Checkpoint 4: Machine Learning

Team: The Powerful Turtles

Introduction

In this project, we would like to study how the demographics comparison between the complainants and the respective police officers reported correlating to the number of complaints. In this checkpoint, we break down the analysis from two different perspectives. The first part is to predict the number of complaints about each police officer based on the available police information. In the second part, we analyze the correlation between police demographics and complaint demographics to predict how such a correlation affects whether complaints take place.

Relational Analytics Questions

In this checkpoint, we would like to address the following questions:

- *Regression analysis to predict the number of complaints based on police information*
- *Predict the demographic difference interval between the officer and the complainant with the community that the two groups can live with each other with minimal complaints*

Results

Q1: Regression analysis to predict the number of complaints based on police information.
Colab notebook: https://colab.research.google.com/drive/1W-CW69jNw6j5Jm2hbW3H1UagJwPfaWM4?usp=sharing

In this part of the checkpoint, we attempt to predict the number of allegations for a given police officer. From the CPDP dataset, we extracted the following information for all officers: gender, race, rank, birth_year, complaint_percentile, civilian_allegation_percentile, honorable_mention_percentile, internal_allegation_percentile, trr_percentile, allegation_count, sustained_count, civilian_compliment_count, current_badge, current_salary discipline_count, honorable_mention_count, last_unit_id, major_award_count, trr_count, unsustained_count, has_unique_name. Since we would like to perform feature selection, we include features such as has_unique_name, which we do not expect to have a significant contribution to the prediction, to see whether we can eliminate such features from our model. We import the dataset as a Pandas dataframe.

First, we removed the rows where we do not have “allegation_count” information. Since there are entries with “allegation_count = 0,” we would not assume empty “allegation_count” would indicate zero allegation; also, since there is a significant amount of officers who do not have “allegation_count” information, simply setting the value to the

average would screw the data towards the mean value. Therefore, we removed these entries for better prediction.

We then split the dataset into three portions. The first one contains “allegation_count” as the y values for the analysis. For the rest of the data, we extracted “gender,” “race,” and “rank” as one group (g2), and set the rest as another (g1). We split these two data sets because g2 is a string dataset, and g1 is a numerical dataset. Both g1 and g2 have some missing values, and we took care of them separately.

In the g1 dataset, we used sklearn.impute.SimpleImputer to set the missing values as the mean values of the columns. From the machine learning course (Data_Sci 423), it was shown that imputing the missing values to the mean values can lead to significantly better prediction compared to not processing such entries. In the g2 dataset, because we need to make the string entries into numerical entries to feed into the machine learning model, we translate the dataset into a Spark dataframe to take advantage of the StringIndexer function. To compromise the unknown information about gender, race, and rank information for police officers in real situations, we replaced the missing values with “Unknown” as the entry. We then use StringIndexer function to translate string columns “gender,” “race,” and “rank” into numerical columns “gender_label,” “race_label,” and “rank_label” as the new g2 dataset. We combine g1 and g2 and take the overall dataset as x for machine learning. We have 20 features in x: birth_year, complaint_percentile, civilian_allegation_percentile, honorable_mention_percentile, internal_allegation_ percentile, trr_percentile, sustained_count, civilian_compliment_count, current_badge, current_salary, discipline_count, honorable_mention_count, last_unit_id, major_award_ count, trr_count, unsustained_count, has_unique_name, gender_label, race_label, and rank_label.

We performed an 80-20 split using sklearn.model_selection.train_test_split to obtain our training and test datasets. We use sklearn.linear_model.LinearRegression to fit complete x training data. Such fitting returns an r2 value of 0.9341 and root mean squared error of 2.726 for the training dataset, and an r2 value of 0.9343 and root mean squared error of 2.797 for the test dataset. These metrics look good for machine learning, but we might be overfitting the data by assigning features that are not significantly influencing the model with coefficients in the regression. Therefore, we performed a statistical analysis on the fitting. Table 1 shows the results. Parameter #0 is the y-intercept and the twenty other parameters are the twenty features in x. Typically, a p-value greater than 0.05 is not considered as statically significant. From the results, we can see that several entities have high p-values, including parameter #17 which is “has_unique_name” which we expected not to play a role in the model.

