Jakub Zurakowski, Michael Sibbald and Alan Cahill

C00205431, C00206817, C00174533

4th Year Data Science Project Proposal

Investigating Baltimore Arrest Rates

Institute of Technology, Carlow

Table of Contents

[Purpose and goals 2](#_Toc525667727)

[Project strategy 2](#_Toc525667728)

[Scope 2](#_Toc525667729)

[Deliverables 2](#_Toc525667730)

[Functional interfaces 2](#_Toc525667731)

[The data 2](#_Toc525667732)

[Standards 3](#_Toc525667733)

[Reference/research documents 3](#_Toc525667734)

# Purpose and goals

The purpose of this project is to investigate Baltimore crime / arrests and provide statistics and predictions for Baltimore police department to increase their effectiveness.

The students will also attempt to illustrate the dangerous zones in Baltimore using google maps and the frequency of arrests.

Machine learning can be used to attempt to predict future crime locations and rates.

# Project strategy

The students will retrieve the data from the Baltimore police department API [1][2], process the data using python and display several statistical graphs using Flask, Jinja 2 templates, Google Maps and JavaScript.

# Scope

The data retrieved has 36,853 entries of arrests as of 20th September 2018. The data spans up from 1st January 2017 to August 2018. The students will develop models to determine the most susceptible areas, times and people to crime in Baltimore. A heat map will be developed for Baltimore that will be possible to alter based upon the time of the day. People will also be characterised on susceptibility to crime based upon their age, race and sex.

# Deliverables

A flask-based web-application displaying statistics and predictions in html format.

The following items will be delivered:

* Functional Specification
* Design Document
* Working Implementation
* Project Report

# Functional interfaces

The web-application will attempt to contain the following features:

* Heat-like map of Baltimore showing where crime is most common. The heat map will be adjustable via a slider that will change the map based on the time of day.
* Various graphs visualising the processed data. These will be updated dynamically as new data is published, allowing the user to see the change in crime over time.
* Prediction of crime occurrence in the city for the following months.
* Suggestions on preventing crime. This will include areas, times and people that crime is most likely to occur. This may even delve deeper allowing the user to see what crime in the selected area is most likely to occur at the current time and what profile of person is most likely to commit the crime.

# Data

The data is received in JSON format and filtered using a python script to suit developer’s needs. Below is a sample entry from the 36,853 records.

|  |
| --- |
| [  {  "sid": 1,  "id": "0609AFCF-A239-4115-B74E-80272A1FE44A",  "position": 1,  "created\_at": 1514490052,  "created\_meta": "883136",  "updated\_at": 1514490052,  "updated\_meta": "883136",  "meta": null,  "Arrest": "17201066",  "Age": "30",  "Sex": "M",  "Race": "B",  "ArrestDate": "2017-12-23T00:00:00",  "ArrestTime": "23:24",  "ArrestLocation": "900 STOLL ST",  "IncidentOffense": "4ECOMMON ASSAULT ",  "IncidentLocation": "900 STOLL ST",  "Charge": "1 1415",  "ChargeDescription": "2ND DEG ASSAULT",  "District": "Southern",  "Post": "913",  "Neighborhood": "Brooklyn",  "Longitude": "-76.598589000000",  "Latitude": "39.230243000000",  "Location 1": [  null,  "39.230243000000",  "-76.598589000000",  null,  false  ]  }  ] |

# Standards

The python code will follow the pep-8[3] standard.

# Reference/research documents

[1] BPD Arrests | Open Baltimore | City of Baltimore's Open Data Catalog. 2018. *BPD Arrests | Open Baltimore | City of Baltimore's Open Data Catalog*. [ONLINE] Available at: <https://data.baltimorecity.gov/Public-Safety/BPD-Arrests/3i3v-ibrt>. [Accessed 20 September 2018]

[2] Socrata Developer Portal | Socrata. 2018. *Socrata Developer Portal | Socrata*. [ONLINE] Available at: <https://dev.socrata.com/foundry/data.baltimorecity.gov/icjs-e3jg>. [Accessed 20 September 2018]

[3] Python.org. 2018. *PEP 8 -- Style Guide for Python Code | Python.org*. [ONLINE] Available at: <https://www.python.org/dev/peps/pep-0008/>. [Accessed 20 September 2018]