Introduction

This report analyzes 400 data-analyst job postings to explore two main questions:

- How does the minimum years of experience listed relate to the median salary estimate?

- How does the median salary estimate vary with the data-language requirements in a posting (R, Python, both, or neither)?

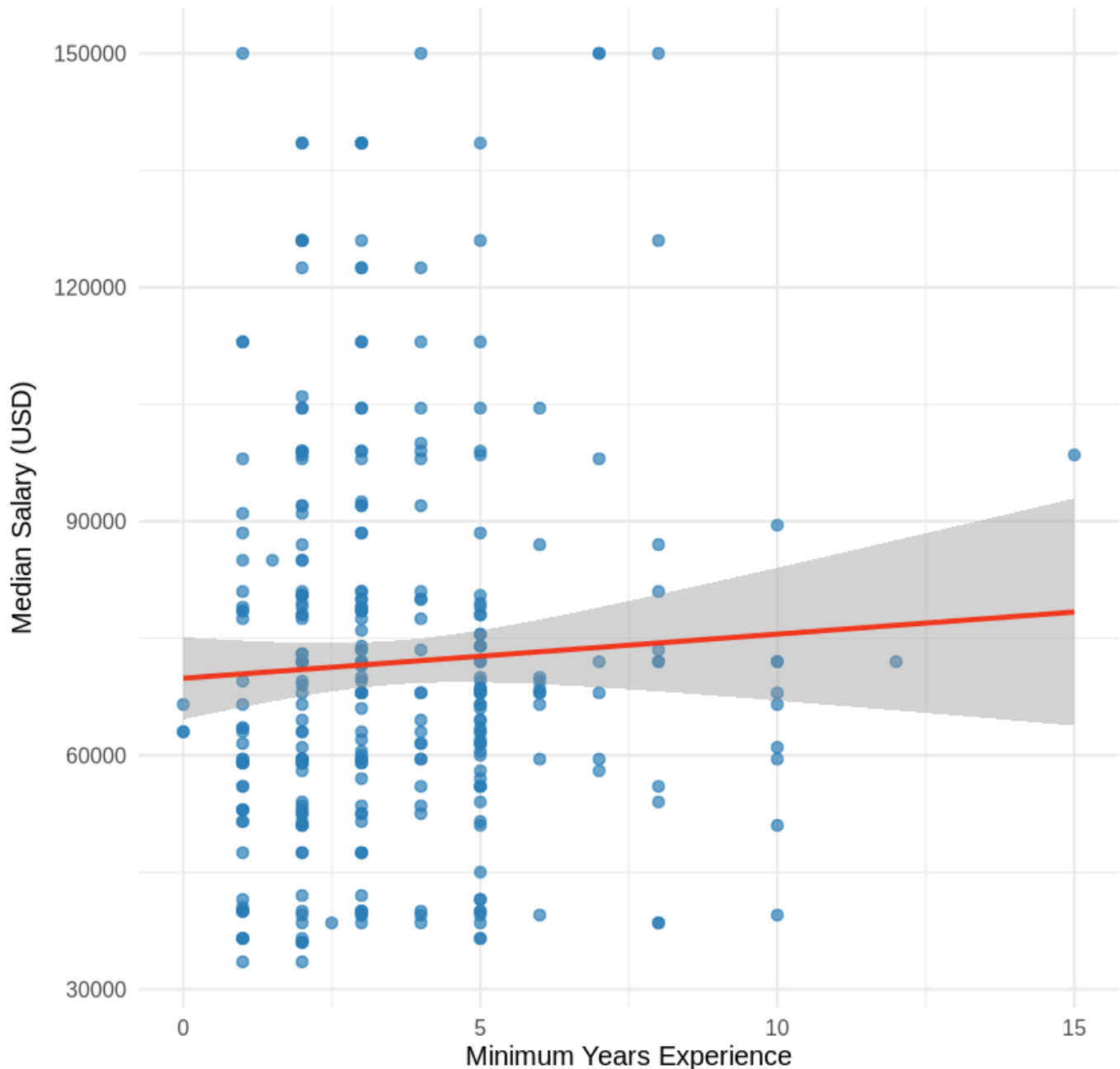This analysis was conducted with the help of Julius.ai, and the key findings are presented below.

Key finding

1. Years of Experience Analysis

Scatter plot

Minimum Years of Experience (x-axis) vs. Median Salary Estimate in USD (y-axis)

Minimum Years Experience vs. Median Salary (USD)

Interpretation

The scatter plot shows an upward trend: postings with higher minimum-experience requirements generally list higher median salaries. A fitted line with a confidence band highlights this positive relationship.

However, salaries vary widely at nearly every experience level, which means experience explains part but not all of the pay differences. A "5-year" requirement, for instance, appears alongside both lower and higher salary postings depending on other factors like role scope, company, industry, and location.