Table 1. Statistical analysis of the fitting

	Coefficients	Standard Errors	t values	p values
0	-9.4546	3.103	-3.047	0.002
1	0.0029	0.002	1.847	0.065
2	0.0171	0.001	16.699	0.000
3	0.0428	0.001	41.022	0.000
4	0.0157	0.001	17.595	0.000
5	0.0069	0.001	11.053	0.000
6	-0.0159	0.001	-17.673	0.000
7	1.0689	0.046	23.391	0.000
8	0.0479	0.007	7.055	0.000
9	-0.0000	0.000	-2.905	0.004
10	0.0000	0.000	15.101	0.000
11	-0.0140	0.050	-0.283	0.777
12	0.0355	0.001	31.734	0.000
13	0.0027	0.000	10.400	0.000
14	0.1560	0.049	3.200	0.001
15	0.2277	0.005	42.435	0.000
16	1.3498	0.004	371.897	0.000
17	-0.0255	0.050	-0.513	0.608
18	0.2680	0.046	5.765	0.000
19	-0.1178	0.021	-5.689	0.000
20	-0.0096	0.005	-2.080	0.037

With the p-value information about the features, we can select only the significant features into the model. Thus, we filtered to only include features with p-values less than 0.0001, which is the value that is very statistically significant for a statistical model (represented by four stars **** in the literature). Using this filter, we have fourteen features left: 'complaint_percentile', 'civilian_allegation_percentile', 'honorable_mention_percentile', 'internal_allegation_percentile', 'trr_percentile', 'sustained_count', 'civilian_compliment_count', 'current_salary', 'honorable_mention_count', 'last_unit_id', 'trr_count', 'unsustained_count', 'gender_label', and 'race_label.'

By performing linear regression on this selected feature set again, we found that we obtained a model that has a similar prediction performance with the complete feature set (Table 2). We confirmed that all features in this model are statistically significant with p-values close to 0 (results shown in the Colab notebook). By plotting the y-actual vs. y-predicted (Figure 1), a mostly linear relationship was obtained. Therefore, we conclude that using this 14-feature model, we can predict the number of allegations for an officer with excellent accuracy.

Table 2. Performance Metrics of the models

	Complete Features (20 features)		Selected Features (14 features)	
	Training	Test	Training	Test
R^2	0.9341	0.9343	0.9340	0.9342
Mean square error	7.4311	7.8212	7.4389	7.8385
Root mean square error	2.7260	2.7967	2.7274	2.7998

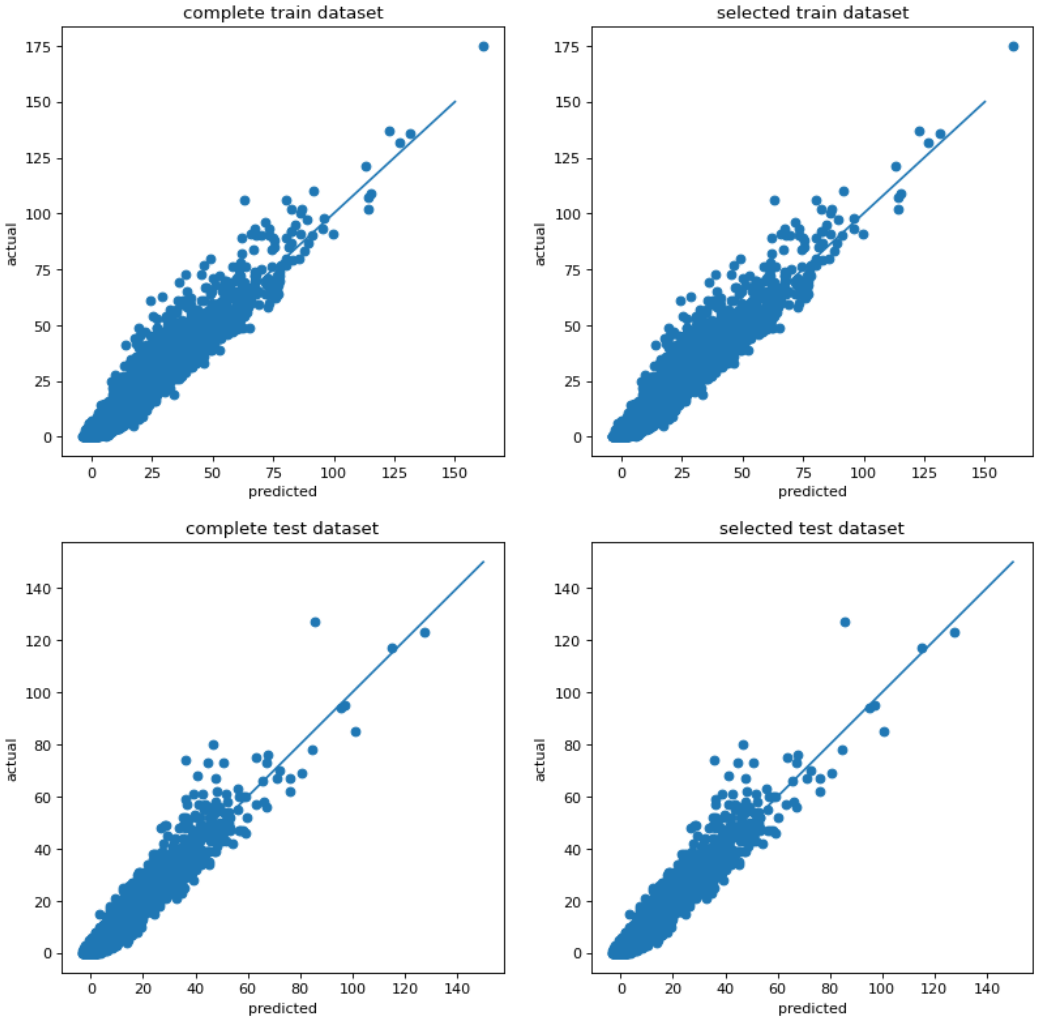


Figure 1. Actual vs. Predicted plot of the models.

