Polynomial Interpolation and K-Mean Mining to Predict Crime Rates

The Gwinnett School of Mathematics, Science, and Technology
Lawrenceville, Georgia

Team Leader: Anish Goyal

Partner(s): None

Teacher: Nguyen

Table of Contents

Return to Table of Contents

DIrections and Tips

Delete this entire page before submitting FINAL logbook check (not before)

- Remember that this is a legal document.
- Anyone should be able to follow exactly what you did in your project by reading your logbook. That's the level of detail you need.
- You MAY NOT delete any entries/data from this notebook. If you need to delete anything, you should strike through it (format > text > strikethrough). You can also make multiple versions of your entries, if appropriate.
- If you did work on other documents, you may cut and paste the items from those other locations. Just provide citations/reference those other locations. If you can't cut & paste easily (like you are referencing a physical lab notebook), you should decide if it's important to scan in the document or to just reference your previous work.

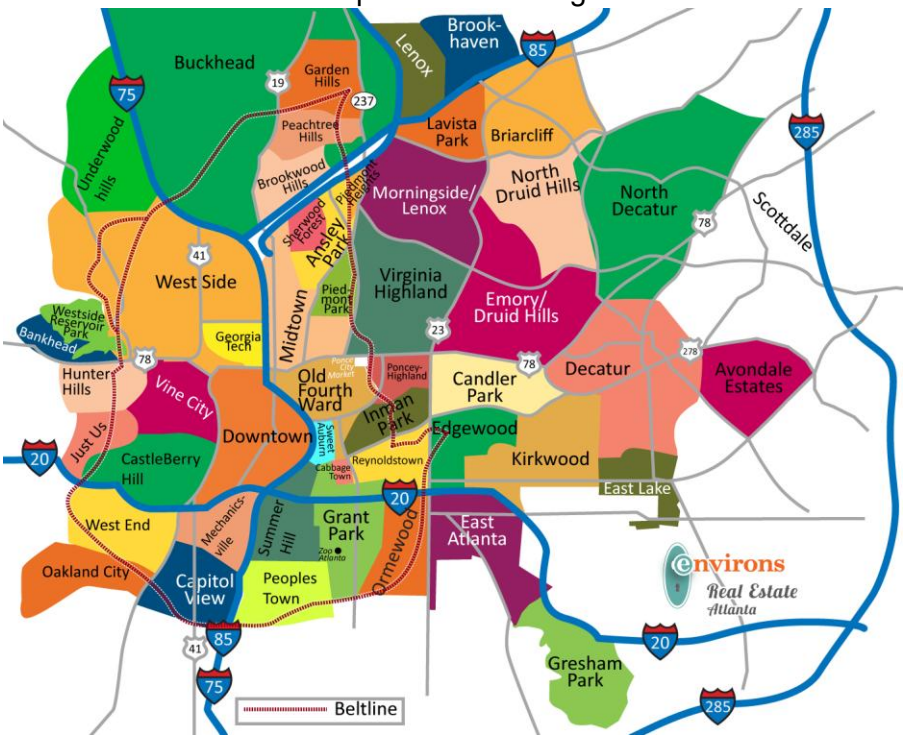
EVERY SKETCH OR DESIGN OR CODE you create should be included in your journal. Every design change must be documented. If you have any physical lab set ups or things you are building, include photos in your reflection entries. You must have AT LEAST SOME PICTURES that include **you** doing work on your project.

- Every day you work on a project, you should include a reflection entry with a summary of what you accomplished that day, any ideas you discussed, and other important ideas (even if you're not sure and are just considering them).
- Use your reflection entries to plan out what your next day should accomplish.
- At some point, you will need to do project planning, and you should include images of your project planning document (like a kanban board in your journal).
- Research/bibliography pages should be set up correctly using proper.
- If appropriate, title your entries (not reflection entries but other items in your lab notebook). You can insert ADDITIONAL sections in your logbook, but you must have everything in the template.
- Date all entries and put your initials. If you modify the entry, put modified dates and re-initial. You can also make version 1, 2, etc. for entries.
- Include page numbers

Return to Table of Contents

Brainstorming

- **Data visualization applied after post-processing.**
  - This is a map of Atlanta's neighborhoods:



Perhaps it is possible to color the neighborhood crime intensities using the COBRA data sets (hopefully I will have a blank vector in the future, which is easier to fill). I will have to research sci-kit visualizations on Python.

- **I have to figure out a method to clean the COBRA data sets.**
  - Since the data is raw and not processed, there are several thousand lines of null data present in the Atlanta data sets.
  - Maybe there is some kind of software that automatically cleans up .csv files? If not, I will have to create a program that cleans the data.
- **How do I implement a crime score for each crime type?**
  - I will have to research the severity for every crime type that is reported in the COBRA sets. This includes various misdemeanors and felonies
  - This will allow me to assign a crime score to each neighborhood on a particular day and predict the likelihood of different crimes occurring
- **Is it viable to add supervised learning algorithms to analyze the COBRA data sets and/or dimensionality reduction?**
  - ~~Dimensionality reduction would be difficult to implement, so it does not seem like a realistic goal to pursue (maybe in a future iteration). It could be useful, however, if it explained whether certain discriminants had significantly explained variances~~. I may end up using dimensionality reduction after all to reduce the data to a manageable size yet preserve its integrity (this may also be how I clean the null values of the COBRA data sets)
  - Various supervised learning algorithms could be applied, but I will only be able to test their effectiveness by measuring the predicted outcome with the actual crime rate. This algorithm comparison aspect of the project could be a science fair project in itself.

Return to Table of Contents

Research

Source #1

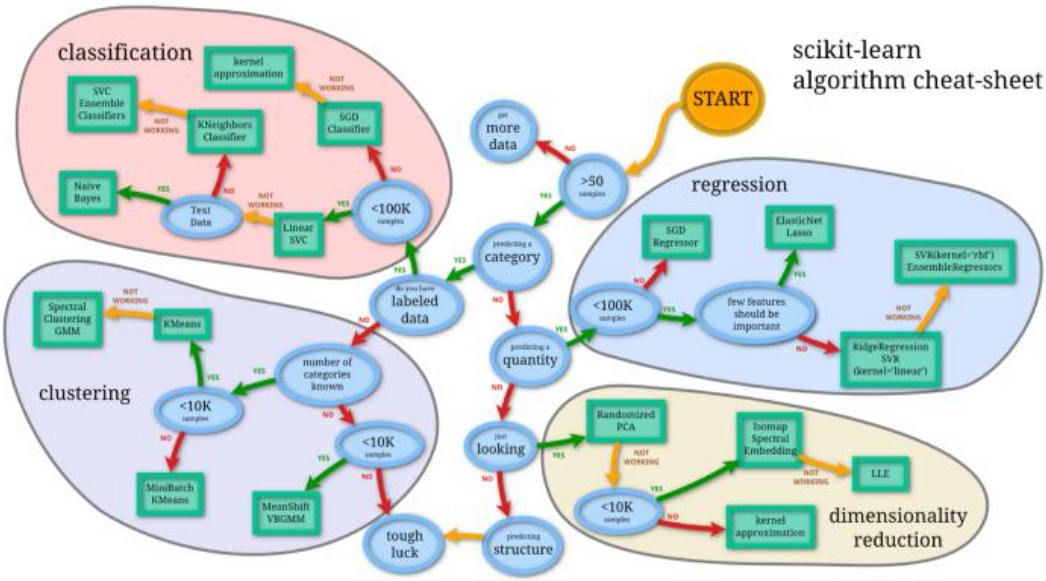
Type of Source:
Website

APA Citation, with url or DOI:
Towards Data Science. (2019, March 11). *Which machine learning model to use?* Towards Data Science. Retrieved December 4, 2021, from https://towardsdatascience.com/which-machine-learning-model-to-use-db5fdf37f3dd

Notes:

- Regression:
  - Regression problems forecast a continuous variable given a set number of features, such as estimating property prices given house data such as size, number of rooms.
  - Accurate but slow regression methods:
    - Random forests
    - Neural networks (needs many data points)
    - Gradient boosting trees (easier to fit)
  - Fast
    - Decision trees
    - Linear regression
- Clustering & Dimensionality Reduction
  - Hierarchical clustering is a cluster analysis technique that aims to create a hierarchy of groups. There are two sorts of strategies for hierarchical clustering:
    - Agglomerative
      - Each observation begins in its own cluster, and as one advances up the hierarchy, pairs of clusters are merged.
    - Divisive
      - All observations begin in one cluster, and as one proceeds down the hierarchy, splits are performed recursively.
  - Nonhierarchical Clustering:
    - DBSCAN (do not need to specify a k-value)
    - K-means
    - Gaussian mixture models
  - Dimensionality reduction
    - Each axis of the ellipsoid represents a principal component, and principal component analysis fits an n-dimensional ellipsoid to the data. If the variance along one axis of the ellipsoid is modest, we lose just a little amount of information by eliminating that axis and its corresponding primary component from our representation of the dataset.
    - Topic modeling
      - Topic modeling is a statistical method for identifying the abstract "themes" that appear in a set of documents. Topic modeling is a text-mining technique for identifying hidden semantic structures in a text body (this could be useful for analyzing documents such as utility bills for patterns and OCR).

Return to Table of Contents



Questions this source helped me answer:
What algorithms should I use for my project? Which algorithms are best suited for meeting the needs of this project? When should I use each algorithm? What order should the algorithms be used?

Questions I have for the author:
Is it common for data scientists to undergo PCA dimensionality reduction before classifying data?
Does it matter whether you use dimensionality reduction before or after classifying the data?
How can I increase the number of features added to the data after clustering with K-means?

Vocabulary:
Hierarchical clustering: a method of cluster analysis which seeks to build a hierarchy of clusters
Agglomerative clustering: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.
Divisive clustering: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.
Topic modeling: a type of statistical model for discovering the abstract "topics" that occur in a collection of documents.

Summary (paraphrase!):
Every sci-kit learn machine learning algorithm can be split into four categories based on their function: classification, regression, clustering, and dimensionality reduction. Depending on the intended outcome that the user wants once the data is processed, the data scientist can use a variety of methods to simplify the data being analyzed and use the best algorithm. If you want to predict a category with labeled data, you should use a classification algorithm; if you have unlabeled data, you should use clustering. If you want to predict a quantity, regression models are the best. If you are "just looking" (aka trying to simplify the data to further determine which method is the best to use according to sample size), dimensionality reduction with PCA is recommended.

Implications for my project/thinking:
For my project, I will be using Principal Component Analysis to dimensionally reduce the data so that it is easier to process and analyze. I may even make the use of PCA an independent variable in itself in my experiment because using PCA can drastically affect the accuracy of the predictions of data. I also know that I will need to have a separate output table for my supervised algorithms (K-means and Naïve Bayes), as they sort categorically. As a result, I will need to sort the various different crimes on the COBRA data set into different categories, and I think I will categorize them based on severity and assign different values based on that.

