An Analysis of the Relationship between Years of Experience, Salary and Programming Languages in Data Analyst Job Descriptions

Ibari J. Nwosu

August 31, 2025

Introduction: This report analysed the relationship between salaries and two variables: years of experience and two programming languages, using data scraped from Data Analyst job descriptions in Glassdoor.

Methods: A preliminary data manipulation exercise generated variables for years of experience and programming languages from the job descriptions, using a custom AI search function based on Gemini 2.5 in Google Sheets. The resulting data was then analysed using Python in Julius, working with the Claude IV Sonnet LLM, and using the prompts provided in the course materials.

General:

The dataset included 400 job descriptions, all of which contained salary information. Of these, 310 job descriptions included data on years of experience. Programming languages were categorised for all 400 job descriptions as R only, Python only, both R and Python, and neither R nor Python.

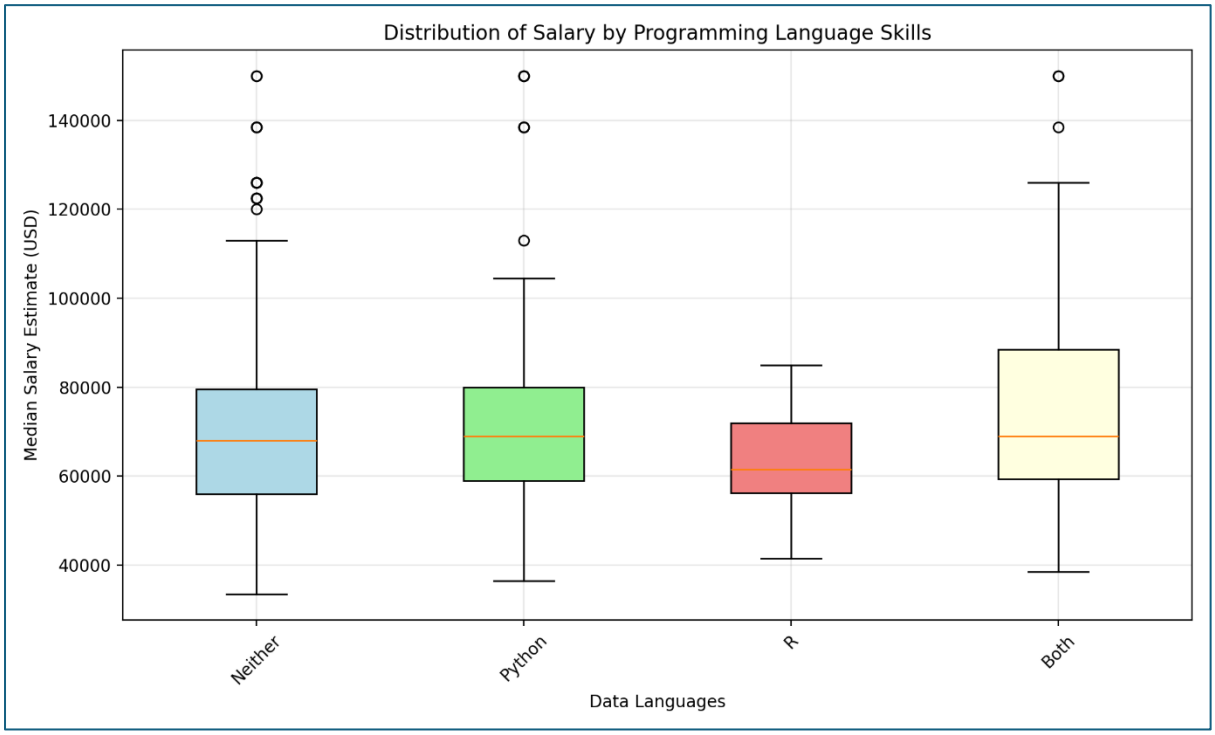
1. Analysis of Relationship between Years of Experience and Salary

Correlation Analysis between Years of Experience and Median Salary USD

- Pearson correlation coefficient: 0.0249
- P-value:0.6621
- Sample size for correlation: 310

The scatter plot shows no relationship between years of experience and salary in this dataset. The correlation coefficient of 0.025 indicates an extremely weak positive relationship, but with a p-value of 0.662, this relationship is not statistically significant.

2. Analysis of Relationship between Programming Language and Salary



Summary salary statistics for data languages:

R (n=15):

- Mean salary: $64,533
- Median salary: $61,500
- Std Dev salary: $12,219

Python (n=61):

- Mean salary: $74,836
- Median salary: $69,000
- Std Dev: $27,997

Both languages (n=68):

- Mean salary: $76,125
- Median salary: $69,000
- Std Dev: $25,579

Neither language (n=256):

- Mean salary: $70,344
- Median salary: $68,000
- Std Dev: $23,896

Test of Significance (ANOVA) for Salary Differences by Language Categories

- F-statistic: 1.7535
- P-value: 0.1555
- Not significant at α=0.05
- Effect size (η^2): 0.0131
- Effect size interpretation: Small

Interpretation:

1. Experience vs. Salary: There is no meaningful relationship between years of experience and salary in this dataset. This could suggest that other factors (such as company size, location, specific skills, industry, etc) may be more important determinants of salary than experience alone.

2. Programming Languages vs. Salary: While there are observable differences in mean salaries between language groups with "Both" (Python + R) having the highest mean salary at $76,125 and "R" having the lowest at $64,533), these differences are not statistically significant. The ANOVA test shows no significant difference between the different groups ($p = 0.156$), suggesting that programming language skills alone don't significantly impact salary in this sample.

The analysis reveals that neither experience nor programming language skills show statistically significant relationships with salary in this job posting dataset. This suggests that salary determination in data analyst roles may depend more heavily on other factors not captured in these variables.

Reflection:

What was easy or difficult? The instructions for the exercises were clear, and the prompts worked as expected. I had no difficulties. I was once again surprised by the ease with which AI performs tasks that would previously have taken weeks or even months.

What was surprising or noteworthy? The dataset I passed to Julius contained different variable names from the ones in the prompts I copied from the course material. I was surprised to see Julius examine the variable headings and correctly identify which ones matched the instructions given. For example, the prompt said 'Minimum Years Experience' while my column heading was 'Years of Experience'. Data analysis programmes generally require variables to be exact matches, and a single letter missing in a variable name will usually give you an error message. The fact that the AI can detect and flag things like this and correct them is such a time saver.

What did you learn? I learnt that Julius has the ability to generate a preview of the dataset and also produce some data tables and summary statistics. I had some problems in my analysis last week because I did not examine the dataset before starting the analysis. I know how to anticipate and manage this now.

I also noticed that the AI output from this week's exercise had a different visual and narrative format from last week, even though the task was essentially the same (correlation analysis with scatterplot). I find this variability in output interesting and a little disconcerting as I am used to receiving standardised output for specified commands when I analyse data.