The dataset that I chose to present in this notebook is called "**Data Science Job Salaries 2024**". I found it rather interesting as I first entered the tech industry as a Data Analyst back in 2019. The dataset is authored by Abhinav Shaw and can be found on Kaggle at https://www.kaggle.com/datasets/abhinavshaw09/data-science-job-salaries-2024 . The data has been sourced from ai-jobs.net. It includes thousands of salaries (13,972 records to be precise) for Data Science professionals around the world across several years (confusingly, not just 2024).

These are the dataset's 11 fields (with types):

work_year (temporal; the year that the salary was awarded to the employee)

experience_level (ordinal; whether the employee is junior, intermediate, senior, or executive-level)

employment_type (ordinal; whether the employee works part-time, full-time, contract, or freelance)

job_title (nominal)

salary (ratio)

salary_currency (nominal)

salary_in_usd (ratio)

employee_residence (nominal; the employee's country of residence)

remote_ratio (ratio; percentage of work that the employee completes remotely; binned as 0, 50, or 100)

company_location (nominal; the country where the company is based)

company_size (ordinal; whether the company is classified as small, medium, or large)

The main question that I would like to answer using this data is: on average, what are the best-paying Data Science jobs? I would then like to subdivide this data to contrast the salaries of early, mid, and late-career professionals. Lastly, I would like to see whether the amount of work done remotely has an effect on the received pay. With this in mind, I will narrow down this data and focus on the 2023 US job market (as 2024 does not have as many varied records yet), focusing exclusively on full-time employees. I also decided to remove executive-level employees' salaries as there were not enough of them in the dataset and the ones that were present tended to skew the analysis results towards the upper managerial positions.

With that in mind, let us formalize our 3 primary tasks:

1. Present the 10 highest-paying Data Science jobs.

The **goal** here is to present the 10 highest-paying Data Science professions. The **means** through which this will be achieved is organizing the salary data by profession, calculating the averages, and sorting said professions by their respective averages in descending order, presenting the first 10 using a bar chart. We are therefore seeking to find and present the high-level **characteristics** of our data in form of professions with the highest average associated salaries. Our **target data** operates within the absolute reference frame as we are calculating and presenting fixed targets in form of average salaries for different available professions. In terms of the **workflow**, this task ought to come first as the others will build upon our findings here. The **role** of the individual executing the task could be an analyst interested in gaining knowledge about the trends within the Data Science job space.

2. Using the previous task's outcome, contrast the salaries of junior, intermediate, and senior-level employees.

The **goal** here is to use the first task's findings and contrast the salaries of different seniority level employees. The **means** through which this will be achieved is exploring how the pay differs between different seniority levels within each profession by interactively updating a second bar chart to display the average salaries of junior, intermediate, and senior-level employees within each profession after clicking on the relevant profession's bar in the original bar chart. Once again, we are presenting the high-level data **characteristics** in form of the salary distributions within the ten highest-paying professions. Again, our **target data** operates within the absolute reference frame as we are calculating fixed targets in form of average salaries for different seniority levels. In terms of the **workflow**, this task will come second as it requires the results of the initial task to work. The **role** of the individual executing it could once again be an analyst, this time seeking deeper insight into our data.

3. Using the first task's outcome once more, contrast the salaries of employees whose work is done 0, 50, or 100 percent remotely.

