Irene(Jiaying) Xu, UAL Creative Computing Institute

23/24 Introduction to Data Science
In-Class Assignment Portfolio - Exercise 1: Gaining Insights from Dataset

0 Introduction

**Related Course Material: Week 3 Basic Statistics, Week 4 Probabilities & Distribution*

In Week 3 & Week 4, we looked at some basic approaches to summarising datasets and extracting valuable statistical insights from them. Based on that, this exercise focuses on applying these techniques to a new dataset. The goal is to familiarise myself with essential data analysis skills and explore the use of various data visualisation tools.

1 Dataset Prep & Overview

The dataset used for this exercise is "*Jobs and Salaries in Data Science*" created by Hummaam Qaasim from Kaggle, which includes more than 7,000 entries of salaries related to data science jobs in the year 2023.

1) Data Pre-processing

   The dataset itself is already pretty organised. I have removed some rows and only kept those of full-time salaries with the work year of 2023, as well as sorted it based on experience level.

	work_year	job_title	job_category	salary_in_usd	employee_residence	experience_level	work_setting	company_location	company_size
6857	2023	Data Scientist	Data Science and Research	19910	Brazil	Entry-level	Remote	Brazil	L
814	2023	Data Integration Specialist	Data Management and Strategy	240000	United States	Entry-level	Remote	United States	M
815	2023	Data Integration Specialist	Data Management and Strategy	100000	United States	Entry-level	Remote	United States	M
6802	2023	Data Analyst	Data Analysis	48000	United States	Entry-level	In-person	United States	M
6801	2023	Data Analyst	Data Analysis	55000	United States	Entry-level	In-person	United States	M
...	...	...	...	...	...	...	...	...	...
6110	2023	Data Architect	Data Architecture and Modeling	155000	United States	Executive	In-person	United States	M
6109	2023	Data Architect	Data Architecture and Modeling	180000	United States	Executive	In-person	United States	M
2121	2023	Analytics Engineer	Leadership and Management	200000	United States	Executive	In-person	United States	M
4977	2023	Data Scientist	Data Science and Research	250000	United States	Executive	Remote	United States	M
2420	2023	Head of Data	Leadership and Management	329500	United States	Executive	Remote	United States	M

7437 rows × 9 columns

2) Dataset Overview

   The dataset comprises 4 experience levels, 10 job categories, 109 job titles, 55 employee residences, and 47 company locations (some are working remotely).

   In terms of job category, it appears that most people working in Data Science were doing research, data engineering, and machine learning and AI-related work. However, as it's uncertain about the data collection methods of this dataset, it would be imprudent to hastily conclude that 'Data Engineer' is the most popular job position in data science, or that 'Data Science and Research' is the most popular sub-field.

Nevertheless, the substantial number of entries for these positions and categories could provide valuable insights into the salary ranges associated with them - which is the main point of this exercise.

```
job_category
Data Science and Research         2410
Data Engineering                  1697
Machine Learning and AI           1189
Data Analysis                     1120
Leadership and Management          414
BI and Visualization               305
Data Architecture and Modeling     208
Data Management and Strategy        48
Data Quality and Operations         43
Cloud and Database                   3
Name: count, dtype: int64
```

```
job_title
Data Engineer                1662
Data Scientist               1536
Data Analyst                 1090
Machine Learning Engineer     862
Applied Scientist             254
                             ...
Lead Data Scientist             1
Principal Data Engineer         1
Finance Data Analyst            1
Power BI Developer              1
Azure Data Engineer             1
Name: count, Length: 109, dtype: int64
```

```
experience_level
Senior         5485
Mid-level      1405
Entry-level     320
Executive       227
Name: count, dtype: int64
```

While job titles seem to provide more details, they appear to have some replications and confusing subcategories (e.g., can an Azure Data Engineer also be referred to simply as a Data Engineer?). This seems to be a common issue when building datasets from various sources, especially when the information is contributed by individuals who may express the same thing/position with varying levels of detail or use different phrases. Should there be more data processing? Maybe create a list of pre-labelled positions and a model to classify different job titles? For now, just to gain some broad but more reliable insights, I will only use the 'job category' and 'experience level' columns for further analysis.

In addition, it's important to note that the majority of the recorded jobs in this dataset are based in the U.S. – so the insights drawn from the dataset may not be suitable to be applied to a larger scale or a different region (e.g., inapplicable for EU job market).

Overall, analysing a dataset involves not only understanding its numerical data, but also being aware of how the data was collected and any potential biases may exist.

```
employee_residence
United States     6637
United Kingdom     339
Canada             172
Spain               66
Germany             29
France              20
```

```
company_location
United States     6654
United Kingdom     340
Canada             172
Spain               65
Germany             29
France              21
```

2 Statistics & Insights

1) <u>Central Tendency</u>

Based on the salaries recorded (across all experience levels and all job categories) in the dataset, the average annual salary for jobs in data science is $155,289.18, the median is $147,100, and the mode is $150,000. (doesn't seem there is a significant difference between these measures, might need to examine more measures, e.g., distributions, to gain a better understanding of the data).