Return to Table of Contents

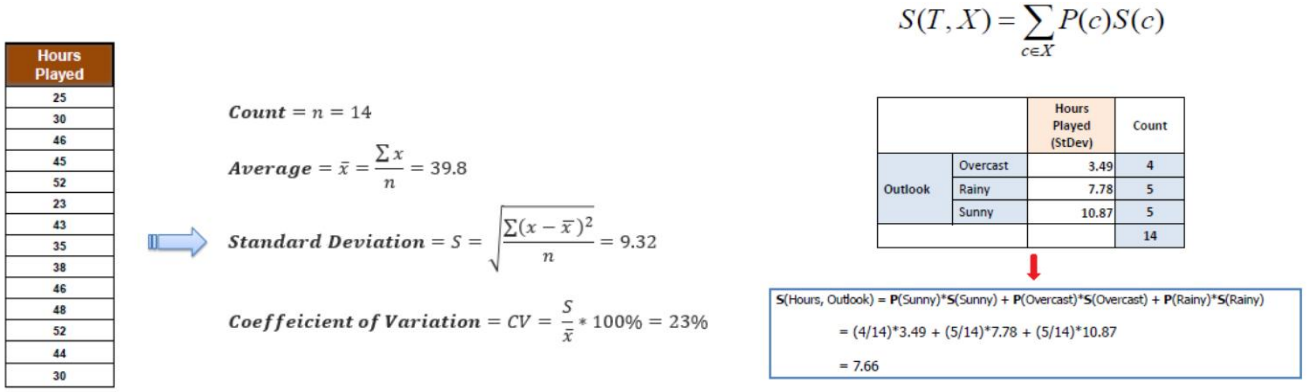
Source #2

Type of Source:
Website

APA Citation, with url or DOI:
Sayad, S. (2012, November 8). Decision Tree - Regression. SaedSayad. Retrieved December 3, 2021, from http://www.saedsayad.com/decision_tree_reg.htm

Notes:

- Can handle both categorical and numerical data
- Generates regression or classification models by breaking down a dataset into smaller and smaller subgroups while simultaneously developing an associated decision tree.
- A decision node has two or more branches, each of which represents a value for the characteristic under consideration.
- A leaf node reflects a numerical target decision.
- The root node of a tree is the highest decision node that corresponds to the best predictor.
- Standard deviation is used to calculate the homogeneity of a numerical sample
    - A numerical sample is considered completely homogeneous if it has a standard deviation of zero
- The coefficient of variation tells the algorithm when to stop branching
- Standard deviation reduction
    - Based on the reduction in standard deviation after dividing a dataset by an attribute
        1. The standard deviation of the data set is calculated
        2. The data set is then split into its various attributes or "branches," in which the standard deviation of each branch is calculated and subtracted from the original calculated standard deviation
        3. The branch with the highest standard deviation is assigned as a decision node by the algorithm
        4. The data is then categorized based on the values of the selected branch until the entire data set is processed
- Termination criteria and overfitting
    - When analyzing data with decision trees, it is always a good idea to implement termination criteria. This is so that the data splits the correct amount depending on how many trees can be generated, and no branch remains too small or too large
    - This can be done by setting a threshold on the coefficient of variation so that the algorithm will make the proper amount of branches
    - Whenever a coefficient of variation goes past the threshold in generating a new branch, it further splits the branch until the coefficient is met. This process is known as "overfitting"



Hours Played
25
30
46
45
52
23
43
35
38
46
48
52
44
30

$$Count = n = 14$$

$$Average = \bar{x} = \frac{\sum x}{n} = 39.8$$

$$Standard\ Deviation = S = \sqrt{\frac{\sum (x - \bar{x})^2}{n}} = 9.32$$

$$Coeffeicient\ of\ Variation = CV = \frac{S}{\bar{x}} * 100\% = 23\%$$

$$S(T, X) = \sum_{c \in X} P(c) S(c)$$

		Hours Played (StDev)	Count
Outlook	Overcast	3.49	4
	Rainy	7.78	5
	Sunny	10.87	5
			14

S(Hours, Outlook) = P(Sunny)*S(Sunny) + P(Overcast)*S(Overcast) + P(Rainy)*S(Rainy)

= (4/14)*3.49 + (5/14)*7.78 + (5/14)*10.87

= 7.66

Return to Table of Contents

Questions this source helped me answer:

- What are decision trees?
- What do decision trees use to calculate decision nodes?
- What is the coefficient of variation and how is it calculated?
- What is a decision tree used for?
- How do you make splits in a decision tree?
- How do you determine the most important variables in decision trees?

Questions I have for the author:

- What is the difference between a decision tree and a random forest?
- What are the disadvantages of a decision tree?
- What is the time complexity of building decision trees according to their depth?
- How do you handle missing values in a decision tree?

Vocabulary:
Homogeneity: A measurement of how alike the data is
Splitting: The process of dividing a node into sub-nodes
Pruning: The process of reducing the complexity of a decision tree by removing negligible sections of data
Overfitting: Whenever the coefficient of variation surpasses the threshold defined by the user while creating a decision node. Usually occurs during pruning.

Summary (paraphrase!):
Decision trees construct regression models in the form of a tree structure by breaking down the data into incrementally smaller subsets and taking the standard deviation of each subset recursively to make a prediction about the data. Decision trees start at the root node, which is at the top of the tree and represents the entire sample space. The decision nodes are where the tree start branching off and is the result of a split based on parameters that the user defines. Nodes that cannot be divided any further are called leaf nodes. Pruning reduces the size of a decision tree by getting rid of unimportant sections, which leads to more accurate predictions, less computing time, and less overfitting. Whenever the standard deviation of a branch is less than the coefficient of variation for the data set, the tree stops branching unless overfitting occurs.

Implications for my project/thinking:
For my project, I will be using decision trees as one of the supervised machine learning algorithms to predict crime rates. This source is important for my project because it tells me how decision trees work and how they calculate the predicted outputs of a data set, categorical or numerical. It also tells me how overfitting works and when to best overfit the data. Finally, this source tells me how to calculate decision nodes on my own and the scenarios that decision trees are best utilized for. This tells me how the sci-kit learn decision trees will output all of the information to the output console, make any changes, and troubleshoot if necessary.

Return to Table of Contents

Source #3

Type of Source:
Website

APA Citation, with url or DOI:
Stojiljković, M. (2019, April 15). *Linear Regression in Python – Real Python*. Real Python. Retrieved December 14, 2021, from https://realpython.com/linear-regression-in-python/

Notes:

- Linear regression is one of the most essential and extensively used regression techniques.
  - It's also one of the most basic regression techniques.
  - The ease with which the results can be interpreted is one of its key merits.
- Regression looks for connections between variables. You try to build a relationship between the features in a data set based on the assumption that at least one of them is dependent on the others.
- In linear regression, you consider a phenomenon of interest and a number of observations that are related with that phenomenon.
- There are always two or more features in each observation.
- The dependent features are called the dependent variables, outputs, or responses.
- The independent features are called the independent variables, inputs, regressors, or predictors.
- Regression problems usually have one continuous and unbounded dependent variable that can either be numerical or categorical.
- Regression is required to determine whether and how one phenomenon affects another or how numerous variables are connected.
  - It essentially shows whether two variables are connected and to what extent they are connected
- Regression is also beneficial when forecasting a response with a new set of predictors once the model has been trained with said predictors.
- The linear regression equation is $y = \beta_0 + \beta_1 x_1 + \cdots + \beta_r x_r + \varepsilon$.
  - $\beta_0, \beta_1, \ldots, \beta_r$ represents the regression coefficients.
  - ε is a constant that represents the random error.
  - Where $\mathbf{x} = (x_1, \ldots, x_r)$ and r is the number of predictors
  - The linear regression equation is used to find the estimated regression function, or f(x).
- The estimated regression function is $f(\mathbf{x}) = b_0 + b_1 x_1 + \cdots + b_r x_r$.
  - Where $b_0, b_1, \ldots, b_r$ represents the predicted weights
- The differences between y_i and $f(\mathbf{x}_i)$ or the linear regression equation and the estimated regression function for every observation in the data set are called the residuals
  - The best predicted weights have the smallest residuals
- The coefficient of determination (R^2) tells you how much y is dependent on x
  - A larger R^2 value means that the regression function can be used to explain the trends of the data set for different features or that the model "fits" the data and vice-versa.

Questions this source helped me answer:

- What is linear regression?
- What is linear regression used for?
- How does linear regression work?
- How do you implement linear regression in Python?
- What does the linear regression equation and estimated regression function tell you?
- How do I analyze a linear regression model for overfitting and underfitting?
- What does the R^2 value tell you about a regression model?

Return to Table of Contents

Questions I have for the author:

- How do you perform linear regression on a Pandas data frame?
- How do you store the results of a linear regression in a Pandas data frame?
- How do you check for correlation among continuous and categorical variables?
- How do you calculate the number of dots that lie above and below the regression line with the R^2 value?
- How do you remove features from a linear regression model using PCA while retaining the same R^2 value?
- How do you predict values from a linear regression from multiple groups and export it to a single data frame?
- How do you merge two linear regression prediction models?
- How does sklearn calculate the regression coefficients, coefficient of determination, and random error constant?

Vocabulary:
Underfitting: When a model's R^2 value is poor because it can't accurately capture data dependencies, usually due to its simplicity.
Overfitting: A model learns the existing data too well and has an extremely high R^2 value overall. Can be good or bad. Usually happens with large data sets. Often do not generalize well and yield small R^2 values when used with new data because intervals of increase or decrease can fluctuate.
Observation: Independent variable
Feature: 2 or more features are present for each observation that potentially affects it in some way. Features can also be influenced by each other.

Summary (paraphrase!):
Linear regression is one of the most essential and extensively used regression techniques that looks for connections among observations that each have two or more features. They can be used for numerical or categorical data. Dependent features are known as outputs/responses, and independent features are known as inputs/predictors. The linear regression equation is $y = \beta_0 + \beta_1 x_1 + \cdots + \beta_r x_r + \varepsilon$ and is used to find f(x), the estimated regressor function. The difference between the regression equation and estimated function for each observation is called a residual, and smaller residuals are better. Each estimated function has a coefficient of determination (R^2) that tells you how much y is dependent on x. A smaller R^2 value means that the estimated function does not fit or resemble the original model and the features have minimal statistical significance on the observations, while a larger R^2 value means that the estimated function resembles the original model very well and the features have a large statistical significance on the observations. Estimated functions with small R^2 values tend to underfit, which means they have not enough data to make a qualitative prediction on the outcome of the data or establish a relationship between variables, while estimated functions with large R^2 values tend to overfit, which means there is too much data, and the model will be inaccurate for new incoming data.

Implications for my project/thinking:
Linear regression will be one of the methods that I am using in this project. It is simple to use, results are easy to interpret, and it requires little to no computational power. With linear regression, I will use a number of features such as crime date, crime type, and location to predict what the crime occurrence measured as a percentage will be, or the observation.