Q2:*Predict the maximal demographic difference between the officer and the complainant with the community that the two groups can live with each other with minimal complaints*
Colab notebook:
https://colab.research.google.com/drive/1zQyuDwcFXEiZ2HKkPA3S1eWnSTPecCwX?usp=sharing

From the previous interactive visualization assignment, as shown in Figure.1, we found out that there was an empty space in each group. Assuming that the empty space has no allegations, in this part of the checkpoint, we want to find a percentage range in each race group that promises the lowest allegations.

Algorithms:

Select the center of the circle (i.e., x and y) and radius (r) as random float values between 0 to 1. Calculate the distance between each point to the center and check if the point is in the circle. If the distance is less than the radius, the point is in the circle and the number adds 1. After each round, the model calculates the accuracy, which is defined as:

$$Accuracy = Number\ of\ points\ out\ of\ the\ circle\ /\ all\ number\ of\ points$$

For each round, the model would save x, y, r, and accuracy. The next round repeats the same thing —— randomly choose x,y, and r, calculate the distance, and compute the accuracy. If the new r is larger than the previous r, and the accuracy is higher, update the final results. If the new r is larger than the previous r, but the accuracy is lower, save both parameters. The round per experiment was set to 10000. Finally, the model would find the optimal center and radius.

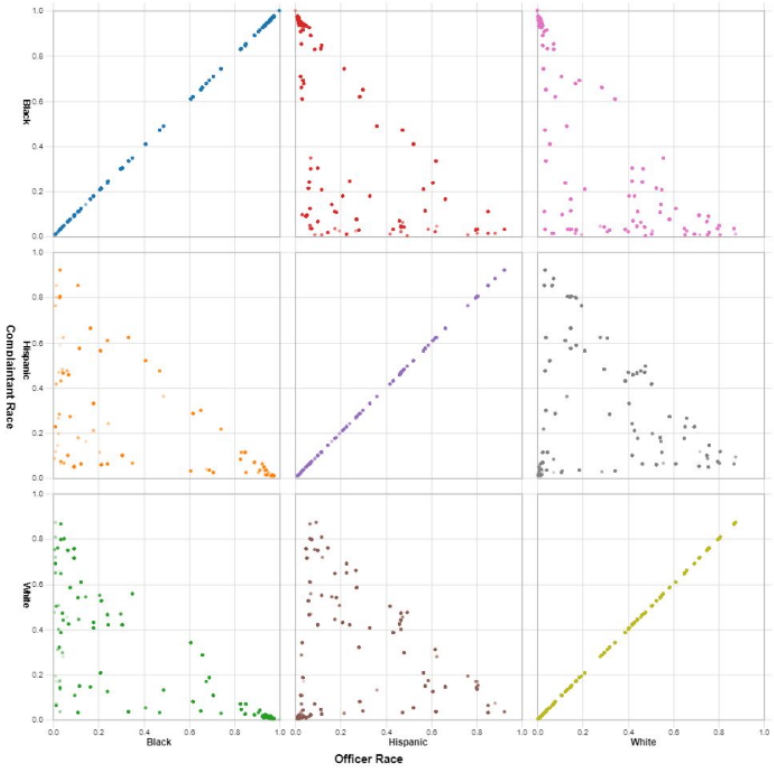
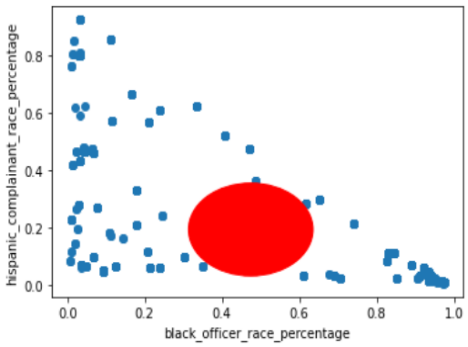
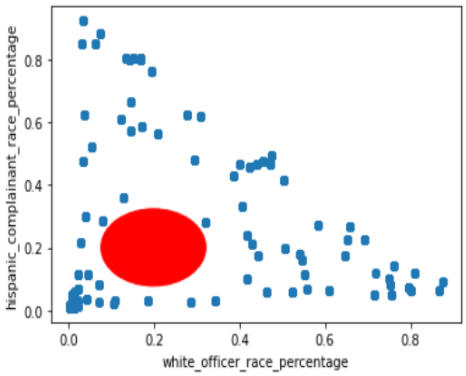