If we look at different categories (across all experience levels), Machine Learning and AI have the highest average annual salary of $187,841.38, while the salary in Data Management and Strategy is only around half of that, at $99,130.06.

job_category	salary_in_usd
Data Management and Strategy	99130.062500
Data Quality and Operations	105374.906977
Data Analysis	109895.885714
BI and Visualization	135989.600000
Cloud and Database	141666.666667
Leadership and Management	147428.229469
Data Engineering	149753.327637
Data Architecture and Modeling	153436.043269
Data Science and Research	170201.760166
Machine Learning and AI	187841.375105

experience_level	salary_in_usd
Entry-level	94496.375000
Mid-level	122881.299644
Senior	165721.592160
Executive	189496.048458

Furthermore, when examining the average salary for each category under different experience levels, though the overall salary ranking undergoes slight changes, it is obvious that individuals in the field of machine learning and AI, as well as data science and research, earn considerably more than those in data analysis, data management, and operations, regardless of their career level.

job_category	salary_in_usd
Data Quality and Operations	36627.000000
Data Analysis	73944.782178
BI and Visualization	87583.333333
Data Management and Strategy	89837.500000
Data Engineering	94836.806452
Leadership and Management	97444.600000
Machine Learning and AI	100278.600000
Data Science and Research	114933.169811

job_category	salary_in_usd
Data Quality and Operations	67974.000000
Data Management and Strategy	94833.333333
Data Analysis	94898.663230
BI and Visualization	105052.745455
Leadership and Management	117272.302752
Data Engineering	121693.218487
Data Architecture and Modeling	122496.965517
Data Science and Research	138842.275204
Machine Learning and AI	154635.426036

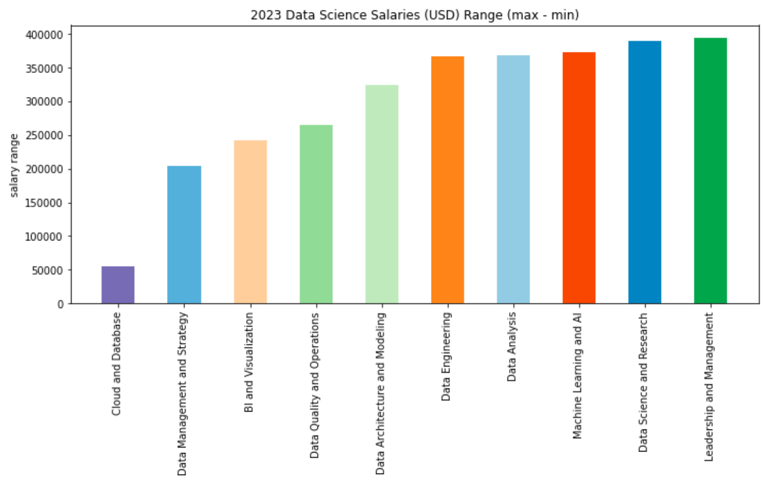
job_category	salary_in_usd
Data Management and Strategy	109621.833333
Data Analysis	121008.321229
Data Quality and Operations	124716.666667
Cloud and Database	141666.666667
BI and Visualization	143109.357143
Leadership and Management	153398.279528
Data Architecture and Modeling	158346.242938
Data Engineering	158630.323232
Data Science and Research	178091.458422
Machine Learning and AI	195531.377665

job_category	salary_in_usd
Data Analysis	113125.000000
Data Architecture and Modeling	167500.000000
Data Engineering	181713.466667
BI and Visualization	185566.666667
Leadership and Management	191352.608696
Machine Learning and AI	210463.600000
Data Science and Research	212272.704918

Also, the mode of Machine Learning and AI is higher than its median, indicating right-skewed data. In such cases, the median may be more representative compared to the mean as it is less likely to be influenced by outliers.

