COMPENSATION MODEL

Department of Finance, Bureau of Budget Management
City of Baltimore, MD
Sara Brumfield — Data Lead

coding it forward >

VENNILA ANNAMALAI
UC Berkeley
BA Statistics, BA Data Science

PURPOSE

- The City of Baltimore has a lot of data about how much workers are being paid
    - Pay grade, yearly salary, vacancies, Classification ID, budgeted salary, etc.
- Some positions' salaries have not been updated in the past couple years and as a result there are some people being possibly underpaid

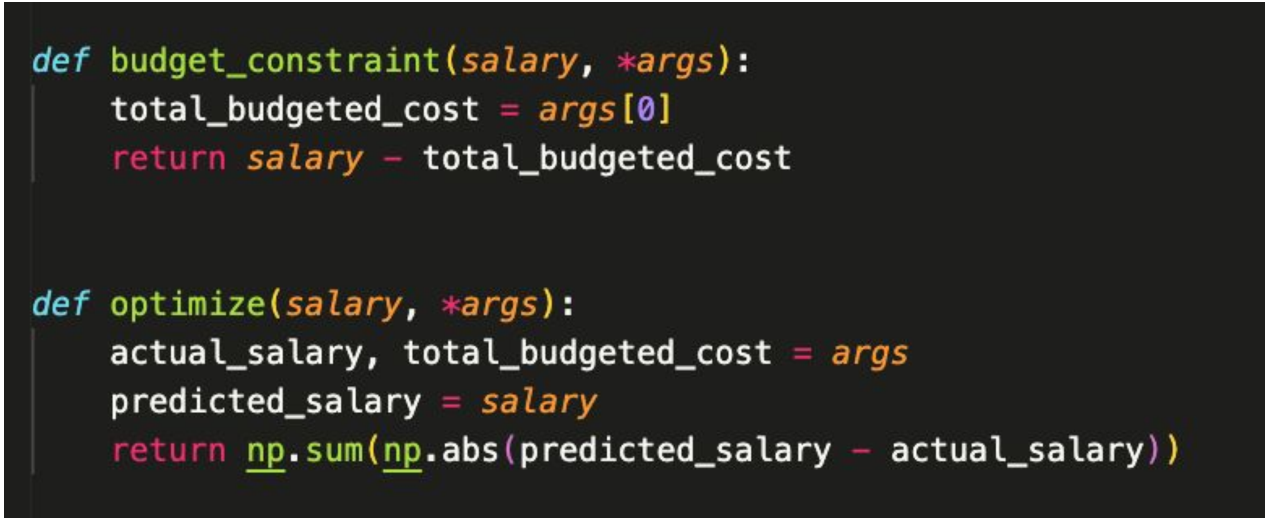
MAIN QUESTION

- How can we use the compensation data to more effectively allocate money and identify positions that are being underpaid?

METHODOLOGY

- I began by merging and cleaning all the data tables I was given
- I also to removed all “Abolished” positions to make sure I was only working with positions that still existed
- To make my results more readable, I chose to group the data by “Position Grade”, however the code can be altered to provide more specific details/changes to salaries

METHODOLOGY

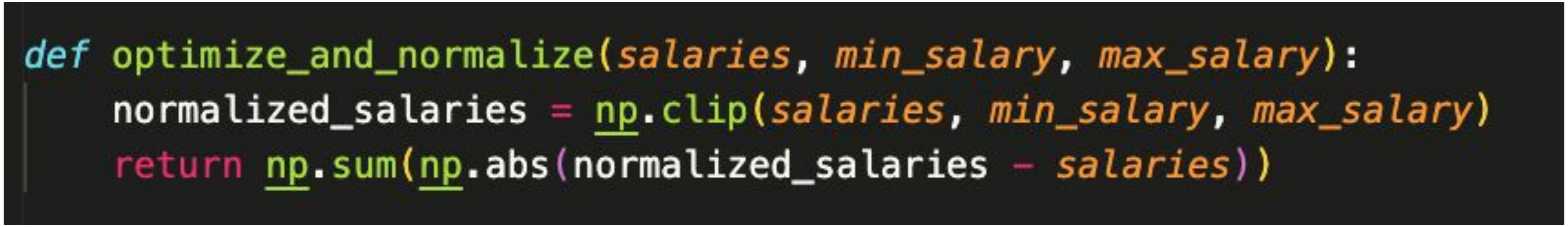


```
def budget_constraint(salary, *args):
    total_budgeted_cost = args[0]
    return salary - total_budgeted_cost


def optimize(salary, *args):
    actual_salary, total_budgeted_cost = args
    predicted_salary = salary
    return np.sum(np.abs(predicted_salary - actual_salary))
```