Return to Table of Contents

Source #4

Type of Source:
Website

APA Citation, with url or DOI:
Saini, A. (2021, September 16). *Naive Bayes Algorithm: A Complete guide for Data Science Enthusiasts*. Analytics Vidhya. Retrieved December 14, 2021, from https://www.analyticsvidhya.com/blog/2021/09/naive-bayes-algorithm-a-complete-guide-for-data-science-enthusiasts/

Notes:

Questions this source helped me answer:

Questions I have for the author:

Vocabulary:

Summary (paraphrase!):

Implications for my project/thinking:

Return to Table of Contents

Source #5

Type of Source:
Website

APA Citation, with url or DOI:
Kaloyanova, E., Ganchev, M., & Guide, S. (2020, March 10). How to Combine PCA and K-means Clustering in Python? 365 Data Science. Retrieved December 13, 2021, from https://365datascience.com/tutorials/python-tutorials/pca-k-means/

Notes:

Questions this source helped me answer:

Questions I have for the author:

Vocabulary:

Summary (paraphrase!):

Implications for my project/thinking:

Return to Table of Contents

Reflection Entries

Entry #1

Date : 9/26

Worked on research, learned about the different types of machine learning algorithms there are and when to best use them, will continue to work on research to complete all five sources.

Initials: AG

Entry #2

Date : 10/05

Worked on research, learned about linear regression and Naïve-Bayes algorithms, and prospecting to use those algorithms for my project.

Initials: AG

Entry #3

Date: 10/14

Worked on research, found a scholarly article that talked about all of the different sklearn algorithms, and plan to use decision trees and random forests in my project.

Entry #4

Date: 10/27

Worked on background research outline, used research from logbook for my literature review, and I plan to finish my background research outline and hypothesis by next month.

Entry #5

Date: 11/11
Finished RulesWizard and required forms, turned in all signatures and mentorship signatures, and I plan to finish my background research outline and hypothesis by Christmas break.

Return to Table of Contents

Entry #6

Date: 12/18

Finished revising the forms that I had not submitted properly, worked on mentorship signatures and risk safety form, and I plan to finish my background research outline and hypothesis by Christmas break.

Entry #7

Date: 12/29

Finished background research outline over break, worked on research plan for engineering, and I plan to work on how I am going to collect data for this project as well as my procedure.

Entry #8

Date: 1/8

Worked on research plan (finished proof of concept), added procedure and hypothesis for beta I, plan to start the actual coding part of my project now. Also informed Mr. Nguyen that I plan on conducting RMSE tests instead of calculating the % change for each neighborhood.

Entry #9

Date: 1/21
Worked on beta I, cleaned up both the 2009-2018 and 2019 COBRA data sets, plan to begin importing the cleaned data to a data frame on Jupyter Notebook.

Entry #10

Date: 1/30
Worked on beta I, applied K-Mean clustering to cleaned data, created GitHub repo (currently empty), plan to begin applying supervised and unsupervised algorithms.

Entry #11

Date: 2/19
Worked on coding the project some more (the backend is basically done), partially coded PCA dimensionality reduction to create an optimal number of features for the data, but I am getting a weird error, don't know why? Plan to fix dimensionality reduction before March.

Return to Table of Contents

```
In [22]: def LDA(x, y):

             from sklearn.discriminant_analysis import LinearDiscriminantAnalysis

             init_lda = LinearDiscriminantAnalysis(n_components=2)
             x_lda = init_lda.fit(x,y).transform(x)
             return(x_lda)
             raise NotImplementedError

         x_lda = LDA(x, y)
         lda_var = x_lda.explained_variance_ratio_
         print('Explained variance ratio of the components: %s'
               % str(lda_var))
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_7152\3311163549.py in <module>
      8     raise NotImplementedError
      9
---> 10 x_lda = LDA(x, y)
     11 lda_var = x_lda.explained_variance_ratio_
     12 print('Explained variance ratio of the components: %s'

NameError: name 'x' is not defined
```

Entry #13

Date: 2/30

Successfully applied PCA dimensionality reduction, was able to determine the cause of the error as well as whether LDA reduction would be better for this project—it is not. I plan to finish Beta I before mid-March to April and, from there, Beta II will be significantly easier as it is basically just Beta I but with data visualizations.

Entry #14

Date: 3/16

Worked on Beta I, added sample data tables and additional experimental controls and I/O, plan to finish Beta I by next week.

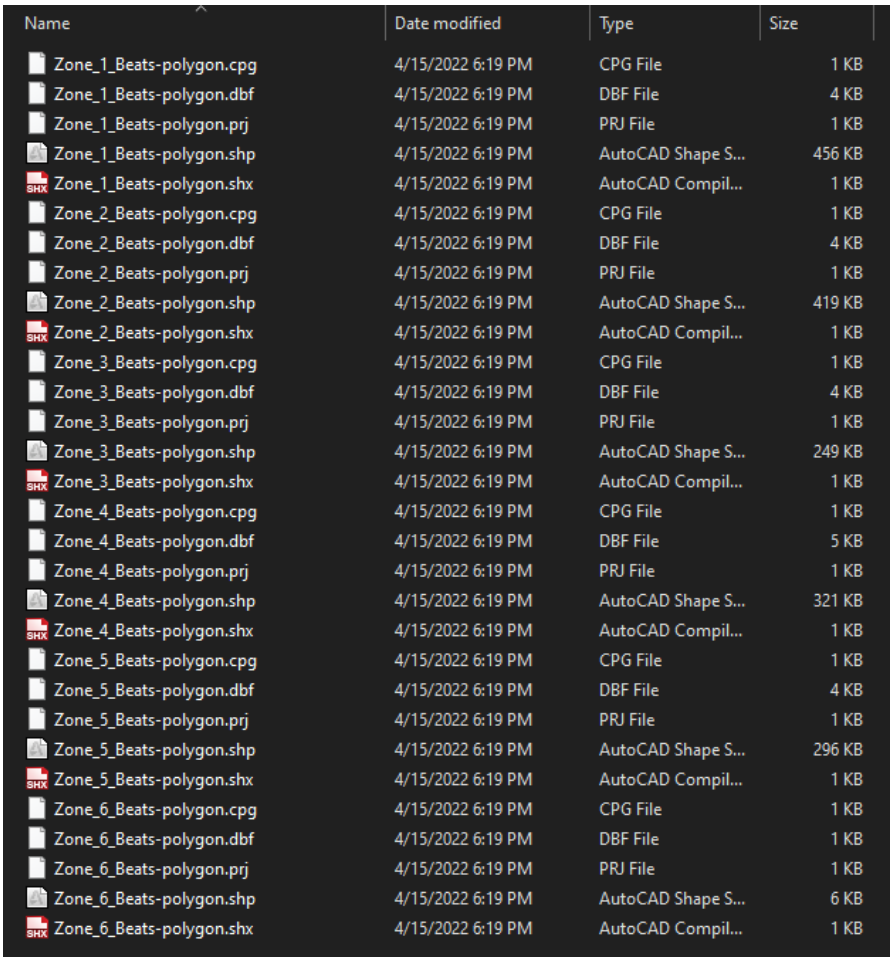
Entry #15

Date: 3/29

- The shapefile being used for the visualizations of this project will no longer be from the Atlanta Regional Commission (ARC). Instead, they will be .kml files sourced directly from the Atlanta PD website. The beat/patrol zone map can be found on the website by searching up “Zone (num) Beats”
    - The reason for this change is because I had to convert the ARC shapefile (originally in XML vector format) to a bitmap layout, which took lots of computing power to render a single image. Kml files are much easier to work with when using the Seaborn library
- Aggregated “crime score” (To be added to “outputs”)
    - A sum of the crime category counts for a particular neighborhood for a particular day
    - Category 1 crimes are weighted by 1000x because they are the most severe; category 2 by 100x; category 3 by 10x; and category 4 (larceny) by 1x

Return to Table of Contents

The predicted crime score is a dimensionless scalar value, so it has no units



Name	Date modified	Type	Size
Zone_1_Beats-polygon.cpg	4/15/2022 6:19 PM	CPG File	1 KB
Zone_1_Beats-polygon.dbf	4/15/2022 6:19 PM	DBF File	4 KB
Zone_1_Beats-polygon.prj	4/15/2022 6:19 PM	PRJ File	1 KB
Zone_1_Beats-polygon.shp	4/15/2022 6:19 PM	AutoCAD Shape S...	456 KB
Zone_1_Beats-polygon.shx	4/15/2022 6:19 PM	AutoCAD Compil...	1 KB
Zone_2_Beats-polygon.cpg	4/15/2022 6:19 PM	CPG File	1 KB
Zone_2_Beats-polygon.dbf	4/15/2022 6:19 PM	DBF File	4 KB
Zone_2_Beats-polygon.prj	4/15/2022 6:19 PM	PRJ File	1 KB
Zone_2_Beats-polygon.shp	4/15/2022 6:19 PM	AutoCAD Shape S...	419 KB
Zone_2_Beats-polygon.shx	4/15/2022 6:19 PM	AutoCAD Compil...	1 KB
Zone_3_Beats-polygon.cpg	4/15/2022 6:19 PM	CPG File	1 KB
Zone_3_Beats-polygon.dbf	4/15/2022 6:19 PM	DBF File	4 KB
Zone_3_Beats-polygon.prj	4/15/2022 6:19 PM	PRJ File	1 KB
Zone_3_Beats-polygon.shp	4/15/2022 6:19 PM	AutoCAD Shape S...	249 KB
Zone_3_Beats-polygon.shx	4/15/2022 6:19 PM	AutoCAD Compil...	1 KB
Zone_4_Beats-polygon.cpg	4/15/2022 6:19 PM	CPG File	1 KB
Zone_4_Beats-polygon.dbf	4/15/2022 6:19 PM	DBF File	5 KB
Zone_4_Beats-polygon.prj	4/15/2022 6:19 PM	PRJ File	1 KB
Zone_4_Beats-polygon.shp	4/15/2022 6:19 PM	AutoCAD Shape S...	321 KB
Zone_4_Beats-polygon.shx	4/15/2022 6:19 PM	AutoCAD Compil...	1 KB
Zone_5_Beats-polygon.cpg	4/15/2022 6:19 PM	CPG File	1 KB
Zone_5_Beats-polygon.dbf	4/15/2022 6:19 PM	DBF File	4 KB
Zone_5_Beats-polygon.prj	4/15/2022 6:19 PM	PRJ File	1 KB
Zone_5_Beats-polygon.shp	4/15/2022 6:19 PM	AutoCAD Shape S...	296 KB
Zone_5_Beats-polygon.shx	4/15/2022 6:19 PM	AutoCAD Compil...	1 KB
Zone_6_Beats-polygon.cpg	4/15/2022 6:19 PM	CPG File	1 KB
Zone_6_Beats-polygon.dbf	4/15/2022 6:19 PM	DBF File	1 KB
Zone_6_Beats-polygon.prj	4/15/2022 6:19 PM	PRJ File	1 KB
Zone_6_Beats-polygon.shp	4/15/2022 6:19 PM	AutoCAD Shape S...	6 KB
Zone_6_Beats-polygon.shx	4/15/2022 6:19 PM	AutoCAD Compil...	1 KB

Entry #16

Date: 4/9

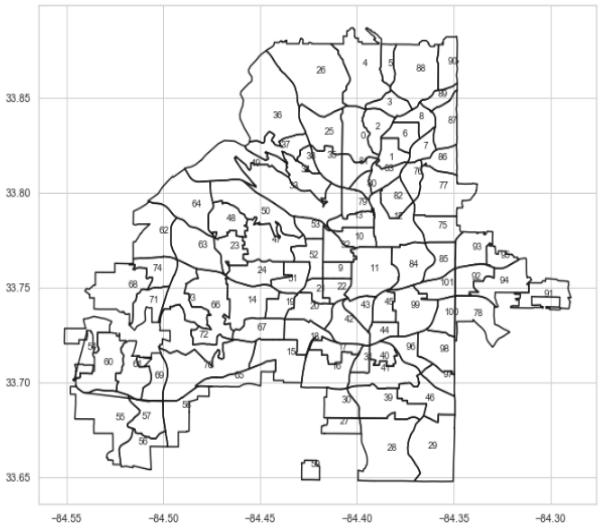
Worked on Beta II, started using geopandas and seaborn to code the visualizations but first I had to send the predicted crime scores of each neighborhood to one data frame. I plan to finish the visualizations by this week.