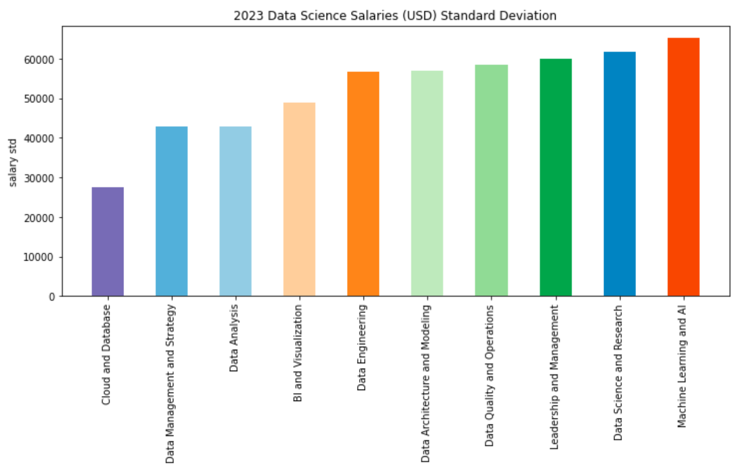
2) Range

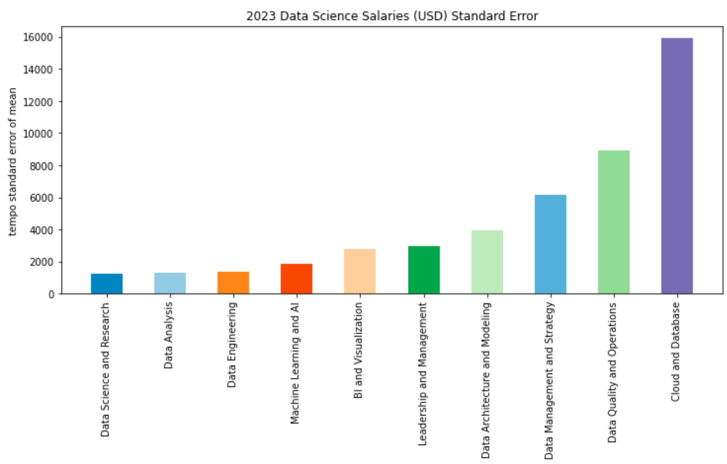
Among all the categories, Leadership and Management-related positions have the largest gap between the lowest and highest salary, while Cloud and Database-related positions have the smallest gap. However, the latter one is statistically meaningless because there are only 3 entries related to this category — simply not enough data to draw useful insight. This can also be validated in the Standard Error Plot attached in the next section, showing that Cloud and Database have the highest standard error – underscores the importance of examining multiple measures, such as range and standard error together, to obtain meaningful insights.



3) <u>Variance</u>

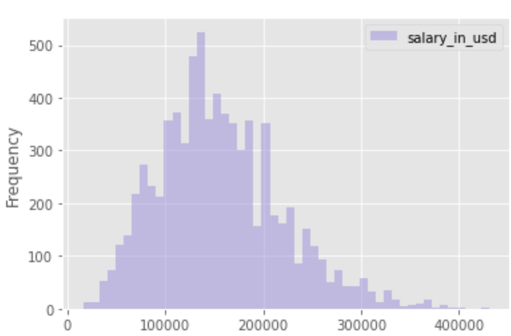
In terms of variance, Machine Learning and AI, with the highest salary, also exhibit the greatest variation. This indicates a higher salary ceiling and (possibly?) much more potential compared to other categories. In contrast, data management and data analysis jobs, overall, offer less room for growth. This could maybe explain the common career path in Data Science where individuals with limited technology background often start with data analysis and then gradually transition to other areas. The Cloud and Database data, as mentioned above, failed to produce meaningful results as the collected dataset size was not sufficient.

