aCS506 Team E Police Overtime Team Weekly Scrum Report

1. What we accomplished this week:

- Analyze the discovered topics in a deeper level and detailed exploration
- Analyze the earnings by pay range (the % of officers' pay range is from $50k-$70k, $70k - $100k, $100k-$200k, >$200k). Compare this % with the average pay range of other states.
- Connect the dots, try to understand what are the insights that the data conveys
- Explore and analyze overtime datasets
- Investigate whether certain demographic characteristics such as age, gender, race, tenure and rank influence the worked-to-paid ratios of police officers. This can shed light on potential disparities in compensation within the force
- Pie chart of each category of the earnings for from 2011-2022

  (a) Deliverable Links:
  - https://colab.research.google.com/drive/1gVfObsV1cbzK5XpTjjA9xc1m9cBou1LY
  -

    [Links to New Project deliverables for this week, if relevant*]
    **specify a specific page or segment for the deliverables for a given week – please make them easy to find!*

2. Individual team member updates:

[Each team member should give a summary of their personal contributions]

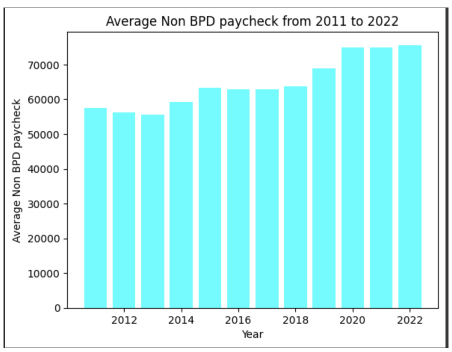
[Nurassyl Medeu]

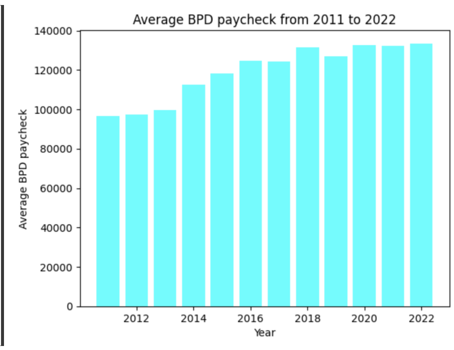
- Overtime data preprocessing
  - Bringing 2013 data to the same format as the rest

```python

o_2013["City/State/Zip"] = o_2013["City/State/Zip"].apply(clean)

clean_list = []
for row in range(o_2013.shape[0]):
    temp = o_2013["City/State/Zip"][row].split(' ')
    if len(temp) == 6:
        print(o_2013["City/State/Zip"][row])
    if len(temp) == 4:
        temp = [temp[0] + ' ' + temp[1]] + temp[2:]
    if len(temp) == 5:
        merged_string = ' '.join(temp[:3])
        temp = [merged_string] + temp[3:]

    clean_list.append(temp)

df_temp = pd.DataFrame(clean_list, columns=['City', 'State', 'Zip'])
df_temp['Zip'] = df_temp['Zip'].astype(str)


o_2013[['City', 'State', 'Zip']] = df_temp
# Dropping the original column 'City/State/Zip'
o_2013.drop('City/State/Zip', axis=1, inplace=True)

o_2013.to_csv('Overtime data/csv/details-2013(2).csv', index=False, quoting=1)