Return to Table of Contents

```
Neighborhood,Crime Score
Downtown,1951.1054283536478
Greenbriar,2015.3334355536704
Wildwood (NPU-C),2362.164674433794
Lindbergh/Morosgo,2102.0412452737014
Grant Park,2005.6992344736673
Sylvan Hills,2285.0910657937666
Adair Park,1751.998606033577
Mechanicsville,2124.5210477937094
Poncey-Highland,2224.0744589537453
Blandtown,1825.8608143136032
Marietta Street Artery,2118.098247073707
Westview,2349.3190729937896
Benteen Park,1813.0152128735986
Lenox,2095.618444553699
Capitol View,1886.8774211536252
Edgewood,1970.3738305136546
Lakewood,2082.7728431136948
Oakland City,2159.8464517537222
Atlantic Station,1793.746810713592
Vine City,2310.782268673776
Pittsburgh,2211.228857513741
Collier Heights,1928.62562583364
Midtown,2130.9438485137116
Lindridge/Martin Manor,2105.2526456337027
Custer/McDonough/Guice,1944.6826276336456
Knight Park/Howell Station,2076.3500423936925
Ardmore,1774.4784085535848
Berkeley Park,1816.2266132335997
Randall Mill,2227.285859313746
Loring Heights,2108.4640459937036
East Lake,1967.1624301536538
Grove Park,2018.5448359136717
Brookview Heights,1848.340616833611
The Villages at East Lake,2294.72526687377
Peoplestown,2195.1718557137347
Cascade Avenue/Road,1899.7230225936296
West End,2323.627870113781
Venetian Hills,2307.5708683137746
Chosewood Park,1925.4142254736387
Center Hill,1912.5686240336345
Kirkwood,2073.138642033691
Colonial Homes,1938.2598269136433
Buckhead Forest,1861.1862182736159
Pine Hills,2208.017457153739
Inman Park,2057.0816402336854
Sweet Auburn,2281.8796654337657
North Buckhead,2147.000850313718
Garden Hills,1996.0650333936637
Virginia Highland,2313.993669033777
Peachtree Park,2188.7490549937324
Brookwood Hills,1854.7634175536134
```

Construct full map

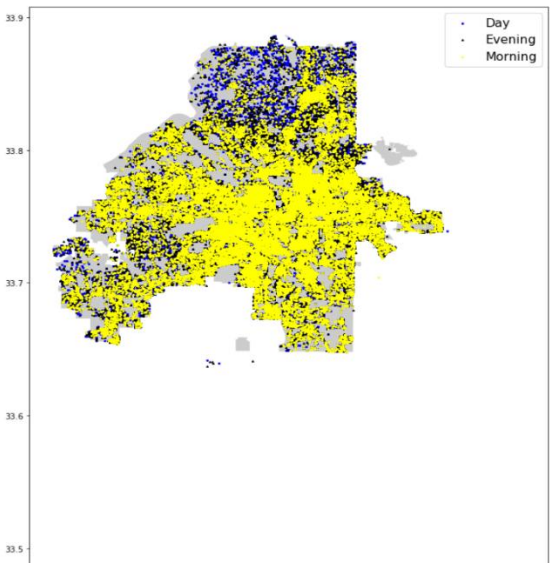
```
In [6]: def plot_map(sf, x_lim = None, y_lim = None, figsize = (11,9)):
            '''
            Plot map with lim coordinates
            '''
            plt.figure(figsize = figsize)
            id=0
            for shape in sf.shapeRecords():
                x = [i[0] for i in shape.shape.points[:]]
                y = [i[1] for i in shape.shape.points[:]]
                plt.plot(x, y, 'k')

                if (x_lim == None) & (y_lim == None):
                    x0 = np.mean(x)
                    y0 = np.mean(y)
                    plt.text(x0, y0, id, fontsize=10)
                id = id+1

            if (x_lim != None) & (y_lim != None):
                plt.xlim(x_lim)
                plt.ylim(y_lim)
        # Plot the map
        plot_map(sf)
```



Entry #17

Date: 4/15

Worked on beta II, aggregated crime scores as clustered points in each neighborhood according to the time of occurrence, plan on shading each neighborhood to account for heavy crime scores and add a legend this weekend.



Return to Table of Contents

Entry #18

Date: 4/27

Type of Algorithm Used vs Accuracy

	Root Mean Squared Error Value (dimensionless)	Accuracy (%)
Neighborhood 1		
Neighborhood 2		
Neighborhood 3		
Neighborhood 4		
…last Neighborhood (243)		

I decided to use RMSE instead of % difference for my data table, as it is better for data scientists in general in the context of machine learning. I got approval to do this from Mr. Nguyen in early January, as we technically have not learned this statistical analysis method in Biology. Same number of rows—one for each neighborhood. I also discovered that employing PCA dimensionality reduction had no statistically significant impact on the data. As a result, I have decided to switch to using K-means as a classifier method for adding features to the data. This would require me to utilize the Elbow Method to choose an optimal K-value and normalize the data per the Euclidean distance between the centroids. As a result, I will be using K-means for dimensionality reduction instead of as an algorithm to predict crime metrics, and since PCA dimensionality reduction was of no use, there is no need to incorporate it in the final charts (I got an RMSE value of zero for all of the values in the PCA table). Therefore, I will be using decision trees, random forests, and linear regression as my supervised algorithms and Naïve Bayes as an unsupervised algorithm. It may seem strange that I am using only one unsupervised algorithm compared to three supervised algorithms in this project; however, when you are forced to use one unsupervised algorithm for dimensionality reduction because the other one does nothing, you have to compromise, especially with large data sets such as these (300,000+ rows). I plan to finish Beta II before May.

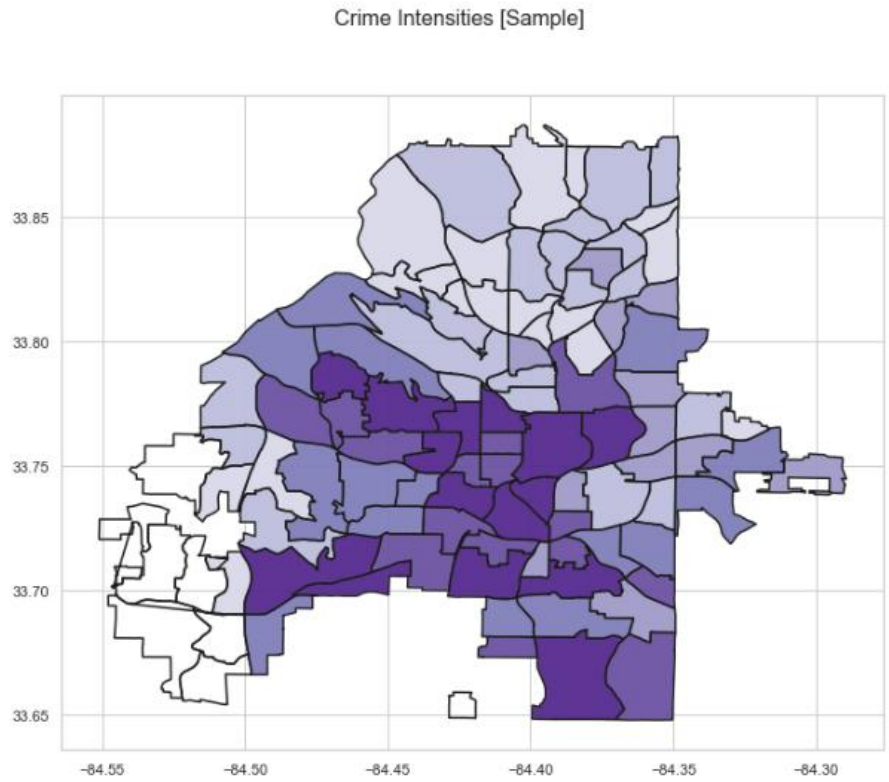
Entry #19

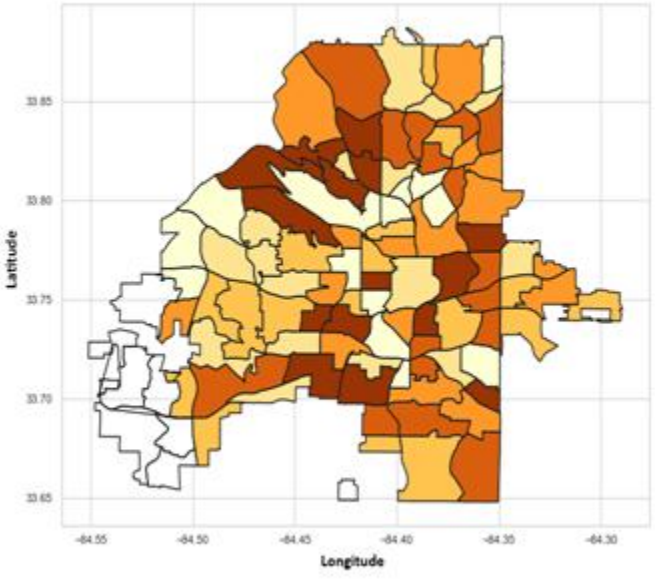
Date: 5/2

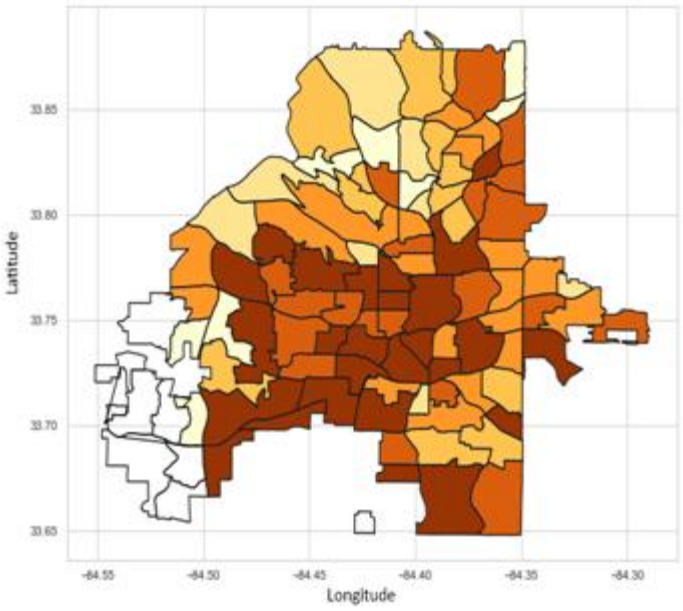
Finished beta II, altered final procedure and independent variable in accordance with addendum and fully completed the shapefile visualizations for the 2008-2019 predicted crime scores and the actual 2019 crime scores. The algorithm that I used for predicting the 2019 crime scores was the Naïve Bayesian classifier, as it had the greatest overall accuracy of about 63.9%. I also finished collecting ALL of my data. I plan to start working on my data analysis and CERA and finish my project by the end of this week.