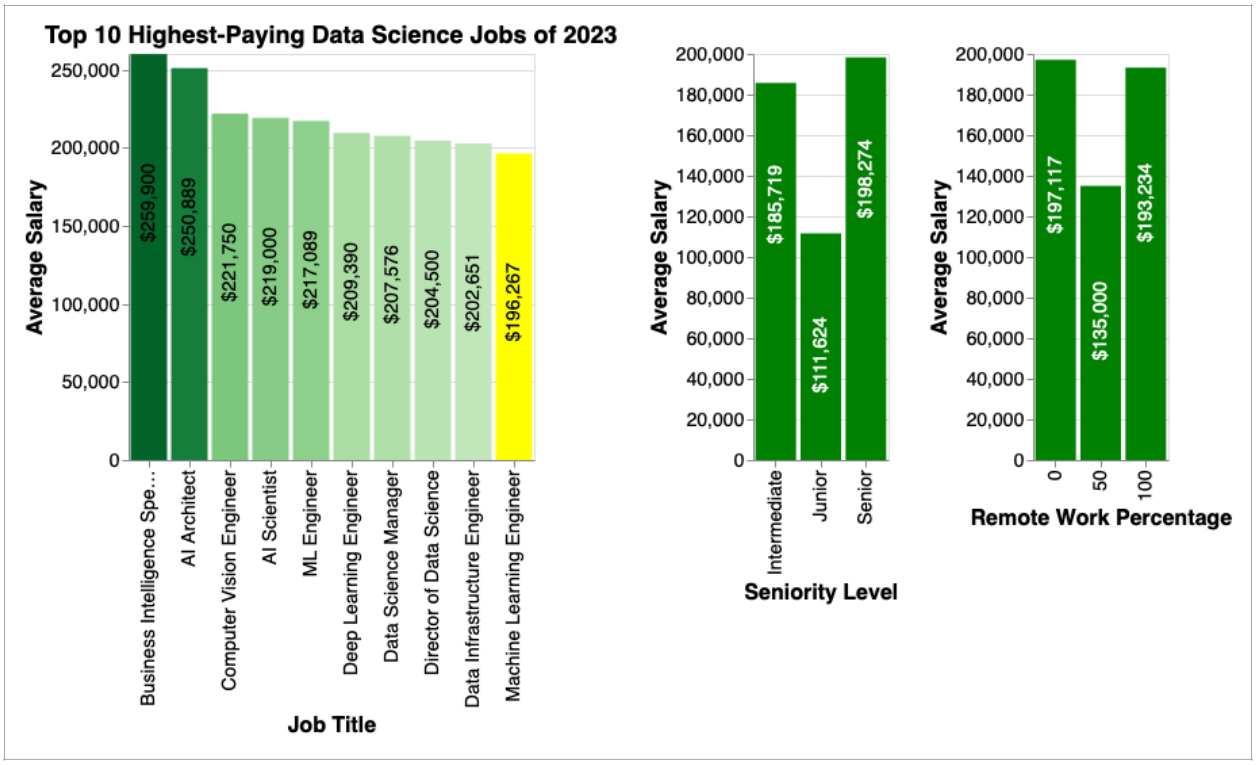
The third task is similar to the second. The **goal** is to use the initial task's findings and contrast the salaries of employees working remotely 0, 50, or 100 percent of the time. The **means** through which this will be achieved is exploring how the pay differs between different percentages of remote work within each profession by interactively updating another bar chart to display the average salaries of employees with differing percentages of remote work after clicking on the relevant profession's bar in the original bar chart. We are again presenting the high-level data **characteristics** in form of salary distributions within the ten highest-paying professions. Our **target data** once again operates within the absolute reference frame as we are calculating fixed targets in form of average salaries for different amounts of remote work. As for the **workflow**, this task will come third, although it could come second as it only requires the results of the initial task to work. Again, the **role** of the executing individual could be an analyst seeking further insight into our data.