```

```python
import pandas as pd

o_2013 = pd.read_csv("Overtime data/csv/details-2013.csv")


o_2013.insert(5, 'XStreet', '')
o_2013.insert(15, 'Customer Address 1', '')
o_2013.insert(16, 'Customer Address 3', '')
o_2013.insert(17, 'City', '')
o_2013.insert(18, 'State', '')
o_2013.insert(19, 'Zip', '')

# remove all commas


def clean(value):
    if "SO.BOSTON" in value:
        value = value.replace("SO.BOSTON", 'SOUTH BOSTON')
    if "NO.QUINCY" in value:
        value = value.replace("NO.QUINCY", 'NORTH QUINCY')
    if "E.BRIDGEWATER" in value:
        value = value.replace("E.BRIDGEWATER", 'EAST BRIDGEWATER')
    if "E BRIDGEWATER" in value:
        value = value.replace("E BRIDGEWATER", 'EAST BRIDGEWATER')
    if "N.OXFORD" in value:
        value = value.replace("NO.OXFORD", 'NORTH OXFORD')
    if "HUDSON N H" in value:
        value = value.replace("HUDSON N H", 'HUDSON NH')
    if "MASS02124" in value:
        value = value.replace("MASS02124", 'MA 02124')
    if 'N.Y.' in value:
        value = value.replace('N.Y.', 'NY')
    if 'NEW YORK, NEW YORK 10017' in value:
        value = value.replace('NEW YORK, NEW YORK 10017', 'NEW YORK NY 10017')
    if 'NH03874' in value:
        value = value.replace('NH03874', 'NH 03874')
    if 'MA02116' in value:
        value = value.replace('MA02116', 'MA 02116')
    if ' - ' in value:
        value = value.replace(' - ', '-')
    if 'MA02138' in value:
        value = value.replace('MA02138', 'MA 02138')
    if ' PHONE X #355' in value:
        value = value.replace(' PHONE X #355', '')

    return value.replace(',', ' ').replace('.', ' ').replace('  ', ' ')

```

- This week I did initial analysis of the average BPD and non BPD paychecks over the years of 2011-2022





-
- It is clearly seen from the data that the average non BPD paycheck has always been significantly higher than the non BPD paycheck. Even then, both graphs seem to have an upward trend, with BPD almost reaching 140000$ and non BPD barely passing 70000$ mark in 2022, which is 2x difference. In further analysis we will decide if this difference justifies itself by looking at further data about BPD.

[Riva Sun]

- Pie chart of each category of the earnings for from 2011-2022
- For example:

- By observation, Regular earning had the most percentage among all the years. The proportion of overtime earnings are increasing, leading to the increasing total earnings.
- But the percentage of each category is almost the same each year and there is no significant sudden change.

[Truc Duong]

- Discern inconsistencies in information across datasets from various years, as well as disparities between the supplied dataset and the client's directive questions.
- Analyze Overtime Pay type

Overtime Type Analysis

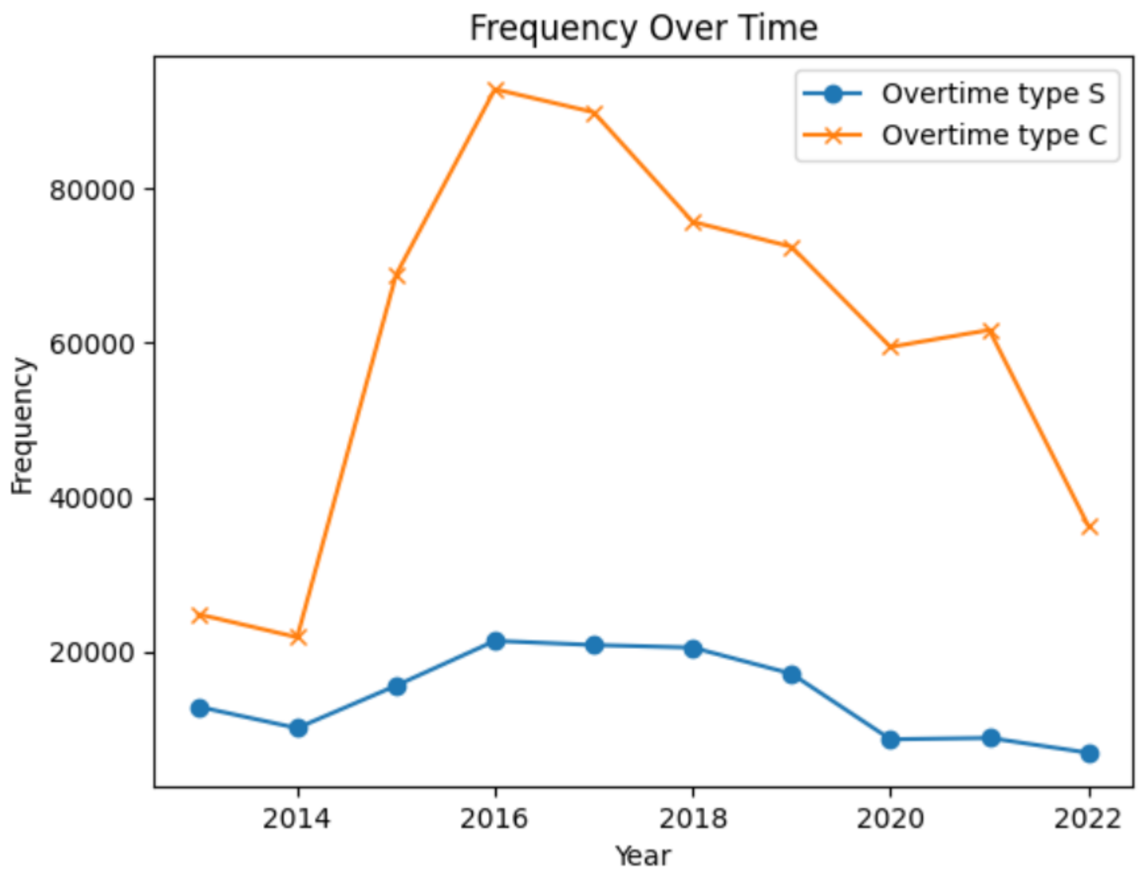
- Truc

```
[50] print("Unique Overtime pay type:")
     for i in range(0,10):
       print("Year", i+2013, ":", overtime_data_list[i]['TYPE'].unique())
```

```
Unique Overtime pay type:
Year 2013 : ['Z' 'C' 'S' 'R']
Year 2014 : ['S' 'R' 'Z' 'C']
Year 2015 : ['Z' 'S' 'R' 'C']
Year 2016 : ['S' 'C']
Year 2017 : ['C' 'S']
Year 2018 : ['C' 'S']
Year 2019 : ['C' 'S']
Year 2020 : ['C' 'S']
Year 2021 : ['C' 'S']
Year 2022 : ['C' 'S']
```