Return to Table of Contents

<Figure size 792x648 with 0 Axes>







Return to Table of Contents

Entry #20

Date: 5/3

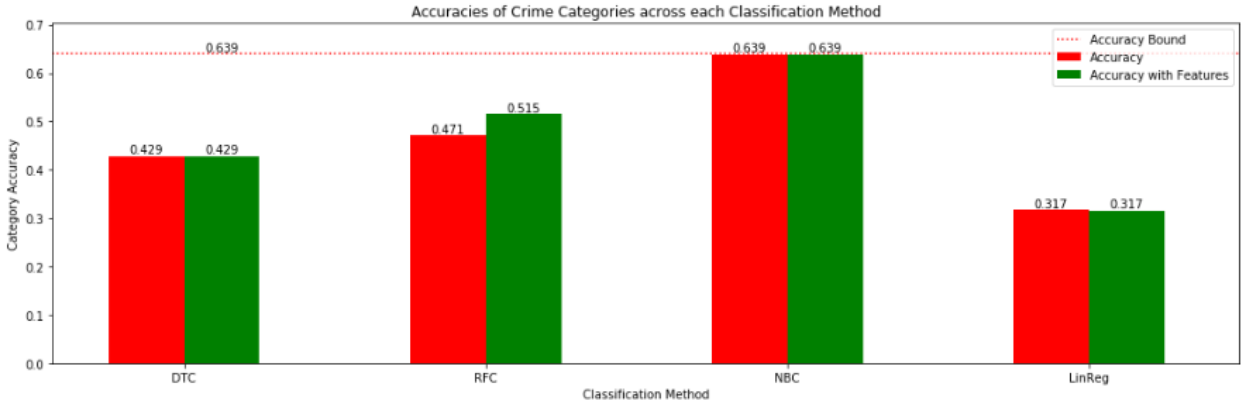
Worked on data analysis and CERA, imported data collected from processed algorithmic data frames in Excel and used the RMSE formula to create new columns, which I then exported as a new data frame into sci-kit learn to graph. I plan on coding the charts for the data visualizations in sci-kit learn, as Excel and Google Sheets probably will not represent the data well enough for this project, but python can.

Entry #21

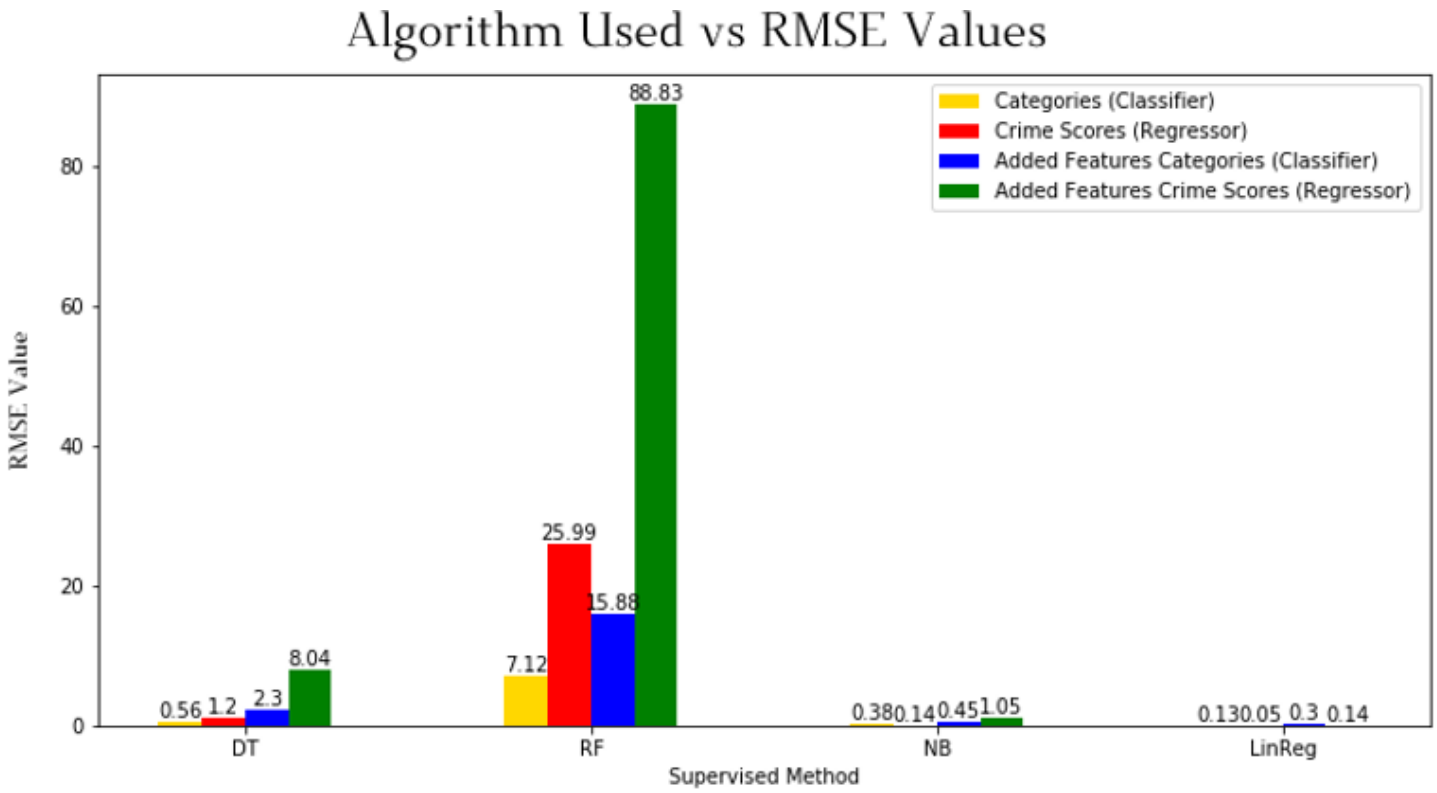
Date: 5/5

Finished the two graphs that I will primarily use for comparing the algorithms in this project. Also updated the GitHub repo. I plan on editing the graph axes with Photoshop a little bit because sci-kit learn isn't making the titles large enough for some weird reason (might be a scaling/resolution issue).

```
plt.margins(x=0, y=0.1, tight=True)
plt.legend(["Accuracy Bound", "Accuracy", "Accuracy with Features"])
plt.xlabel("Classification Method")
plt.ylabel("Category Accuracy")
plt.title("Accuracies of Crime Categories across each Classification Method")

# Precision Plot
plt.sca(ax1)
classNames1 = ["DTC", "DTC w/ Feats", "RFC", "RFC w/ Feats", "NBC", "NBC w/ Feats", "LinReg", "LinReg w/ Feats"]
plt.legend(["Category 1", "Category 2", "Category 3", "Category 4"])
xticks_locs = []
for j in range(len(classNames)):
    xticks_locs.append(j * 5)
    xticks_locs.append(j * 5 + 2)
plt.xticks(xticks_locs, classNames1)
plt.xlabel("Classification Method")
plt.ylabel("Category Precision")
plt.title("Precisions of Crime Categories across each Classification Method")

# Recall Plot
plt.sca(ax2)
classNames2 = ["DTC", "DTC w/ Feats", "RFC", "RFC w/ Feats", "NBC", "NBC w/ Feats", "LinReg", "LinReg w/ Feats"]
plt.legend(["Category 1", "Category 2", "Category 3", "Category 4"])
xticks_locs = []
for j in range(len(classNames)):
    xticks_locs.append(j * 5)
    xticks_locs.append(j * 5 + 2)
plt.xticks(xticks_locs, classNames2)
plt.xlabel("Classification Method")
plt.ylabel("Category Recall")
plt.title("Recalls of Crime Categories across each Classification Method")

plt.savefig('images/Supervised_Algs/apr_categories.png')
plt.show()
```



Return to Table of Contents



Entry #22

Date: 5/6

Finished data analysis and CERA; worked on the CERA part, imported all relevant tables and graphs, and made a summarized table in Excel using the RMSE formula which I then imported as my statistical test data.

Entry #23

Date: 5/7

Finished my poster; worked on the data analysis part, uploading graphs, purpose, conclusions, applications, etc., I plan to finish my project by the end of this weekend given that the project should be completed by then.

Entry #24

Date 5/8

Finished my engineering presentation, abstract, and final paper in a day; uploaded relevant graphs, talked about prior research, added conclusions, what the data means, what previous people have done, my engineering goal, hypotheses, copy/pasted all documents into final paper, etc. I plan to continue upon this work next year by getting neural networks involved or cross-referencing the prediction models with socioeconomic data sets for each neighborhood to add more regressors and make the predictions more accurate.

Return to Table of Contents

SciEng Research Project Proposal: Engineering

This form should be filled out cooperatively with all members of your group.

Today's date: 08/24/21

Team Leader Name: Anish Goyal

Member #2 Name:

Member #3 Name:

ALL ABOUT YOUR PROJECT IDEA...

Select the appropriate choice below:

[ ] Science: Solo	[ ] Science: Team	[x] Engineering: Solo	[ ] Engineering Team

Working Title of Project (65 characters or less, including spaces)

Polynomial Interpolation and K-Mean Mining to Predict Crime Rates

What is your research question or engineering goal?

My engineering goal is to take the locational data from the case spreadsheets of the Atlanta police department and use machine learning algorithms to predict which neighborhoods have a high crime potential, and also generate a report of the most common crimes in specific areas or city-wide. If possible, I would like to make the crime prediction process completely autonomous by having the user upload a data spreadsheet and have the algorithm make the calculations on the back end instead of them having to run the processes manually, as well as support for visual data plotting. This, hopefully, will allow other police departments to utilize my program and allow the country to better protect its citizens by dealing with crimes quickly and efficiently.

Return to Table of Contents

What kind of real-world application does this project have? What is the point of the project? This could, for example, be an improvement on an existing device or solution to a problem. *This question is very important, especially when your project is judged at the Fair.

