This checkpoint was useful in pushing our research forward in understanding new perspectives and findings for our central theme. For your reference, here is an overview of our overall theme:

We are interested in investigating the trends between the seniority of an investigator in terms of their age and rank, and their history of complaints. The focus of this investigation is twofold: First, how does the history of complaints correlate with each investigator over time? For example, as an investigator rises in the ranks, does the rate of complaints increase for this officer? Similarly, is there a trend in the type of complaint that the investigator receives? Second, we can understand potential biases that the officers might hold against their victims.

Additionally, investigating this topic may uncover related trends that may not necessarily be confined to the scope of this topic, but are interesting and important nonetheless. For example, trends between officers that rise in rank and their complaint volume might reveal some information about how the Chicago PD operates as an institution. That is, an officer may receive complaints from the public after an incident but may be awarded or promoted internally but the institution.

We had 4 research questions in place to develop our understanding further:

1. Based on how many years an officer has worked in the force, what the most commonly targeted racial demographic group was per officer, the amount of arrests an officer made, and the race and rank of the officer, to what degree can we accurately predict how many allegations an officer can expect to receive?
2. Using the same parameters as question 1 but this time including amount of allegations per officer, to what degree can we accurately predict the most commonly targeted racial demographic group per officer to predict bias?
3. Using text analytics, we want to label/categorize documents based on select keywords, and find the total amount of allegations that are labeled as racial slurs to see if we can connect or correlate anything back to our initial approach to find racist officers.
4. Of the amount of racial slur allegations found in question 3, how many of them involved weapons or assault? The reason for this is to retrieve more supporting evidence around our initial claims for inherent racism and bias in officers.

Question 1:

To reiterate, question 1 is as follows: Based on how many years an officer has worked in the force, what the most commonly targeted racial demographic group was per officer, the amount of arrests an officer made, and the race and rank of the officer, to what degree can we accurately predict how many allegations an officer can expect to receive?

The feature set is a table containing officer years worked, most common racial group label per officer, total officer arrests, officer race, and officer rank. The target set to train on is total officer allegations. Before running a model, we had to clean and prepare the data so that the models would run properly as well as give meaningful results. We removed all null and NaN values from the dataset as well as removing the officer ID column since it does not actually contribute meaningful data.

We then had to turn string values such as ‘White’, ‘Black’, ‘Hispanic/Latino’, ‘Commander’, ‘Sergeant’, etc. into numerical values. To do this we made a mapping schema that converts strings for officer and victim races as well as officer rank into assigned values (as shown below).

Mappings to "Numerify" the Data

```
victim_race_labs = [
    (0, 'Asian/Pacific Islander'),
    (1, 'Hispanic'),
    (2, 'White'),
    (3, 'Native American/Alaskan Native'),
    (4, 'Black')
]

officer_race_labs = [
    (0, 'Asian/Pacific'),
    (1, 'Hispanic'),
    (2, 'White'),
    (3, 'Native American/Alaskan Native'),
    (4, 'Black')
]

or_list = []
for i, rank in enumerate(officer_ranking_list):
    or_list.append((i, rank[0]))

or_list.append((16, 'Assistant Deputy Superintendent'))
print(or_list)
```

```
[(0, 'Lieutenant'), (1, 'Assistant Superintendent'), (2, 'Sergeant'), (3, 'Commander'), (4, 'Police Officer'), (5, 'D
etective'), (6, 'Superintendent Of Police'), (7, 'Other'), (8, 'Deputy Chief'), (9, 'Director Of Caps'), (10, 'Chie
f'), (11, "Superintendent'S Chief Of Staff"), (12, 'Captain'), (13, 'Field Training Officer'), (14, 'Deputy Superinte
ndent'), (15, 'First Deputy Superintendent'), (16, 'Assistant Deputy Superintendent')]
```

We then made a function called ‘Numerify’ which maps feature values to numerical values based on the schema above.

Now that the feature set is all ready to go, we ran it through our SVM model. The results can be seen below:

```
Results for predicting amount of allegations per officer:
---------------------------------------------------------
F1 score:          0.9525559658566799
Accuracy Score:    0.9365466748017084
```

As you can see, the model has a fairly high accuracy rate and f1 score. We had an f1 score of 95.3% and an accuracy score of 93.7%. This means that the model did a good job reliably predicting officer allegation counts based on our features. While this is a great system to be able to model, a caveat is that the model is only as good as its data. Having said that, we feel confident that based on the parameters mentioned above, we can accurately predict how many allegations an officer can expect to have over their career. We tried several machine learning models such as Logistic Regression, K Nearest Neighbors, and even K means clustering. However, SVM proved to be the highest performing model to make accurate predictions.

Question 2:

To reiterate, question 2 is as follows: Using the same parameters as question 1 but this time including amount of allegations per officer, to what degree can we accurately predict the most commonly targeted racial demographic group per officer to predict bias?

The purpose of this investigation was really about understanding bias from a predictive perspective. To try to uncover if there were any predictable patterns that could be used to gauge racial discrimination. We followed a very similar data cleaning and preparation protocol to question 1. However, the feature set for this question has a table including officer years worked, total office allegations, total officer arrests, officer race, and officer rank. The target set was the most common racial group label per officer. This is to say that whichever racial group had the most allegations against an officer, was the most common racial demographic group for that officer. We acknowledge this may not be the best way to understand bias, but it is certainly a start.