- I'm not sure what overtime pay type 'Z' and 'R' are (in 2013-2015).
- My guess for overtime pay 'S' and 'C' is:
  - 'S' : Scheduled Overtime
  - 'C' : Court Overtime

Thus, I will do analysis of change in frequencies of 'S' and 'C' overtime pay type.



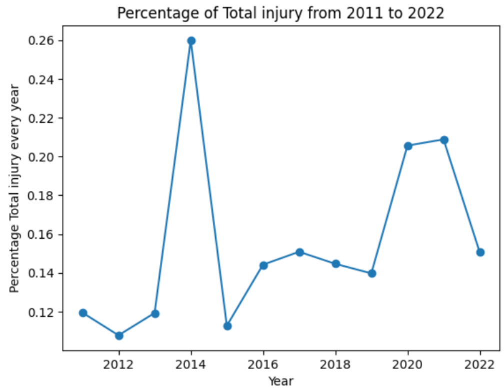
- The graph indicates a notable disparity between the frequencies of court overtime (Type C) and scheduled overtime (Type S).
- This difference is particularly pronounced in the years 2015, 2016, and 2021, where the occurrence of court overtime experienced a significant surge.
- These specific years warrant focused attention in our investigation, as they may hold crucial insights into the underlying reasons for the elevated overtime expenditure.

[Can Wang]

- Analysing data from previous graphs. Making observations.
- Noticing that there is a increase on injury in 2014.
- the unusually high injury rate for the Boston Police Department in 2014 is associated with the Boston Marathon bombing that occurred in 2013. While the bombing itself took place in 2013, the subsequent manhunt, investigations, and related events spanned into 2014 and beyond. During the bombing and the manhunt that followed, several officers

were injured, which contributed to the elevated injury rates for the department during that period. Boston Marathon bombing - Wikipedia