In Atlanta, the city crime rate is 108% higher than the national average. 30,000 crimes occur annually with a 6.1% crime rate per capita, fostering it in the top 3% most dangerous cities in the United States. With such a high rate of crime, police officers need to preemptively be allocated to areas with a high predicted rate of crime, instead of dealing with matters on a case-by-case basis.

How do people currently solve this problem? This could serve as a control for your experiment by serving as a comparison that you are trying to improve against. You can cite existing designs or patents (recommended but not required for engineering).

Many people are currently developing ways to solve this problem, and some have even done so already, so what makes my idea novel? My method will be unique in that every crime type is given a "crime score." In order to distinguish which neighborhoods in Atlanta are crime-heavy, an aggregated score that accounts for the severity of specific crimes has to be given for each area. This will allow the program to epitomize the crime level of a neighborhood in one value, and allow the crime-heavy areas to be sorted by time much easier, which will allow the analysts in the police departments to determine what factors specifically may have increased the rate of crime in a specific neighborhood on a particular day.

How do you envision a control for your experiment? In engineering, you are usually comparing against an existing solution to the problem (for example, a physical device that you are comparing your device against or it can be data that you are trying to compare against). You should aim to make some improvement(s) against that solution.

Police response times and the number of crimes without the use of machine learning predictions.

Return to Table of Contents

SciEng Research Project Proposal: Engineering

This form should be filled out cooperatively with all members of your group.

Today's date: 09/02/21

Team Leader Name: Anish Goyal

ALL ABOUT YOUR PROJECT IDEA...

Select the appropriate choice below:

[ ] Science: Solo	[ ] Science: Team	[x] Engineering: Solo	[ ] Engineering Team

Working Title of Project (65 characters or less, including spaces)

Polynomial Interpolation and K-Mean Mining to Predict Crime Rates

What is your research question or engineering goal?

My engineering goal is to take the locational data from the case spreadsheets of the Atlanta police department and use machine learning algorithms to predict which neighborhoods have a high crime potential, and also generate a report of the most common crimes in specific areas or city-wide. If possible, I would like to make the crime prediction process completely autonomous by having the user upload a data spreadsheet and have the algorithm make the calculations on the back end instead of them having to run the processes manually, as well as support for visual data plotting. This, hopefully, will allow other police departments to utilize my program and allow the country to better protect its citizens by dealing with crimes quickly and efficiently.

Return to Table of Contents

What kind of real-world application does this project have? What is the point of the project? This could, for example, be an improvement on an existing device or solution to a problem. *This question is very important, especially when your project is judged at the Fair.

In Atlanta, the city crime rate is 108% higher than the national average. 30,000 crimes occur annually with a 6.1% crime rate per capita, fostering it in the top 3% most dangerous cities in the United States. With such a high rate of crime, police officers need to preemptively be allocated to areas with a high predicted rate of crime, instead of dealing with matters on a case-by-case basis.

How do people currently solve this problem? This could serve as a control for your experiment by serving as a comparison that you are trying to improve against. You can cite existing designs or patents (recommended but not required for engineering).

Many people are currently developing ways to solve this problem, and some have even done so already, so what makes my idea novel? My method will be unique in that every crime type is given a "crime score." In order to distinguish which neighborhoods in Atlanta are crime-heavy, an aggregated score that accounts for the severity of specific crimes has to be given for each area. This will allow the program to epitomize the crime level of a neighborhood in one value, and allow the crime-heavy areas to be sorted by time much easier, which will allow the analysts in the police departments to determine what factors specifically may have increased the rate of crime in a specific neighborhood on a particular day.

How do you envision a control for your experiment? In engineering, you are usually comparing against an existing solution to the problem (for example, a physical device that you are comparing your device against or it can be data that you are trying to compare against). You should aim to make some improvement(s) against that solution.

Police response times and the number of crimes without the use of machine learning predictions.

Return to Table of Contents

What is your independent variable? For engineering, this is typically multiple designs or approaches to solving your problem.

The number of k-clusters used in determining the rate of crime.

How will the independent variable be changed? What approaches might you try as you design? Be specific…what are the *conditions* on your IV?

The independent variable can be fully generated by the computer or manually set by the user. Manually setting the number of k-clusters (supervised learning) can produce accurate but imprecise data measurements, while a bad calibration for automatic k-mean cluster generation (unsupervised learning) can lead to precise but inaccurate predictions. I have to find what works best in determining the number of different k-mean clusters in the crime spreadsheets and how to get the computer to interpret the data based on those clusters.

What is the dependent variable(s)? What are your success criteria? There could be more than one for engineering. However, if you have more than one DV, you should decide whether your project must meet all the criteria or some of them or try to be an average of the different criteria.

The dependent variables are the crime predictions by location and the aggravated crime score of each neighborhood, which can be sorted within a date range.

How will the dependent variable(s) be measured? What tool, instrument, or probe? What units? (Be sure they are SI/metric.) For a list of probes you may use from GSMST, click here. Sometimes in engineering, these may not be measurements (they work or do not work), but you still need to decide how you will measure them (for example, take averages).

I can analyze crimes from previous years and see whether the machine's prediction and locational analysis were accurate for the future dates of that period. The dependent variable (aggregated crime score) will be measured by comparing the percent errors between the prediction outputs of each number of k-mean clusters generated and the real crime scores that occurred using historical data.

Return to Table of Contents

Will your project require special approvals? If yes, explain. Click on the links provided for detailed rules. (microbiology/biological agents, vertebrates, or hazardous materials or dangerous activities/devices)? **You cannot perform microbiology projects at home!** Be sure to read the engineering guide if you're doing an engineering project.

	Yes Explain:	
x	No	

You are financially responsible for the cost of your materials. Provide an educated estimate for what this cost might be. It should be reasonable!

Little to none.

List at least 5 research questions that will need to be answered before carrying out your project.

1.	How can I use the pandas python library to read data spreadsheets and cluster the data based on those values?
2.	What different types of crimes occur in the U.S. and which are considered the most severe?
3.	Is this idea already patented? Has my specific solution to the problem already been considered?
4.	What types of interpolation methods already exist in data science? Is polynomial interpolation the best method given the data the computer is analyzing?
5.	What is k-means clustering in data science and how can it assist me with predicting crime rates in Atlanta?

Return to Table of Contents

Research Plan Attachment (screenshot)

Form 1A Research Plan Attachment: Engineering

Team Leader Anish Goyal

Team Member(s) Anish Goyal

Title of Project Polynomial Interpolation and K-Mean Mining to Predict Crime Rates

Rationale

Atlanta has a 108% higher crime rate than the national average. It is within the top 3% most hazardous cities in the United States with approximately 30,000 crimes annually and a 61% crime rate per capita ("Atlanta crime rates," 2019). Given that Atlanta is such a high-profile city with its large number of crimes, the police department cannot respond to each case individually without wasting excess human resources. They must be led to high-crime locations ahead of time. Effective patrols in crime-heavy areas can be established when the best prediction model is used to determine the locations with the greatest crime rates. This can be applied to society because the police force can routinely deal with crimes before they ever occur on a daily basis, as their mere appearance is often sufficient to prohibit crimes from happening. This is where machine learning comes in. Using various machine learning algorithms on released data for past crime occurrences can provide valuable insight as to how crimes are distributed geographically, and which areas are the most crime heavy. Ultimately, as long as useful data is provided into the model—assuming that the best machine learning algorithm is selected, which is what I am trying to determine—the overall rate of crime will continue to fall.

Engineering Goal(s)

I plan to create a machine learning program that uses supervised and unsupervised algorithms coded in a python3 IDE called Anaconda Navigator. The purpose of the machine learning program is to predict and visualize the occurrences of different crime types for each neighborhood in Atlanta for the year 2019 after learning from crimes in the years 2009-2018 and comparing which algorithms are the most accurate with the actual crime counts from 2019 using the Atlanta PD's released COBRA datasets. The accuracy of the results will be a percentage measured by taking the average of the percent differences of each of the crime types for a neighborhood against the actual outcomes. This is needed because identifying clusters of high criminal activity permits the Atlanta PD to assign optimized police patrol routes and other crime-prevention measures like street cameras and neighborhood watches in the areas that need them the most, which allows the police department to efficiently allocate human resources while preemptively stopping large amounts of crime from occurring.

Return to Table of Contents

PROOF OF CONCEPT

Independent Variable

I will test each trial with a different type of machine learning algorithm that I deemed would be the most appropriate to use in this experiment given the information I retained from my literature check and prior knowledge. I will be using three supervised algorithms—K-means clustering, PCA (Principal Component Analysis) dimensionality reduction, and Naïve Bayes classifiers—and three unsupervised algorithms—decision trees, random forests, and linear regression—to predict the crime occurrence of each crime for each neighborhood and return an overall accuracy for that neighborhood using the prediction in comparison to the actual outcomes from 2019. Because my IV is categorical, it does not have units, but it is measured based on the accuracy it returns for the type of algorithm used.

Dependent Variable

- Crime occurrence (1st DV) is measured as a percentage and is defined as the predicted chance of a particular crime occurring in a particular neighborhood. It will be measured with multiple different supervised and unsupervised algorithms that will each return a different crime occurrence.
- Accuracy (2nd DV) is measured as a percentage and is defined as the total resemblance of the predicted crime occurrence (for the year 2019) with the actual crimes that occurred. This DV will be evaluated using the percent difference for the crime occurrences of each neighborhood, adding them, and taking the average. The percent differences for each individual neighborhood will likely also be measured and assessed.

If I am only allowed to report **one** dependent variable for this experiment, I would say the **accuracy** DV more or less takes precedence over the crime occurrence DV, as the crime occurrence DV is used for making internal predictions in the program and, overall, I am measuring the accuracy between the various algorithms that are applied to the data.