We had to use the same mappings and 'Numerify' function to prepare the data properly before running our model. Once the data was ready to go, we ran our model. The results are shown below:

```
Results for predicting most commonly targeted racial group per officer:
-----------------------------------------------------------------------
F1 score:         0.7750373692077728
Accuracy Score:   0.6327028676021964
```

As you can see, the results for this model performed more poorly than that of question 1. We had an f1 score of 77.5% and an accuracy score of 63.2%. We again tried several models and although none of them reached above around 75%, SVM seemed again to perform the best. Our take on this is that it is much more complicated a question to try to predict something like bias. We don't think that the data serves as a good predictor for bias especially since many of the offices might not have a strong bias at all. These results force all officers to have one main racial group as their bias label which needs further investigation to perhaps create a more robust dataset for predicting. Moreover, new approaches must be taken into consideration as to how to label a bias group per officer as it could be by chance that an officer has one group labeled for them much higher than others.

The results from what we did see, however, did seem to show a trend in concurrence with our past investigations. The most commonly predicted label for an officer in our model was 4 ('Black' racial group).

```
Label for most commonly targeted racial group predictions:
 [4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4. 4.]
```

Question 3:

The first step we took to approach this question was to see if we could build a learner to help predict labeled categories in the document_tags table. Our idea here was to count the frequency of the “racial_slurs” label. However, because the data is sparse, and the labels are few and not accurate, the max frequency of “racial_slurs” we could find based on fitting a learner, and just from the initial tabel was 7 out of a total of 1220 documents (filtered from 1683 as we took out the values where document_text = null). Thus, we set out to go for a different approach, and ended up writing a script that scans through each document, looking for a set of keywords, and categorizing each document based on the keywords. An example of the output from our script can be seen here:

```
keyword_counts = find_racism(text_frame)
print(keyword_counts)
```

```
lleges that a uniformed male white officer stated to him "Get your black ass out the playground, Nigger, when someone tells your ass to leave, you leave" a) 9 5 LAD. LOCATION - I.A.D
 1. Rule 2 Prohibits any action or conduct which impedes the Department?s efforts to achieve its policy and goals or brings discredit upon the Department. 2. Rule 6 Prohibits disobed:


n the day and time in the allegations he was at his car, parked on 82nd Street, just west of Stony Island Avenue, retrieving items from the trunk of his car for a workshop that was be
imately 7:30PM, on the 6600 block of South Sangamon, Travie Mitchell (a 15-year old juvenile at that time) was walking down the street with his cousins Sean Mitchell and Keenan Davis,
```

The output here is a fraction of the total results, but here we can clearly see that based on keywords, it has found documents that should obviously belong in the “racial_slur” category.

```
keyword_counts = find_racism(text_frame)
print(keyword_counts)
```

```
44
```

Based on the list of keywords that we used (taken off wikipedia, most words are highly offensive), our script was able to detect 44 complaints that should have been labeled as “racial_slurs”. Based on this number, we see that only about 4% of documents out of the 1220 complaints contain some form of racist content in them.

It should be noted that this is only based upon keywords, which does not account for racism of other forms, but based on the percentage alone, it seems that there were only about 44 incidents where officers were accused of using racial slurs against the complainants, which further supports and narrows down our initial research into finding inherent racism in officers, and in the police force.

Question 4:
The main idea of this question was to find the number of documents that are labeled as both assault, and the use of racial slurs. The reasoning behind this is to see if officers that use racial slurs in their complaints are also violent with their victim. The first approach that we took here was to just scan the documents and label documents that include assault of any kind (weapon, sexual, fists, etc). From our results, we find that 575 of the 1220 documents are categorized as some sort of use of violence, which is roughly 47%.

```
print(find_assault(text_frame))
```

```
575
```

This was just to get a general idea for the fraction of assault complaints. What we aim to find here is the fraction of allegations that contain violence as a part of the allegations that contain racial slurs. Thus, we find that with 44 total documents listed as “racial slurs”, 30 of them also contain some sort of violence. Here is the output and the list of keywords we used to search this:

Question 4

```
#here we employ the same keyword searcher, but account for weapons and assault too, the only difference is to account for a logical "and"
def find_assault(doc_text_column):
  counter1 = 0
  list_of_keywords = ['slur','assault','taser','weapon','punch','kick','hit','shoot','fire']
  for text in doc_text_column:
    for doc in text:
      words = doc.split(" ")
      counter = 0
      for w in words:
        for keyword in list_of_keywords:
          if w.lower() == keyword:
  #           print("doc: ", doc) #if you want to get the corresponding document
            counter += 1
          else:
            pass
    if counter >= 1:
      counter1 += 1
  return counter1
```

```
print(a_counts)
```

```
30
```

Thus, we see that roughly 68% of the allegations listed as “racial_slurs” contain some form of violence in them as well (30/44). This is interesting to us because not only does it help support our initial hypothesis about racism and inherent bias within officers, but it also provides support to the idea that officers that are racist are more likely to be violent with their victims - a very interesting area of future research.