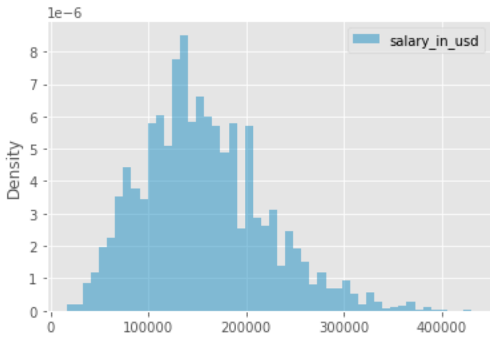




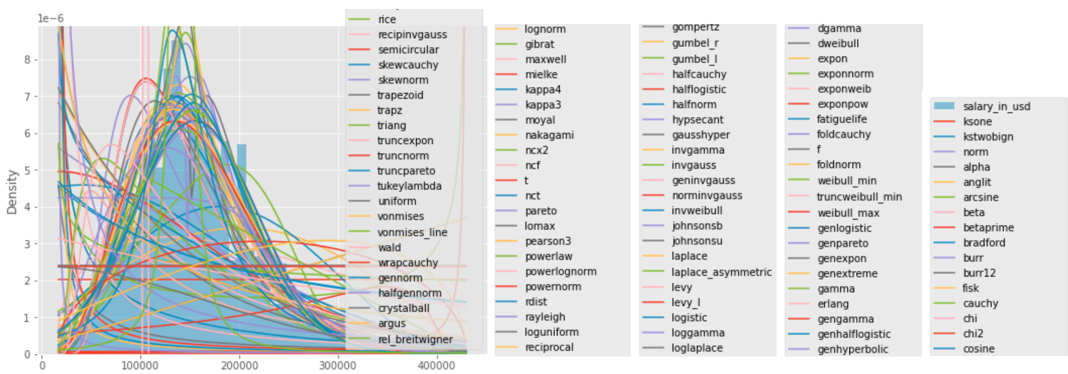
4) Distribution

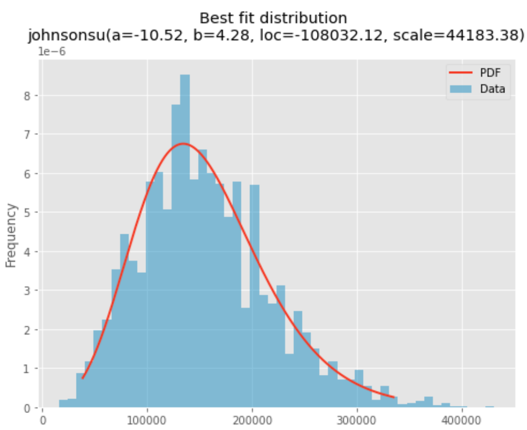
In terms of distribution, overall, this dataset exhibits a distribution that is quite close to normal, with a slight right skewness. The right skewness reflects that there are few entries recorded in this dataset with exceptionally high salaries.

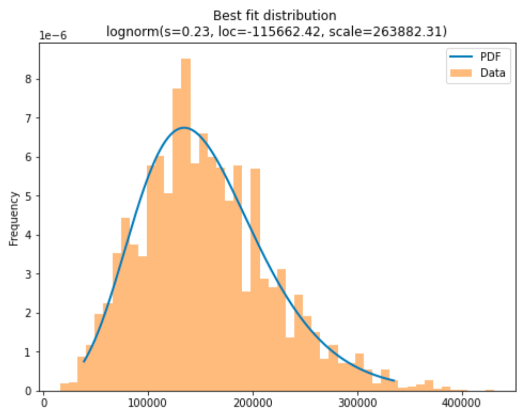




Python Library ***Scipy*** was used for distribution fitting. While the notebook suggests selecting a few distributions to find the best fit, I tried using all available distributions at once and let the computer handle the task – the best-fit distribution for this dataset is johnsonsu (or, if we select only 4 most common distributions - 'norm', 'expon', 'skewnorm', 'lognorm', the best fit would be ‘lognorm’), and lognormal distribution is indeed commonly used to model data that are inherently positive and skewed, such as income (this dataset!).





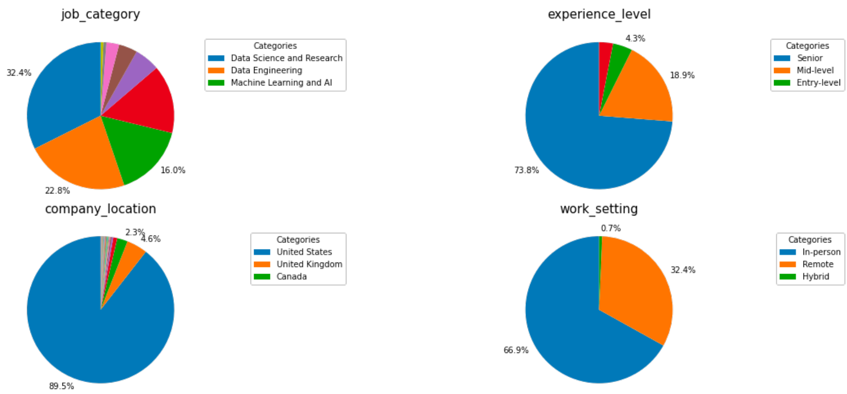


3 Data Visualisation

I have also explored the use of various charts and plots here to see what other information I can obtain from the dataset, mainly using Matplotlib and Seaborn. Some interesting charts and plots and corresponding findings are included below (full exploration can be accessed in the notebook via the provided GitHub link at the end), overall, I find it is much easier to identify patterns and insights from graphs compared to stats alone, and visualised data is more user-friendly and intuitive for interpreting complex information.