Controlled Variables/Constants

- Cleaned COBRA data sets
  - Time period of the csv file that will be used to train the model (2009-2018)
  - Time period of the csv file that will be used to analyze the predictions of the trained model in accordance with the type of algorithm used (2019)
- Shapefile
  - The shapefile that will be used to produce the visualizations of the crime scores or severities in each neighborhood across Atlanta will remain the same.
- Programming language used (python3)
  - Python is a very simple programming language that has many data science packages available to the user. I have extensive prior knowledge of Python in terms of data science have even used it before for robotics and other extracurriculars.
- Crime severity categorization (e.g., Categories 1-4)
  - Each crime category is associated with certain types of crimes, depending on

Return to Table of Contents

the severity. The types of crimes each category is associated with will remain constant during this experiment
- The computer running the program
  - Because my desktop PC has a high-performance graphics card, the runtime of the operations being performed are at a minimum. Running the program with computers that do not have a good GPU is probably not recommended
  - The computer being used to code the program, however, can vary. I will use my school computer, personal laptop, and desktop while making this project
- The IDE used
  - I used Anaconda Navigator (a python3 IDE catered for data scientists) in accordance with Jupyter Notebook to create this program.
  - I am unsure whether the IDE used will have an impact on whether the program runs successfully or not, so I added it as a constant just in case.
- Complete list of variables that will remain constant between groups.
- Python libraries
  - Should remain constant throughout the entire experiment, I will update if I add anything

Materials List

- Computer with python3 installed
- GitHub (to store project files as a repository)
- Jupyter Notebook (for data visualizations and testing)
- Anaconda Navigator IDE
- Excel (applying formulas to collected data and creating charts)
- 2009-2018 & 2019 COBRA data sets from the Atlanta PD
- Shapefile from the Atlanta Regional Commission for plotting visualizations
- Python libraries:
  - Sci-kit learn is a machine learning library containing many predictive algorithms, regression models, and classification trees.
  - Pandas is a library that adds compatibility for the use of CSV, JSON, and other data file types.
  - NumPy is an open-source library that adds complex mathematical functions to meet the needs of data scientists.
  - Matplotlib is a library that allows users to plot and embed graphs using data sets.
  - Seaborn is a plotting library that is very similar to matplotlib, but with some additional features and statistical integration.
- I/O (Input/Output)
  - Inputs (parameters being fed into the model)
    - Occur time
      - The exact time that a particular crime occurred
    - Day of week
      - The number of the day of the week that the crime occurred in
    - Month
      - The number of the month that the crime occurred in
    - Day of month
      - The day of the month that the crime occurred in

Return to Table of Contents

- Year
  - The day of the year that the crime occurred in since the year 2000
  - This parameter will only be passed to the 2009-2018 COBRA data set, as all of the data from the 2019 data set is from the same year and will only be used for comparing the results of the predictions from the previous years
- Latitude
  - The precise latitude where the crime occurred.
- Longitude
  - The precise longitude where the crime occurred.
  - Knowing both longitudinal and latitudinal values are useful for establishing possible predictors by location or for post-processing analysis
- Crime category
  - A number associated with the severity of the crime occurred.
  - Smaller numbers indicate a high severity, while bigger numbers indicate a minimal severity.
  - Homicide and manslaughter = 1
  - Aggravated assault, pedestrian robbery, commercial robbery, and residential robbery = 2
  - Residential burglary, nonresidential burglary, and auto theft = 3
  - Vehicular and nonvehicular larceny = 4
- Category 1*
  - The number of crimes that occurred in category 1 on a particular day
- Category 2*
  - The number of crimes that occurred in category 2 on a particular day
- Category 3*
  - The number of crimes that occurred in category 3 on a particular day
- Category 4*
  - The number of crimes that occurred in category 4 on a particular day

**** Inputs are for supervised algorithms only (i.e., decision trees or random forests)***

- Outputs (information being fed out of the model)
  - Visualization of crime occurrences
    - A plot of the crime scores for every neighborhood in Atlanta; darker shaded neighborhoods are more crime-heavy areas with a larger crime score
    - One visualization will be made for each crime category to see whether the distribution of particular crime types remain the same across multiple neighborhoods
    - The supervised/unsupervised algorithm with the greatest accuracy for 2019 will be visualized along with the actual crime occurrences from 2019
  - Predicted crime percentage
    - The predicted crime percentage for each neighborhood for the

Return to Table of Contents

year 2019
- The predicted crime percentage generated by each algorithm used will be compared to the actual 2019 crime results to test for accuracy

Procedures

These procedures are for one data cycle. Each data cycle will use a different prediction algorithm:

1. Import all important python libraries (sklearn, pandas, numpy, seaborn, and matplotlib)
2. Import 2009-2018 COBRA data set (csv file)
3. Preprocess the data into relevant columns, which will be parameters that will later be fed into the algorithm.
4. Further clean up the csv file by removing all nil values and replacing them with accurate representations
5. Create a simple table of the preprocessed data for documentation
6. Run PCA dimensional reduction to reduce size while retaining features
7. Run every algorithm, which will return a table
8. Use pandas to output the % difference for each algorithm for all of the neighborhoods into an Excel spreadsheet
9. Graph results
10. Use geopandas to plot the locational data of predicted crimes (longitude and latitude) into the Atlanta Regional Commission shapefile and shade in the areas of the bitmap appropriately according to crime occurrence
11. Use geopandas to plot the locational data of actual crimes into the shapefile and shade in the areas of the bitmap appropriately according to crime occurrence
12. Compare the visualizations and clean them if needed

Risk and Safety

This project involves no physical risks because it is a computer science project with no hazardous materials.

Sample Data Table

Type of Algorithm Used vs Accuracy

	% Difference (dimensionless)	Accuracy (%)
Neighborhood 1		
Neighborhood 2		
Neighborhood 3		
Neighborhood 4		
…last Neighborhood (243)		

Return to Table of Contents

Atlanta has 243 neighborhoods, so the table will be very long. A tentative solution for now is to save the tables for all of the algorithms in an Excel spreadsheet, and when the time comes to import the tables, I will include only ten of the neighborhoods in the final table but in equal subintervals (I will of course attach the file as well, or it will be available on the GitHub page).

Addendums since Proof of Concept to be included in Beta II (04/27/22) (from logbook):

Type of Algorithm Used vs Accuracy

	Root Mean Squared Error Value (dimensionless)	Accuracy (%)
Neighborhood 1		
Neighborhood 2		
Neighborhood 3		
Neighborhood 4		
…last Neighborhood (243)		

I decided to use RMSE instead of % difference, as it is better for data scientists in general in the context of machine learning. I got approval to do this from you in early January, as we technically have not learned this statistical analysis method in Biology. Same number of rows—one for each neighborhood.

Graph Description

I will make two graphs. Both of them will be bar graphs with two bar graphs per category—one bar with dimensionality reduction and one bar without dimensionality reduction. There is no need for error bars because matplotlib automatically adjusts the data once the regressors are applied; in fact, the data that I will be graphing itself shows the margin of error for all of the data calculated and processed previously, and the data that was calculated and processed previously will be plotted on the visualization graphs that involve the Atlanta shapefile with all of the neighborhoods on it and colored accordingly. For my first bar graph, the x-axis will be labeled “Classification Method,” and since the x-axis is categorical, it does not have units. The y-axis will be labeled “Category Precision,” and will be a percentage. The first bar graph essentially tells us how accurate the data was for each classification method with and without dimensionality reduction. The second bar graph will be based on the RMSE calculations from earlier on in the experiment. The x-axis will be the type of regression used, while the y-axis will be the RMSE calculated value, and both values are dimensionless. For each algorithm used in the second bar graph, there will be two bars just like the first bar graph—one bar with RMSE alone and one bar with RMSE and dimensionality reduction.

Addendums since Proof of Concept to be included in Beta II (04/27/22) (from logbook): Employing PCA dimensionality reduction had no statistically significant impact on the data. As a result, I have decided to switch to using K-means as a classifier method for adding features to the data. This would require me to utilize the Elbow Method to choose an optimal K-value and normalize the data per the Euclidean distance between the centroids. As a result, I will be using K-means for dimensionality reduction instead of as an algorithm to predict crime metrics, and since PCA dimensionality reduction was of no use, there is no need to incorporate it in the final charts (I got an RMSE value of zero for all of the values in the PCA table). Therefore, I will be using decision trees, random forests, and linear regression as my

Return to Table of Contents

supervised algorithms and Naïve Bayes as an unsupervised algorithm. It may seem strange that I am using only one unsupervised algorithm compared to three supervised algorithms in this project; however, when you are forced to use one unsupervised algorithm for dimensionality reduction because the other one does nothing, you have to compromise, especially with large data sets such as these (300,000+ rows).

Analysis of Results

After the COBRA data set has been preprocessed, fed into the various machine learning algorithms individually, and outputted its results into 6 different tables (one for each algorithm), I will use the RMSE equation to determine the overall accuracy of my program. The RMSE will be calculated for each neighborhood for each algorithm and compared to see which algorithm is the most efficient. My final data tables, all of which will include the results from the RMSE tests, will also include a final "average" row, which will take the average of all of the previous neighborhood RMSE results as an aggregated RMSE value for the algorithm being tested. Keep in mind that the RMSE result in the final data table is NOT the same as the RMSE value from the data collection table—you need the RMSE value to conduct an RMSE test, which will give you the RMSE result. To make the process of applying the RMSE tests efficient for each neighborhood for each algorithm, I will use Excel instead of Python for assisting me through this part of the project (Python lambdas are very CPU heavy even for relatively small tables like these). I might even use Excel for creating charts based off of the applied data values if matplotlib or seaborn cannot do it for me.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Where y_i is the predicted value of an i'th term,
$\hat{y}_i$ is the action value of an i'th term, and
n is the number of data points

Bibliography

Elite Data Science. (2017, May 16). *Modern Machine Learning Algorithms: Strengths and Weaknesses*. Retrieved December 3, 2021, from https://elitedatascience.com/machine-learning-algorithms

Return to Table of Contents

Kaloyanova, E., Ganchev, M., & Guide, S. (2020, March 10). *How to Combine PCA and K-means Clustering in Python?* 365 Data Science. Retrieved December 13, 2021, from https://365datascience.com/tutorials/python-tutorials/pca-k-means/

Kumar, V. (2021, July 2). *Naïve Bayes Algorithm overview explained.* TowardsMachineLearning. Retrieved December 14, 2021, from https://towardsmachinelearning.org/naive-bayes-algorithm/

Kumar, V. (2021, July 9). *Decision Tree Algorithm*. TowardsMachineLearning. Retrieved December 11, 2021, from https://towardsmachinelearning.org/decision-tree-algorithm/

Kumar, V. (2021, July 16). *Random Forest*. TowardsMachineLearning. Retrieved December 14, 2021, from https://towardsmachinelearning.org/random-forest/

Kumar, V. (2021, July 23). *K-Means*. TowardsMachineLearning. Retrieved December 11, 2021, from https://towardsmachinelearning.org/k-means/

Li, H. (2020). *Which machine learning algorithm should I use?* Retrieved December 3, 2021, from https://blogs.sas.com/content/subconsciousmusings/2020/12/09/machine-learning-algorithm-use/

Machine Learning Algorithms. (2021, August 13). Microsoft Azure. Retrieved December 14, 2021, from https://azure.microsoft.com/en-us/overview/machine-learning-algorithms/

Saini, A. (2021, September 16). *Naive Bayes Algorithm: A Complete guide for Data Science Enthusiasts*. Analytics Vidhya. Retrieved December 14, 2021, from https://www.analyticsvidhya.com/blog/2021/09/naive-bayes-algorithm-a-complete-guide-for-data-science-enthusiasts/

Return to Table of Contents

Saxena, S. (2019, October 15). *Mathematics Behind Machine Learning | Data Science*. Analytics Vidhya. Retrieved December 14, 2021, from https://www.analyticsvidhya.com/blog/2019/10/mathematics-behind-machine-learning/

Sayad, S. (2012, November 8). *Decision Tree - Regression*. SaedSayad. Retrieved December 3, 2021, from http://www.saedsayad.com/decision_tree_reg.htm

Stojiljković, M. (2019, April 15). *Linear Regression in Python – Real Python*. Real Python. Retrieved December 14, 2021, from https://realpython.com/linear-regression-in-python/