2. Programming Languages and Pay

Box Plot

Median Salary Estimate in USD distributions by data-language category
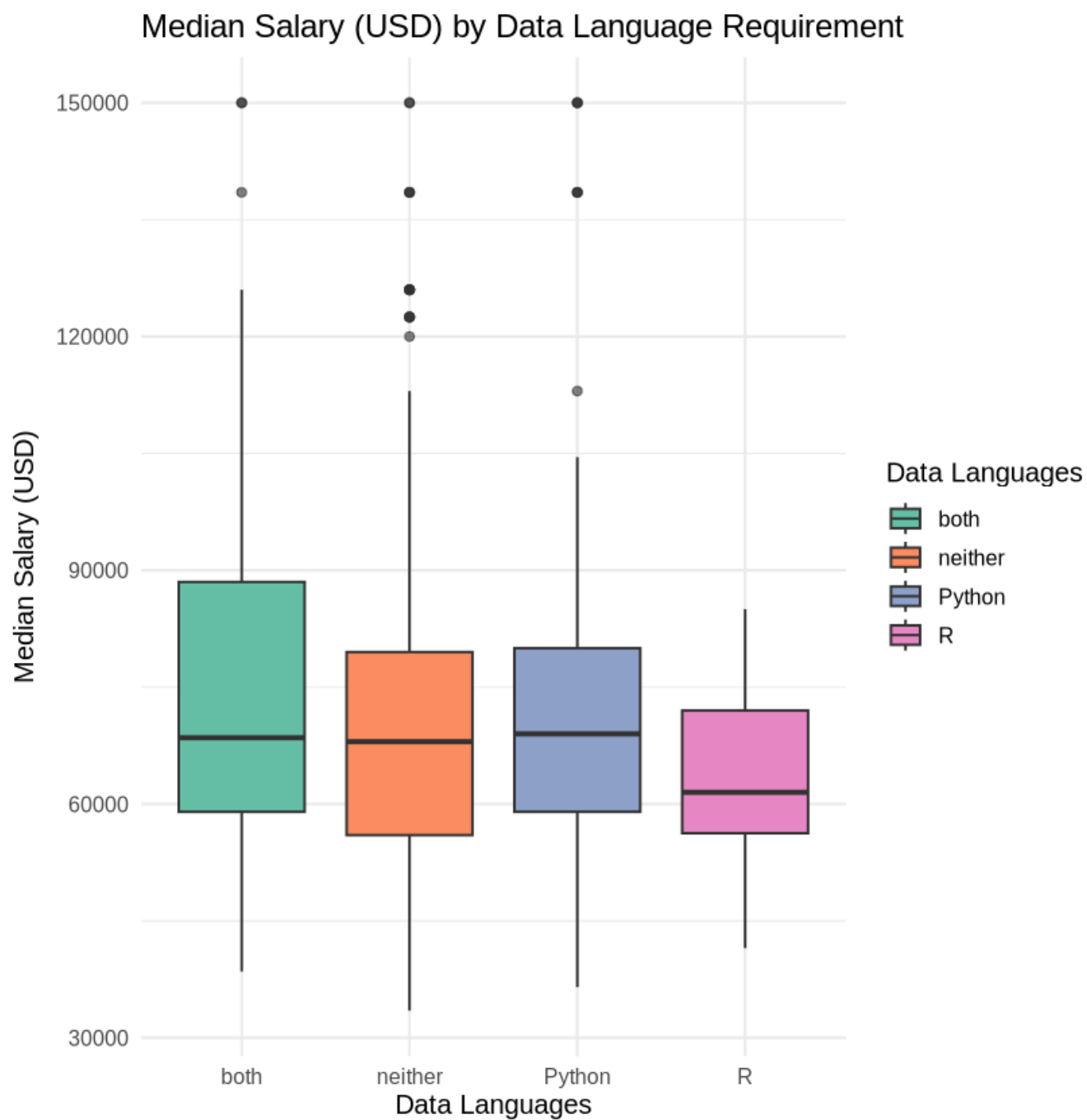
Interpretation

Job postings that require Python or both Python and R tend to show higher median salaries than postings requiring neither language.

The "both" category displays a slightly higher median and a larger upper spread. Python-alone postings also sit above the "neither" category on average. This suggests broader language requirements especially Python plus R are associated with higher pay, likely reflecting expectations for more advanced or wide-ranging technical work.

3. Reflections

What was easy or difficult

The most challenging part was working with the API in the sheets it was a bit confusing, especially when switching between different accounts, which sometimes made it unclear what was linked to which. On the other hand, the easiest and most satisfying part was watching how well the output was generated once everything was set up.

What was surprising or noteworthy?

It was eye-opening to see how combining even basic visuals (scatter + box plots) with clear explanations can tell a practical story for both job seekers and hiring teams with no any code application, just writting and waiting for the output

What I learned?

The biggest takeaway for me is seeing how much a large language model can help in handling complex data. It saves time and produces clear, understandable outputs, which makes analysis and interpretation much easier.