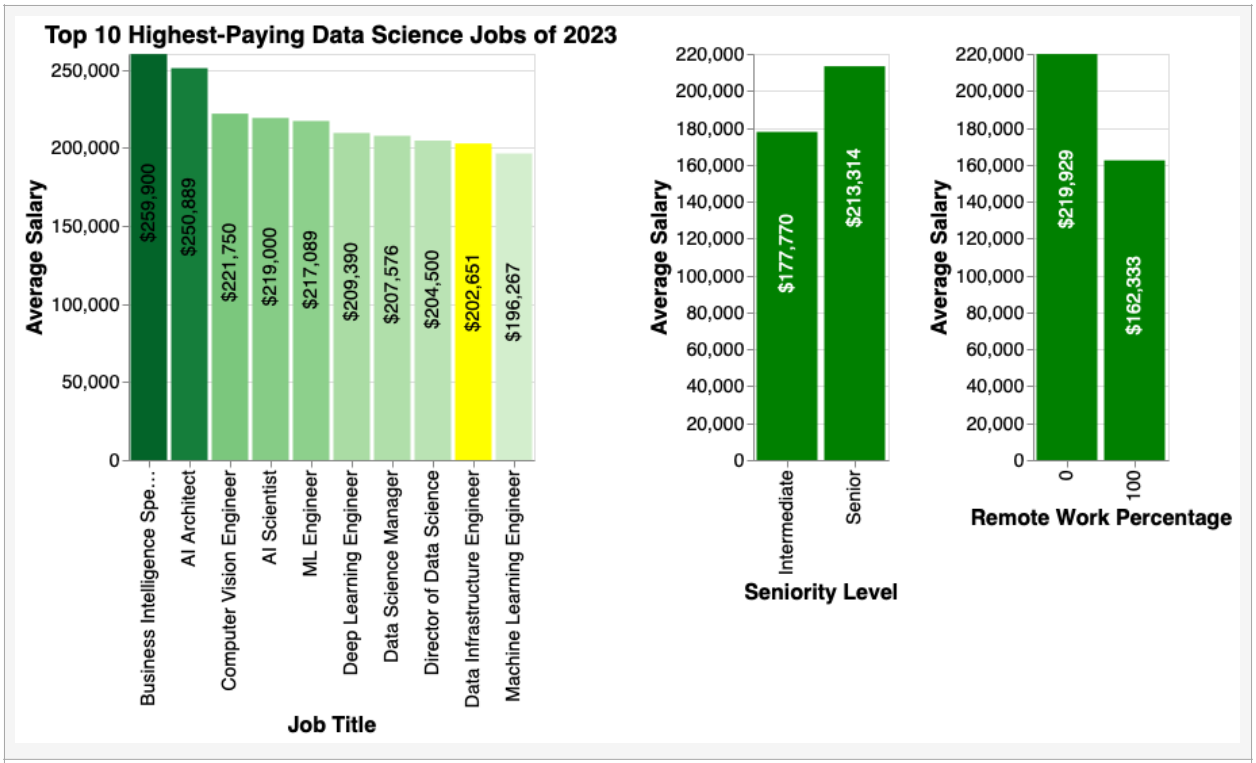
Having identified the primary tasks, let us build our visualization using Python's Pandas and Altair libraries. I will provide the code below, along with some screenshots. Feel free to skip ahead and read the evaluation section.