Towards Data Science. (2019, March 11). *Which machine learning model to use?* Towards Data Science. Retrieved December 4, 2021, from https://towardsdatascience.com/which-machine-learning-model-to-use-db5fdf37f3dd

Atlanta crime rates and statistics—NeighborhoodScout. (2019, September 30). Retrieved March 10, 2022, from https://www.neighborhoodscout.com/ga/atlanta/crime

BETA 1*

Independent Variable

Remained the same

Dependent Variable**

- My dependent variable will be a "predicted crime score" instead of a "predicted crime percentage" (or crime occurrence)
  - It is clear that I need to be measuring the total magnitude of all of the crimes that occur in a particular neighborhood (rather than assuming the crimes are equal or trying to predict the occurrence of a specific crime) because different crimes have different severities
  - Overall, measuring the accuracy of the predicted crime score in comparison to the actual crime score is much easier than working with percentages. It also expresses the crime level for a particular neighborhood in a single number, no matter what the time period or range may be
  - For information on how the predicted crime score is measured and what units

Return to Table of Contents

it will be in, see the 2nd bullet in "Materials List"

(Accuracy still takes precedence over predicted crime score)

Controlled Variables/Constants**

- Crime weighting
  - Depending on the crime category, the weight of each crime category when calculating crime scores remains constant.
  - Category 1 crimes are weighted by 1000x; category 2 by 100x; category 3 by 10x; and category 4 (larceny) by 1x

Materials List**

Addendums since Proof of Concept to be included in Beta I (03/29/22):

- The shapefile being used for the visualizations of this project will no longer be from the Atlanta Regional Commission (ARC). Instead, they will be .kml files sourced directly from the Atlanta PD website. The beat/patrol zone map can be found on the website by searching up "Zone (num) Beats"
  - The reason for this change is because I had to convert the ARC shapefile (originally in XML vector format) to a bitmap layout, which took lots of computing power to render a single image. Kml files are much easier to work with when using the Seaborn library
- Aggregated "crime score" (To be added to "outputs")
  - A sum of the crime category counts for a particular neighborhood for a particular day
  - Category 1 crimes are weighted by 1000x because they are the most severe; category 2 by 100x; category 3 by 10x; and category 4 (larceny) by 1x
  - The predicted crime score is a dimensionless scalar value, so it has no units

Procedures**

- Instead of using the Atlanta Regional Commission shapefile, I will use the beat region maps that are directly available on the Atlanta PD website.
- Instead of using crime occurrence as a metric for predicting future crimes, I will be using the predicted crime score.

* These will be developed throughout the project.
* Everything left unmentioned remained the same from the last prototype/iteration

BETA 2*

Independent Variable**

As stated in an addendum in my proof of concept, I removed PCA dimensionality reduction

Return to Table of Contents

for Beta II because it did nothing. Thus, I have to use K-means for dimensionality reduction instead of as a classifier, which brings the number of algorithms I am testing down to four from six. However, I will have a new independent variable, which is whether or not the data has clustered features added to it or not (basically whether K-means was applied to it or not). This independent variable will be shown in my final two graphs as second bar graph of a different color under the same algorithm category.

Dependent Variable

Remains the same

Controlled Variables/Constants

Remains the same

Materials List**

- Instead of calculating the percent difference of the expected crime score and the actual crime score, I will be using the RMSE (root mean square error) equation to determine how accurate the program was in predicting the crime results
    1. RMSE is better to use than percent difference in this scenario, as I can actually compare the predicted values to the actual results effectively
    2. It also helps me determine whether my predicted values are statistically significant in comparison to the actual values
    3. Look at analysis of results for more insight about RMSE and how I will analyze my data

Procedures

- Instead of doing percent difference for my statistical test, I will be using the RMSE equation
- There will be no principal component analysis dimensionality reduction
- Completed procedure:
    1. Import all important python libraries (sklearn, pandas, numpy, seaborn, and matplotlib)
    2. Import 2009-2018 COBRA data set (csv file)
    3. Preprocess the data into relevant columns, which will be parameters that will later be fed into the algorithm.
    4. Further clean up the csv file by removing all nil values and replacing them with accurate representations
    5. Create a simple table of the preprocessed data for documentation
    6. Run K-means clustering to reduce size while retaining features
    7. Run every algorithm, which will return a table for each algorithm, and

Return to Table of Contents

output them into an Excel spreadsheet using Pandas
8. Use Excel to output the RMSE values for each algorithm for all of the neighborhoods into a separate spreadsheet
9. Graph results as a bar graph
10. Use geopandas to plot the locational data of predicted crimes (longitude and latitude) into the Atlanta Regional Commission shapefile and shade in the areas of the bitmap appropriately according to crime occurrence
11. Use geopandas to plot the locational data of actual crimes into the shapefile and shade in the areas of the bitmap appropriately according to crime occurrence
12. Compare the visualizations and clean them if needed
13. Repeat steps 7-9 but without post-processing (K-means)

Return to Table of Contents

Research Question/Goal

What are you trying to accomplish?

I am trying to determine which machine learning algorithm has the most accuracy by making a program that can preemptively determine crime intensities in specific Atlanta neighborhoods and visualizing them.

Hypothesis

Experimental hypothesis: Decision trees provide the most accuracy when compared to other supervised and unsupervised machine learning algorithms and are easy to process and implement because they require very little computational power and process linear and non-linear data very quickly.
Null hypothesis: Decision trees do not provide the most accuracy when compared to other supervised and unsupervised machine learning algorithms and are difficult to process and implement.
Independent variables: Whether PCA was applied and the type of machine learning algorithm used
Dependent variables: Accuracy, crime severity category, RMSE value

Return to Table of Contents

Data

Date of collection: 05/02/2022

Decision trees	Predicted Crime Score	Actual Crime Score
Downtown	1951.105	2423
Greenbriar	2015.333	2214
Wildwood (NPU-C)	2362.165	2519
…Georgia Tech (243)	1999.276	2109

Random forests	Predicted Crime Score	Actual Crime Score
Downtown	2091.182	2423
Greenbriar	2509.444	2214
Wildwood (NPU-C)	2392.918	2519
…Georgia Tech (243)	1993.057	2109

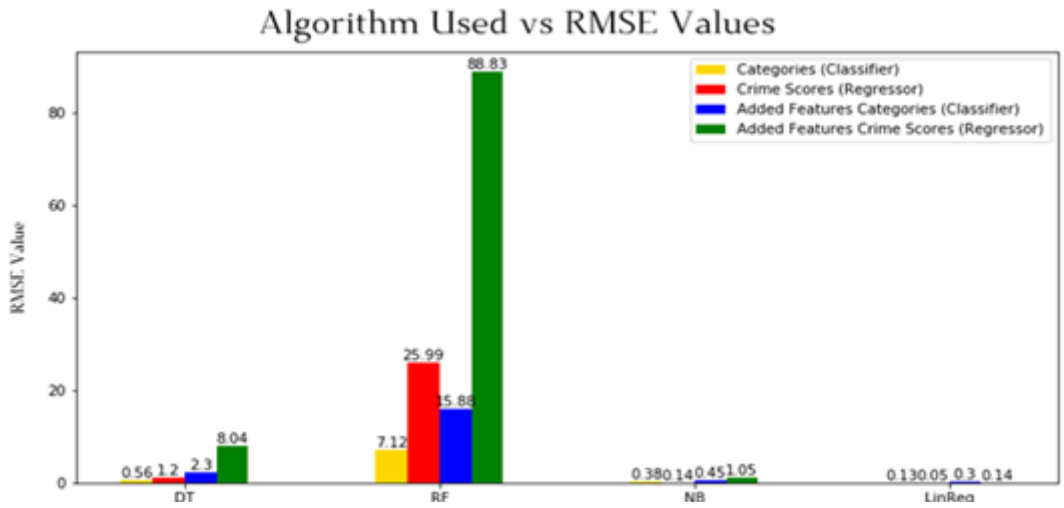
Naive Bayes	Predicted Crime Score	Actual Crime Score
Downtown	2355.491	2423
Greenbriar	2149.219	2214
Wildwood (NPU-C)	2391.773	2519

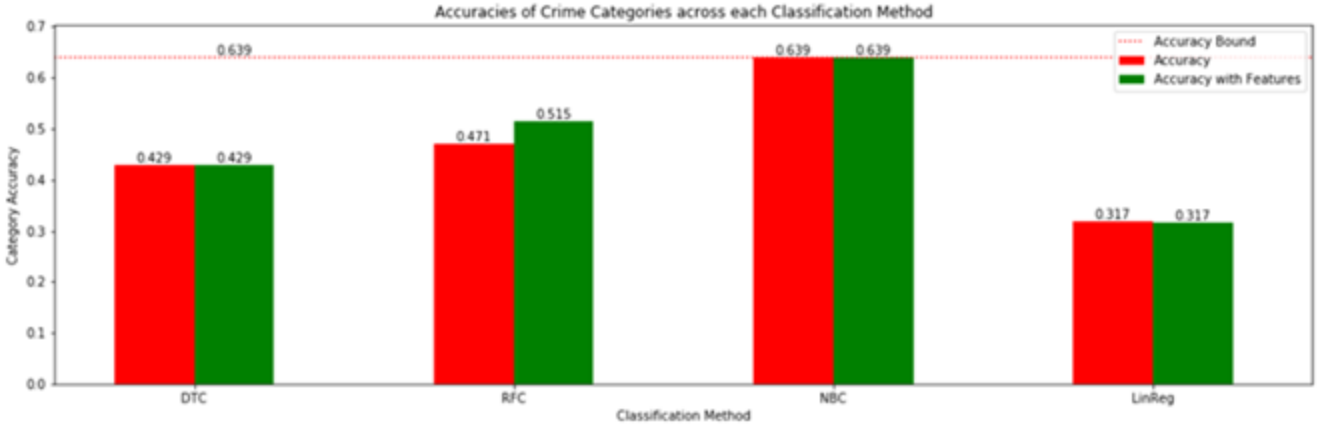
Return to Table of Contents

…Georgia Tech (243)	2003.838	2109

Linear Regression	**Predicted Crime Score**	**Actual Crime Score**
Downtown	2355.491	2423
Greenbriar	2149.219	2214
Wildwood (NPU-C)	2391.773	2519
…Georgia Tech (243)	2003.838	2109

Graphs





of Contents

Statistical Analysis

Type of statistical analysis (t-test, chi square test, ANOVA, etc): RMSE
Degrees of freedom: N/A
Critical Value: N/A
P Value: N/A

Summary statement:
The null hypothesis that decision trees do not provide the most accuracy when compared to other supervised and unsupervised machine learning algorithms and are difficult to process and implement failed to be rejected because decision trees were the third-most accurate algorithm behind random forests and Naive-Bayesian classifiers with an accuracy of 42.9% compared to 51.5% and 63.9% after PCA reduction, respectively. This shows that decision trees did not have the best accuracy in comparison to the other data sets.

Photo Documentation

Return to Table of Contents

Teacher Feedback Log

- [Date] Main takeaways
- [Date] Main takeaways

Return to Table of Contents