Figure 1. The relationship between officer race percentage vs complainant race percentage in the community

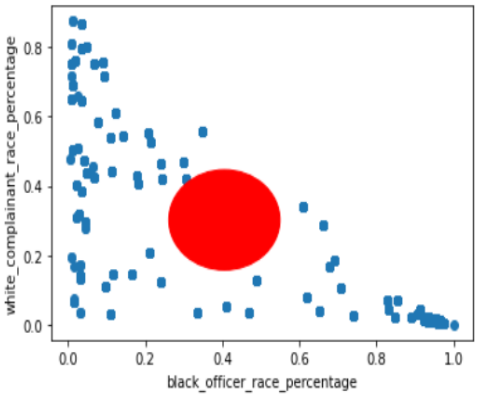
a. The white officer race percentage vs. Hispanic complainant race percentage
Center of the circle: [0.475, 0.194] , radius: 0.161



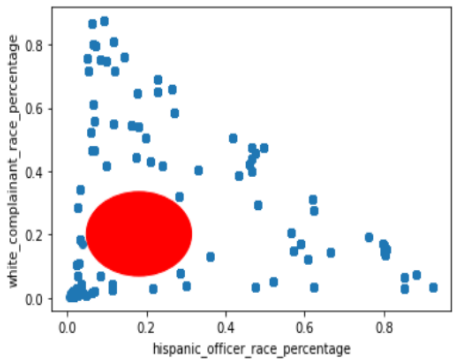
b. The black officer race percentage vs. Hispanic complainant race percentage
Center of the circle: [0.199, 0.201] , radius: 0.123



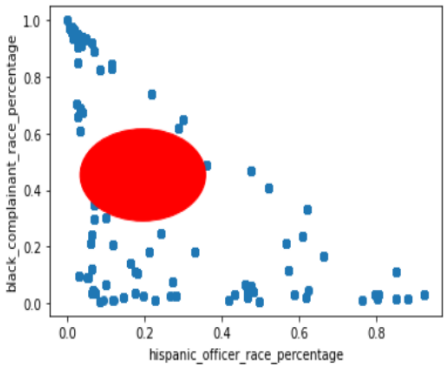
c. The black officer race percentage vs. white complainant race percentage
Center of the circle: [0.405, 0.302] , radius: 0.144



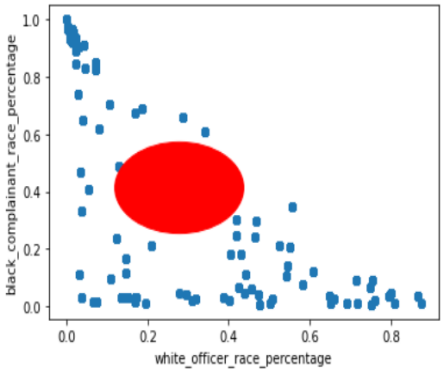
d. The Hispanic officer race percentage vs. white complainant race percentage
Center of the circle: [0.182, 0.202] , radius: 0.133



e. The Hispanic officer race percentage vs. black complainant race percentage
Center of the circle: [0.195, 0.453] , radius: 0.163



f. The white officer race percentage vs. black complainant race percentage
Center of the circle: [0.278, 0.413] , radius: 0.159



Analysis:

We want to calculate the best race percentage that is better for both officers and people to live in that community. For example, in Case f, the best percentage for officers in white is 0.278 and for complainants in black is 0.413, which means that if a new white officer comes, these two percentages are a good reference for him to find the best community to serve. The same with other groups.

Next, we want to calculate the overlapping areas among these six regions and see if a percentage could exist for all races. As shown in Figure 2, there are no completely overlapping areas for all the regions, but some regions with high overlaps. The result shows that different racial groups have different requirements. There is no fixed value that could be applied to all races. However, the officer race percentage range

between 0.25~0.35 and the complainant race percentage around 0.3 is the right choice for most communities.

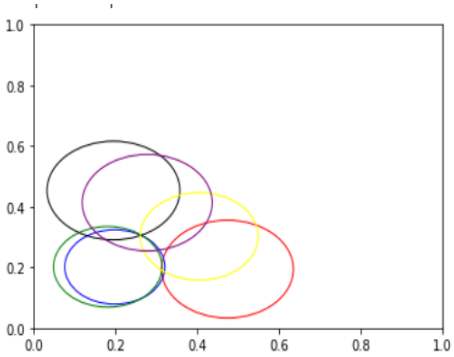


Figure 2. The overlapping areas for six regions