1) Pie Chart

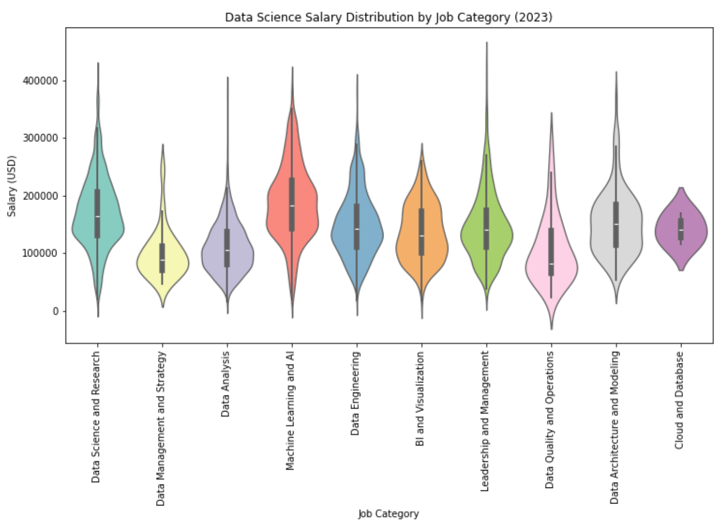
   Pie chart works great in showing the percentage of each category in the whole dataset.



   The above charts show that most data entries are related to data science and research in terms of category, senior level in terms of experience, the U.S. in terms of company location, and in-person in terms of work setting. Though if there are too many labels, the chart and labels may become less organised and difficult to read (this has been fixed in the above chart by displaying only the top 3 labels and autopct values, so the chart is clean and easier to read data).

2) Violin Plot

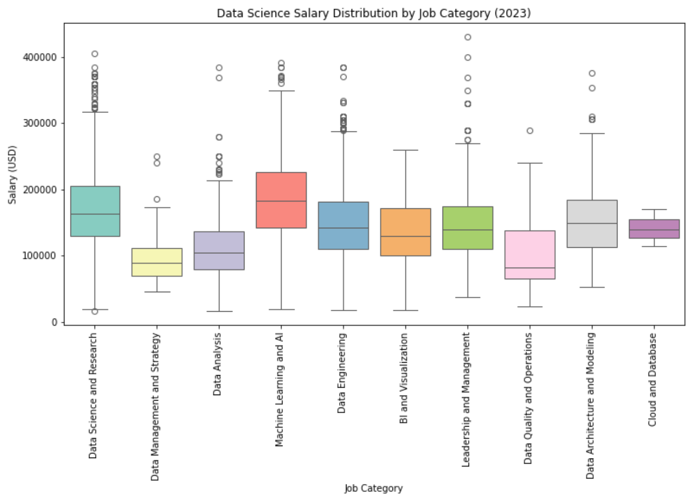
   The violin plot is created by using Seaborn – a Python library that is built on top of matplotlib and it is easier/quicker to create visually appealing plots with less code. It provides a comprehensive overview of the data distribution as well as statistics and shows the comparisons over multiple categories.



From the above chart, we can see that almost every category has a right skewness (longer tail at the top) except for Cloud and Database (not enough data), meaning there are few individuals with extremely high salaries included in the dataset.

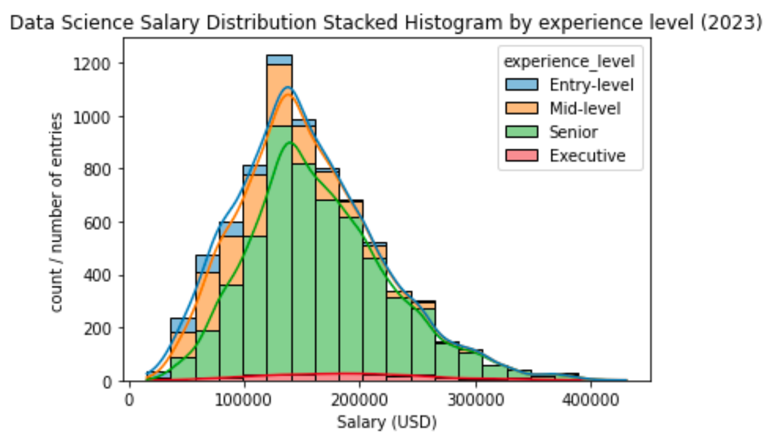
Most positions in Data Management and Strategy seem to have a relatively low avg. salary compared to other categories.

3) <u>Box Plot</u>
The box plot looks similar to the violin plot, but it provides more details in summary statistics and outliers and doesn't really have a view of the data distribution. It explicitly highlights outliers using small circles outside the box.



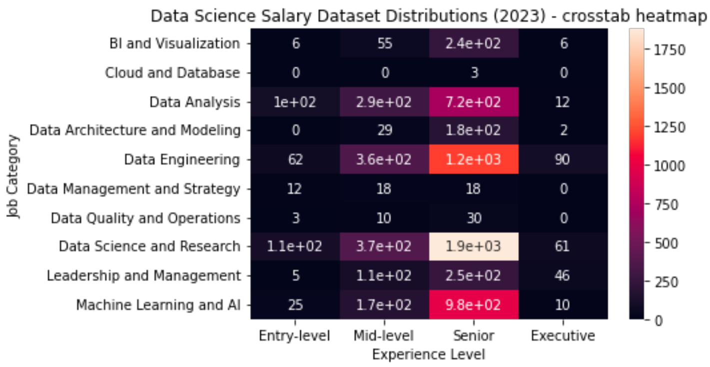
Based on the above, Leadership and Management have the widest outlier spread (makes sense as leadership positions could be offered with high salaries, depending on the size of the company). Cloud and Database seem to have no outliers, but this is because only 3 salary entries were included. Interpretation should consider other plots and graphs as well (looking at only 1 plot will lead to misinterpretation of the dataset).