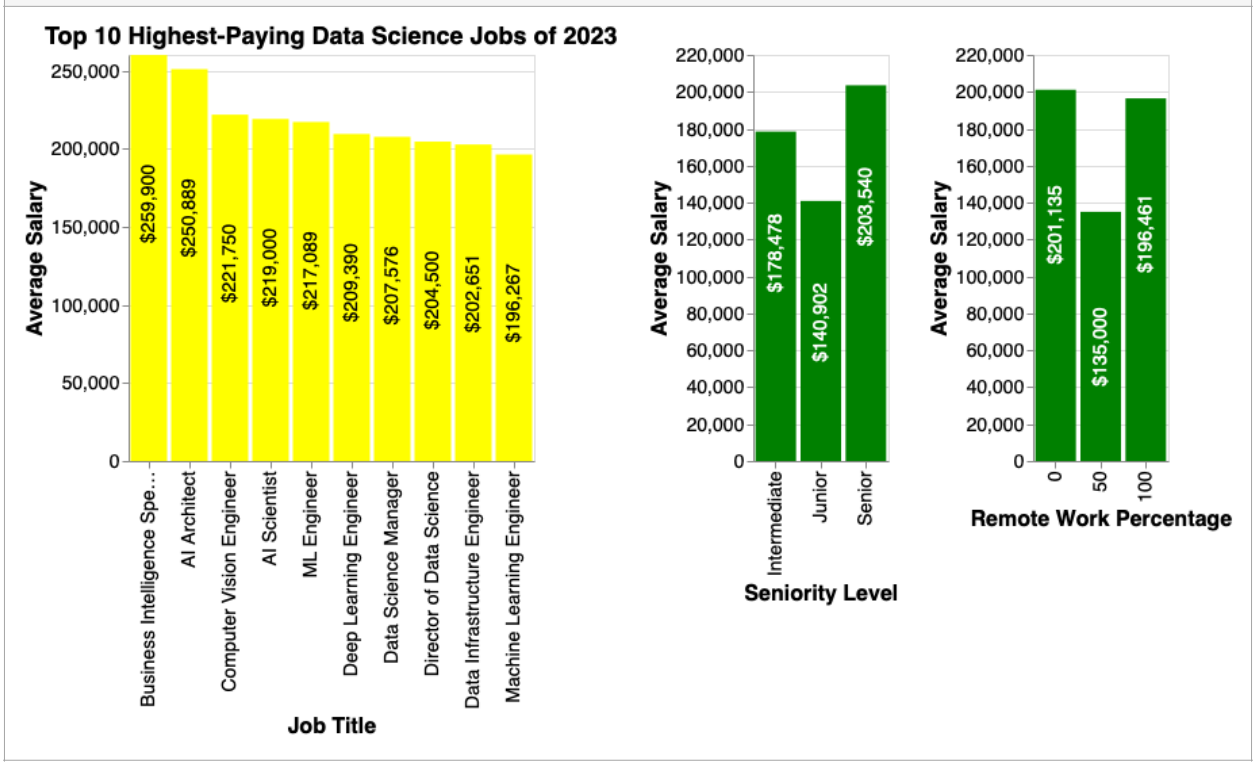
```
import pandas as pd
import altair as alt

# import data
salaries = pd.read_csv('salaries.csv')
# only include salaries from 2023
salaries_2023 = salaries[salaries['work_year'] == 2023]
# exclude executive-level (EX) salaries
salaries_2023_nonex = salaries_2023[salaries_2023['experience_level'] !=
'EX']
# only include salaries for full-time (FT) employees
salaries_2023_nonex_ft =
salaries_2023_nonex[salaries_2023_nonex['employment_type'] == 'FT']
# only include salaries for US-based employees
salaries_2023_nonex_ft_us =
salaries_2023_nonex_ft[salaries_2023_nonex_ft['employee_residence'] == 'US']
# alias experience levels with more readable labels
salaries_2023_nonex_ft_us['experience_level'] =
salaries_2023_nonex_ft_us['experience_level'].replace({'EN': 'Junior', 'MI':
'Intermediate', 'SE': 'Senior'})
# randomly sample 5000 entries (row limit for Altair)
salaries_2023_nonex_ft_us_rand = salaries_2023_nonex_ft_us.sample(5000,
random_state=1)

# get 10 jobs with highest average salaries
salaries_mean = salaries_2023_nonex_ft_us_rand.groupby('job_title')
['salary_in_usd'].mean().reset_index()
salaries_mean_top10 = salaries_mean.nlargest(10, 'salary_in_usd')

# ready interactivity to make bars clickable
click = alt.selection_multi(fields=['job_title'], init=[{'job_title':
'Machine Learning Engineer'}])

# visualize the 10 highest-paying jobs as a descending bar chart; this
relates to task 1
chart_jobs = alt.Chart(salaries_mean_top10).mark_bar().encode(
    x=alt.X('job_title', title='Job Title', sort=None),
    y=alt.Y('salary_in_usd', title='Average Salary'),
    color=alt.condition(
        click,
        alt.value('yellow'),
        alt.Color('salary_in_usd:Q', scale=alt.Scale(scheme='greens'),
legend=None)
    )
)
# overlay the bars with their respective average salaries
chart_jobs_overlays = chart_jobs.mark_text(
    fontSize=14,
    angle=270,
    dx=-110
).encode(
    text=alt.Text('salary_in_usd', format='$,d'),
```

```
    color=alt.value('black')
)
chart_jobs_with_overlays = chart_jobs + chart_jobs_overlays
chart_jobs_with_overlays =
chart_jobs_with_overlays.add_selection(click).properties(
    title='Top 10 Highest-Paying Data Science Jobs of 2023',
    width=300,
    height=300
)

# add a 2nd bar chart that breaks down the job's respective average salary by
the employees' seniority levels; this relates to task 2
chart_seniority =
alt.Chart(salaries_2023_nonex_ft_us_rand).transform_filter(click).mark_bar().
encode(
    x=alt.X('experience_level', title='Seniority Level'),
    y=alt.Y('mean(salary_in_usd)', title='Average Salary'),
    color=alt.value('green')
).transform_aggregate(
    groupby=['experience_level'],
    salary_in_usd='mean(salary_in_usd)'
)
# overlay the bars with their respective average salaries
chart_seniority_overlays = chart_seniority.mark_text(
    fontSize=14,
    angle=270,
    dx=-90
).encode(
    text=alt.Text('salary_in_usd', format='$,d'),
    color=alt.value('white')
)
chart_seniority_with_overlays = chart_seniority + chart_seniority_overlays
chart_seniority_with_overlays = chart_seniority_with_overlays.properties(
    width=100,
    height=300
)

# add a 3rd bar chart that breaks down the job's respective average salary by
the amount of work done remotely; this relates to task 3
chart_remoteness =
alt.Chart(salaries_2023_nonex_ft_us_rand).transform_filter(click).mark_bar().
encode(
    x=alt.X('remote_ratio:N', title='Remote Work Percentage'),
    y=alt.Y('mean(salary_in_usd)', title='Average Salary'),
    color=alt.value('green')
).transform_aggregate(
    groupby=['remote_ratio'],
    salary_in_usd='mean(salary_in_usd)'
)
# overlay the bars with their respective average salaries
chart_remoteness_overlays = chart_remoteness.mark_text(
    fontSize=14,
    angle=270,
    dx=-100
).encode(
    text=alt.Text('salary_in_usd', format='$,d'),
    color=alt.value('white')
)
```

```
chart_remoteness_with_overlays = chart_remoteness + chart_remoteness_overlays
chart_remoteness_with_overlays = chart_remoteness_with_overlays.properties(
    width=100,
    height=300
)

# combine our 3 charts to create the final visualization
chart_combined = alt.hconcat(chart_jobs_with_overlays, 
chart_seniority_with_overlays, chart_remoteness_with_overlays)
chart_combined = chart_combined.configure_title(
    fontSize=18
).configure_axis(
    titleFontSize=16,
    labelFontSize=14
)
```




