns and six previous campaigns. The data from winning and losing campaigns was evaluated to identify patterns or sentiment as a signal to possible popularity and a possible campaign win. The tweet data were gathered, mined for sentiment, evaluated for “popularity” (by retweets and interaction with the candidate) and used in Multinomial Naïve Bayes and Support Vector Machines models to predict outcome.

**Technologies**: python, jupyter notebook, pyplot, seaborn, basemap, sklearn, Multinomial Naïve Bayes, Support Vector Machines, Grid Search Cross-Validation

**Techniques**: data gathering, text mining, plotting, data transformation, descriptive analytics, predictive analytics, machine learning, sentiment analysis, data visualization

**Ethical consideration**: Politics on Twitter and social media promotion of candidates can be influenced by special interests or foreign entities. Any analysis should account for these possibilities.

**Project Document**: [Jupyter Notebook](https://github.com/oxenfree/Syracuse-iSchool-Data-Science-Portfolio/blob/master/Projects/Political-Tweets-Sent-Analysis/main.ipynb), [Twitter API and Sentiment Client](https://github.com/oxenfree/Syracuse-iSchool-Data-Science-Portfolio/blob/master/Projects/Political-Tweets-Sent-Analysis/twitter_sent.py)