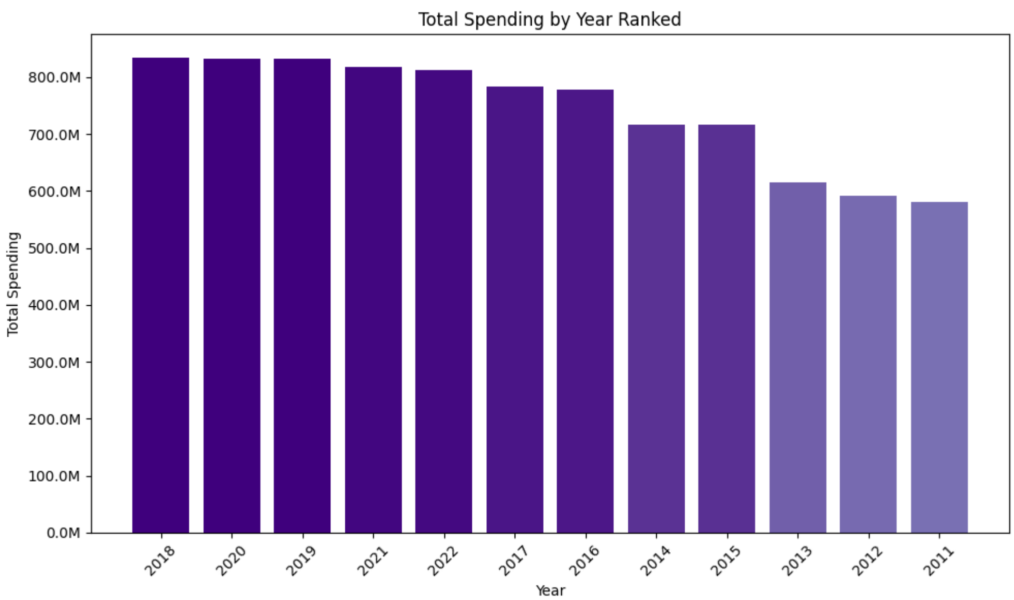
- Fixing some issues related to new imported data.



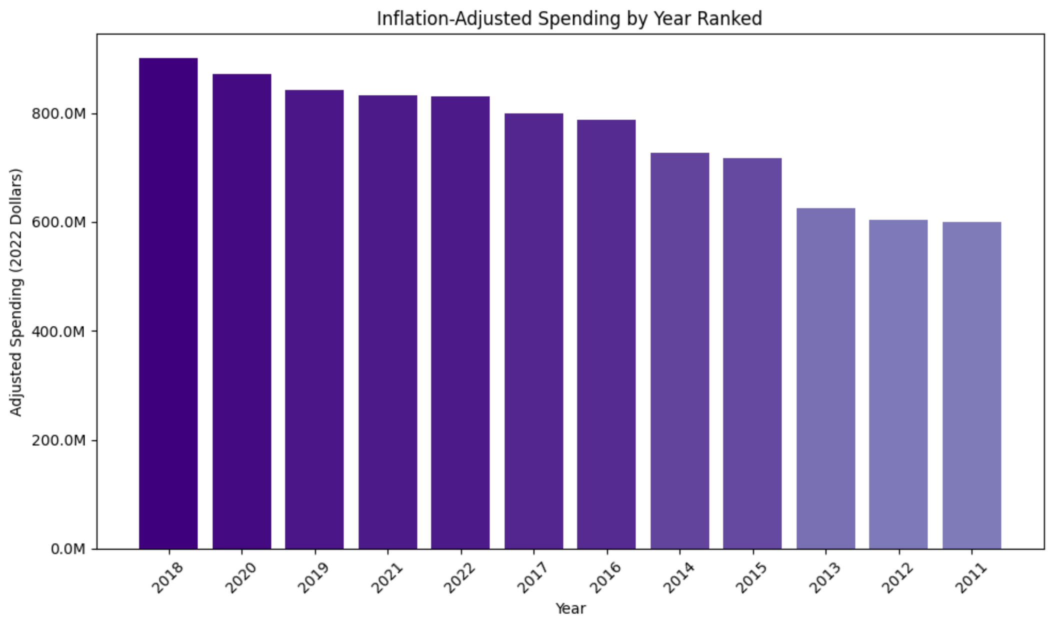
-

[Al Mbaye]

- Displayed a ranking of total spending by year in descending order using a color gradient



- Displayed the same ranking with spending adjusted for real inflation



- As we can see, there is no difference in the ranking order between both graphs when accounting for each individual year's inflation rate from 2011 to 2022.
- Adjusting for inflation gives us a clearer picture of the real value of spending in 2023. For example, raw spending in 2018 seems much higher than 2011 but, when adjusted, the gap between the values decreases.

3. Issues or blockers:

- Similarly as in last week, cleaning up the data took us a while, since some of the years had different formatting than others, and we had to manipulate the data in order to follow the same format across all years

4. Plans for next week

- Answer 2 key questions?
- Find demographics data of BPD officers, and analyze it
- Analyze BPD field activity data
- Analyze BPD campaign contribution data