Top 10 Highest-Paying Data Science Jobs of 2023
Average Salary
250,000
200,000
150,000
100,000
50,000
0
$259,900
$250,889
$221,750
$219,000
$217,089
$209,390
$207,576
$204,500
$202,651
$196,267
Business Intelligence Spe...
AI Architect
Computer Vision Engineer
AI Scientist
ML Engineer
Deep Learning Engineer
Data Science Manager
Director of Data Science
Data Infrastructure Engineer
Machine Learning Engineer
Job Title
220,000
200,000
180,000
160,000
140,000
120,000
100,000
80,000
60,000
40,000
20,000
0
$177,770
$213,314
Intermediate
Senior
Seniority Level
$219,929
$162,333
0
100
Remote Work Percentage


Top 10 Highest-Paying Data Science Jobs of 2023
Average Salary
250,000
200,000
150,000
100,000
50,000
0
$259,900
$250,889
$221,750
$219,000
$217,089
$209,390
$207,576
$204,500
$202,651
$196,267
Business Intelligence Spe...
AI Architect
Computer Vision Engineer
AI Scientist
ML Engineer
Deep Learning Engineer
Data Science Manager
Director of Data Science
Data Infrastructure Engineer
Machine Learning Engineer
Job Title
220,000
200,000
180,000
160,000
140,000
120,000
100,000
80,000
60,000
40,000
20,000
0
$178,478
$140,902
$203,540
Intermediate
Junior
Senior
Seniority Level
$201,135
$135,000
$196,461
0
50
100
Remote Work Percentage