4) <u>Stacked Histogram (with KDE)</u>
A stacked histogram with KDE combines elements of both histograms and kernel density plots, which is very useful for comparing distributions between different categories and visualising the overall distribution.



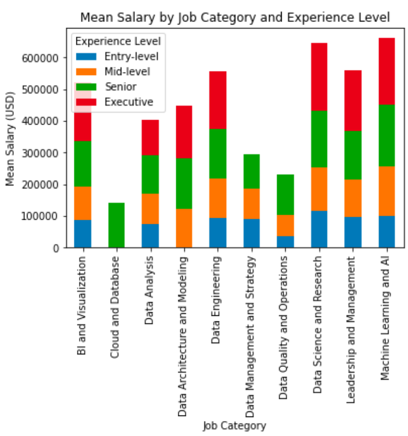
The above KDE green lines/plots tell that most of the data included in this dataset are senior level positions; and much fewer data were included for entry level and executive level positions (blue, orange). The histogram also validates this point. It also shows most entry and mid-level datapoints are in the left part (lower salaries), and executive level jobs salaries are skewed towards the right/the higher end (though it is a bit too small to be clearly seen in this plot).

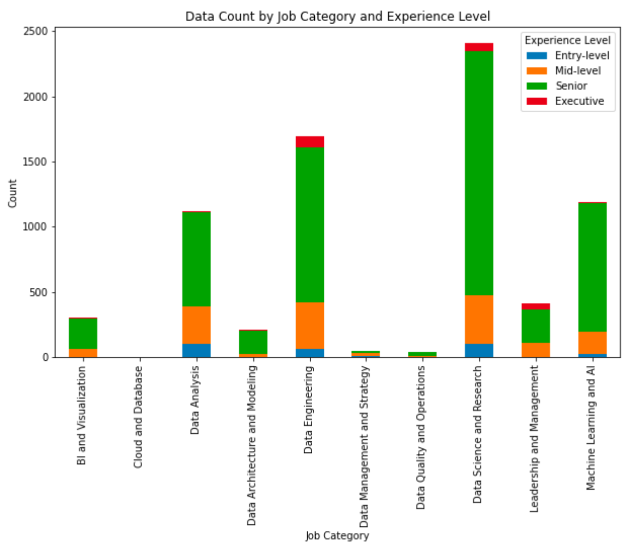
5) <u>Crosstab Heatmap</u>
Crosstab heatmap combines a crosstabulation table and a heatmap to show the relationships and frequency counts between job category & experience level; in terms of overall data distribution, this is an easier (than showing separately) and appealing way to show how data is distributed across two variables.



The heatmap above indicates that most data entries are associated with senior Leadership & Management positions. Additionally, it's evident that there is insufficient data to draw any meaningful insights for the Cloud and Database category. There is also no available data for Executive positions within the Data Management and Strategy and Data Quality and Operations categories.

6) <u>Stacked Bar Chart</u>
Stacked bar chart is effective in showing the composition of a whole dataset into various categories. Each bar represents the total (e.g., avg. salary or total counts), and its segments each represent a different subgroup contributing to that total.





The first stacked bar chart tells how salary (mean by default) is distributed across experience levels within each job category:

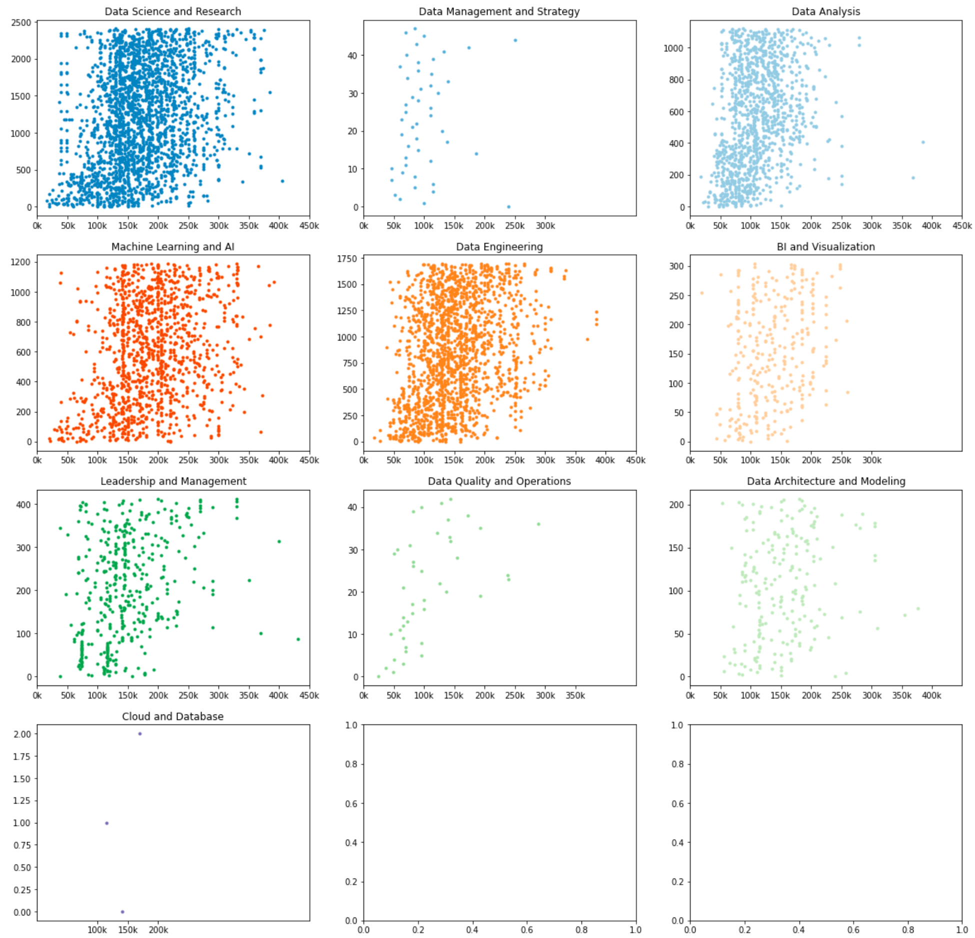
- Executive positions have the highest average salary, while entry-level positions have the lowest.
- Within the executive level, `Machine Learning and AI` roles have the highest salaries compared to other categories at the same level.

The second stacked bar plot focuses on the data count:

- Most data recorded in this dataset are senior level positions and data science and research-related positions (the same insight can also be drawn from the crosstab heatmap - I personally prefer the crosstab heatmap as it is more accurate in terms of exact count).

7) <u>Scatter Plot</u>

When there are many data points available, a scatter plot becomes a useful tool for people to observe the density and distribution of data, helping to understand where most points are concentrated and how the data is spread.



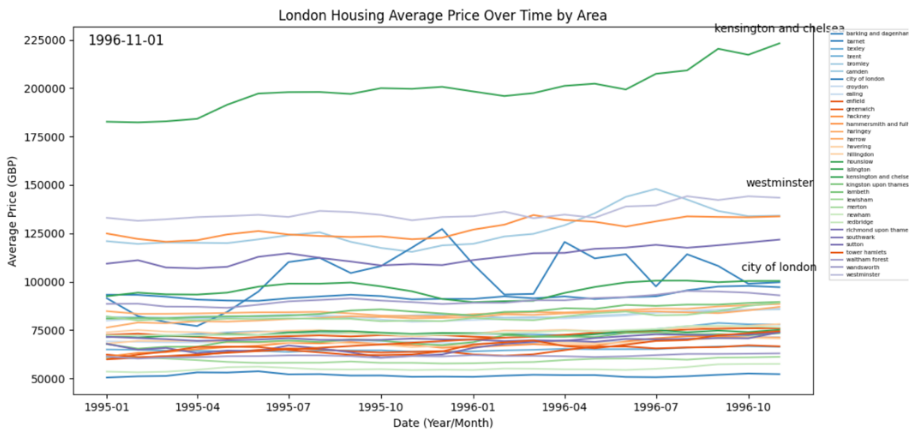
Taking a close look at each job category, we can see that there is a similar spread of variations for Data Science and Research, ML and AI and Data Engineering. BI and Visualisation and Leadership and Management, on the other hand, seem to have the biggest spread of variations. Cloud and Database, again, definitely have too little data to come up with any meaningful result.