Using the existing “Budgeted Salary” and “Total Budgeted Cost” columns, I was able to write functions that optimized each salary by minimizing the difference between the “Total Budgeted Cost” and “Budgeted Salary”

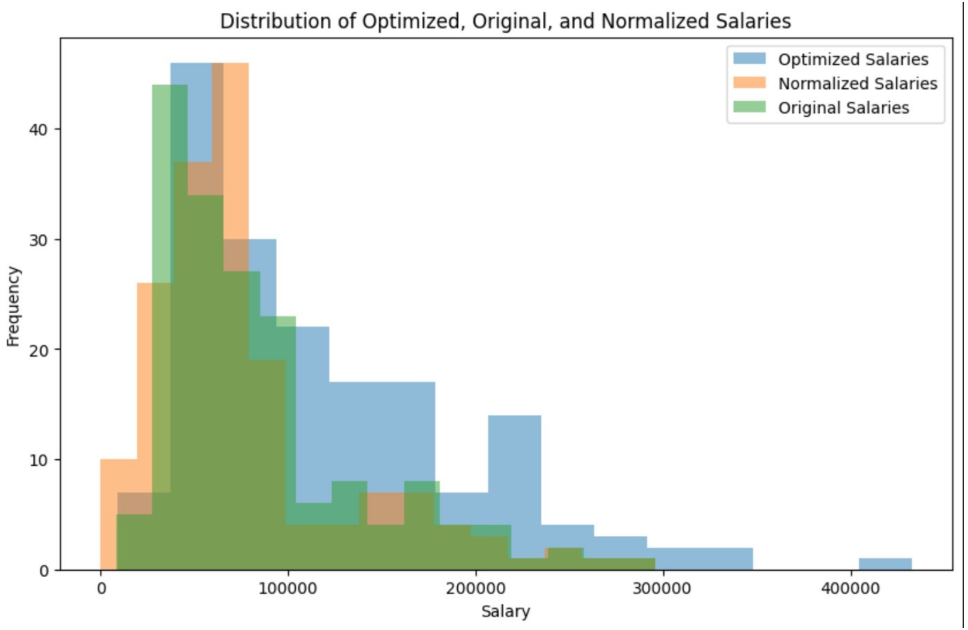
METHODOLOGY



```python
def optimize_and_normalize(salaries, min_salary, max_salary):
    normalized_salaries = np.clip(salaries, min_salary, max_salary)
    return np.sum(np.abs(normalized_salaries - salaries))
```

However, this didn’t alleviate the problem of people being underpaid. Also, “Total budgeted Cost” included benefits and was not an accurate upper bound. I updated the function so the upper bound was “Range Maximum” the lower bound “Range Minimum”. The np.clip function increases any values below “Range Minimum” to be equal to “Range Minimum” and brings high values down to “Range Maximum”

FINAL PRODUCT



This histogram visualizes the distributions of the original, optimized, and normalized salaries

FUTURE STEPS

Overall, this project is a template for Baltimore to visualize and test out different methods for altering salaries. There are obviously many different factors that go into deciding how much someone is paid in order to maintain fairness. My hope is that this model provides an overview of the bigger picture and can help hypothesize various solutions to disparities in how employees are being compensated.

If you're interested, you can find my code here.