With our visualization complete, let us examine some of its key elements and the reasoning behind them. I decided to present the data as three interconnected interactive bar charts. The bar charts were chosen due to the nature of the data that was to be presented: categorical variables on the X-axis and summarized numerical data on the Y-axis. The interconnected, interactive nature stems from the fact that each of the charts is meant to help an analyst execute one of the three above-mentioned tasks, two of which depend on the findings of the first. Hence, the bar chart on the far left presents the initial findings while the other two charts allow us to dig deeper into the data by selecting the different bars in the original chart (multi-selection is allowed). Upon selection, the bar becomes highlighted in yellow and the contents of the other charts become updated to relevant values. Additionally, the charts are layered horizontally to accommodate the modern widescreen displays with more horizontal space than vertical. Since we are dealing with salaries in USD, the bars are colored green. The yellow highlight color contrasts nicely with them while also being associated with money. In the leftmost graph, the higher the salary, the greener the bar becomes. The bars are also sorted in descending order, from tallest to shortest.

Let us now evaluate our visualization. Looking back at our tasks, the main question involved finding 10 best-paying Data Science professions. Next, we wanted to check how much (if at all) the average pay differed between the 3 seniority levels: junior, intermediate, and senior. Lastly, we wanted to do the same but for 3 levels of job remoteness: 0%, 50%, and 100%. So, in an actual study, we would recruit several analysts and see what sort of insights they could draw using our visualization. However, for simplicity sake, I decided to carry out a mock evaluation using 3 individuals: an ex-colleague who works as a Data Analyst, a friend who works retail but is studying to become a Data Scientist, and my father who has no knowledge of or interest in Data Science. I decided that insight-based holistic evaluation would work well in this scenario. Being presented with the above visualization, would my three "analysts" be able to answer the above-stated questions in an efficient manner? To see if they would, I conducted three think-aloud studies with the analysts' insights becoming my unit of evaluation. Time to, number of, and importance of insights were all taken into account. The results of the evaluation were interesting:

The ex-colleague navigated the familiar-looking bar charts quickly and effortlessly. He immediately identified the 10 highest-paying professions and started clicking on different bars to explore the differences between seniority levels and work remoteness. He said that the fact that not every profession included data for all three seniority or remoteness levels made results a bit less interesting and the questions a bit vague. Still, when the data was available, it seemed that the pay usually increased with seniority (although there were a couple outliers that he attributed to lack of variance in the original dataset) while non-remote positions seemed to, in general, pay better. He said that the results were quite predictable.

My friend also quickly identified the 10 highest-paying professions. Wanting to enter the field herself, she found the data interesting. However, it took her a bit longer to figure out the interactive and interconnected nature of the three bar charts. She said that a

prompt would have been helpful. She was also dissatisfied with some higher-paying professions only including data for senior employees. Still, she arrived to the same conclusions as the previous analyst.

Lastly, my father found it difficult to navigate the bar charts. He identified the 10 highest-paying professions rather quickly but had to be guided through answering the other two questions.

The evaluation results were quite sobering. Both the tasks as well as the visualization itself had gone through numerous iterations to reach the current state. I had initially planned on using matrix plots but found them too cluttered. The interactive bar charts were meant to be far more accessible. However, it seems that I had developed a tunnel vision of sorts as non-experts still found it confusing to navigate. Still, the leftmost bar chart proved very effective in answering the primary question of identifying best-paying professions. Furthermore, I would like to return to this analysis in the future using a more complete and varied data set. Also, I would like to build another chart to plot the growth (or lack thereof) of salaries for different professions over time. Such information ought to prove useful. With all of this in mind, I believe that the resulting visualization does what any decent initial iteration should, inviting further inquiry and improvement.