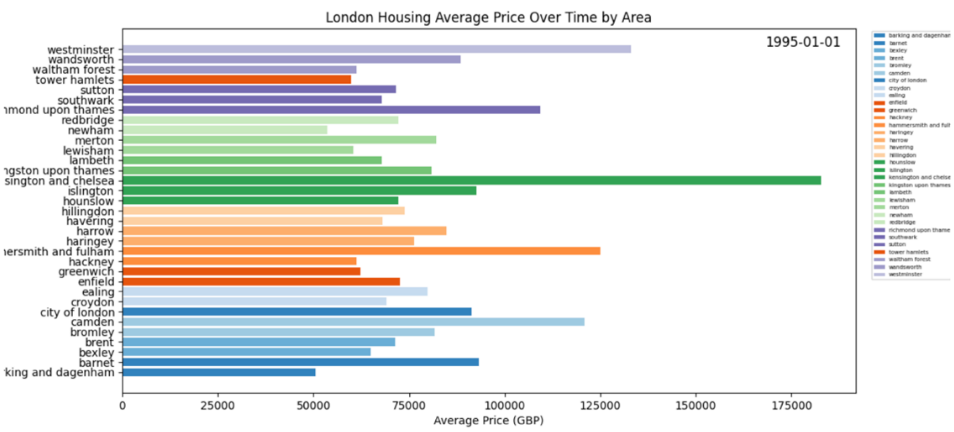
4 Further Exploration – Animated Charts

Compared to static charts and graphs, animated charts and graphs are usually more effective in data presentation and storytelling. This salary dataset contains multiple categorical columns but only one numeric column, making it hard to try more visualisation tools and charts. With more types of data (e.g., time data across different years), we can try to create animated charts to dynamically show trends over time or add more interactive functions to combine multiple charts into one holistic, animated visualisation.

To explore the use of the animated chart, I have found another dataset on Kaggle - "*Housing in London*" by Justinas Cirtautas, which contains datetime and housing price information. I made a brief attempt (with the help of ChatGPT and Blog posts on Spatialthoughts - Creating Animated Plots with Matplotlib, full details included in 'exercise-I-week-3-4-animated-chart-test.ipynb') to create two simple animated plots (line plotting & horizontal bar plotting) that visualises changes in London housing prices over time – from the graphs, we can easily tell that Kensington and Chelsea have always been the most expensive areas regarding housing prices, followed by Westminster. By animating the progression of data over time or other variables, the dataset and the plot are more effective in illustrating trends, patterns, and changes in data, making it easier and more user-friendly for viewers (which is more suitable to use if having a broader audience) to understand data and interpret information.

The screenshots of the animations can be found below:





To provide convenience in accessing the results, those animated charts were all saved as GIF files, which will be included in the submitted files.

5 Code, Supporting Document, and LLM Disclaimer

All datasets, code, Jupyter Notebooks, and GIF files are available to access via the GitHub repository: https://git.arts.ac.uk/23001934/ds-portfolio

ChatGPT 3.5 has been used in this exercise for some code debugging and for searching certain Python codes which have not been taught in class (especially for the last part - animated charts), example prompts used are shown below:

- 'how to set the animation.embed_limit rc parameter to a larger value to avoid error'
- 'how to plot a matplotlib animation in Jupyter Notebook'

6 References and External Resources

External Resources (Blogs, Posts, etc.):

[1] https://pandas.pydata.org/docs/getting_started/intro_tutorials/06_calculate_statistics.html

[2] https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.nlargest.html

[3] https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.pivot_table.html

[4] https://seaborn.pydata.org/examples/wide_form_violinplot.html

[5] https://seaborn.pydata.org/generated/seaborn.histplot.html

[6] https://seaborn.pydata.org/generated/seaborn.heatmap.html

[7] https://matplotlib.org/stable/api/ticker_api.html

[8] https://matplotlib.org/stable/gallery/pie_and_polar_charts/pie_features.html

[9] https://stackoverflow.com/questions/1823058/how-to-print-a-number-using-commas-as-thousands-separators

[10] https://stackoverflow.com/questions/48799718/pandas-pivot-table-to-stacked-bar-chart

[11] https://www.geeksforgeeks.org/zip-in-python/

[12] https://www.freecodecamp.org/news/lambda-sort-list-in-python/#whatisalambdafunction

[13] https://note.nkmk.me/en/python-pandas-nan-judge-count/

[14] https://medium.com/@jb.ranchana/easy-way-to-create-stacked-bar-graphs-from-dataframe-19cc97c86fe3

[15] https://stackoverflow.com/questions/43445103/inline-animations-in-jupyter

[16] https://spatialthoughts.com/2022/01/14/animated-plots-with-matplotlib/

[17] https://www.geeksforgeeks.org/how-to-add-text-to-matplotlib/

[18] https://www.geeksforgeeks.org/add-text-inside-the-plot-in-matplotlib/

[19] https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.text.html

[20] https://stackoverflow.com/questions/10624937/convert-datetime-object-to-a-string-of-date-only-in-python

**Detailed annotations were included in the Jupyter Notebook along with the code to indicate where ChatGPT and external resources, such as StackOverflow posts and other Python study materials were used/referenced.*