Mollie Harreys

Project 1
DSE5002

Given Scenario

Your CEO has decided that the company needs a full-time data scientist, and possibly a team of them in the future. She thinks she needs someone who can help drive data science within the entire organization and could potentially lead a team in the future. She understands that data scientist salaries vary widely worldwide and is unsure what to pay them. To complicate matters, salaries are going up due to the great recession and the market is highly competitive. Your CEO has asked you to prepare an analysis on data science salaries and provide them with a range to be competitive and get top talent. The position can work offshore, but the CEO would like to know what the difference is for a person working in the United States. Your company is currently a small company but is expanding rapidly.

Instructions: In an RMD notebook, do all of the following:

A.) Restate the question in your own words -create several alternative ways to ask this question - think about what you could add to the question, or remove from it

B.) You are being given the data set, you don't need to find another one, or consider other sources in this project. But you do need to do: 1.) Load the data into a dataframe 2.) Make sure all the variable types are set correctly. Think about whether any of the variables should be factors (indicating membership in a group). Use the factor or as.factor function to change these all to factors.

Questions restated in my own words:
Can you find the average pay for a data scientist and give me a range that could attract top talent?
What is the average pay for a lead data scientist in the US vs International?
Does a domestic or international position make more sense based on the pay?
Since we prefer a US-based candidate, what is the competitive pay range for a US position?

Below, we set up the libraries we will use throughout the project:

In [435…
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import os
from plotnine import *
#from plotnine import ggplot, aes, geom_col, labs, geom_text, position_dodge, theme, facet_grid, facet_wrap, ggsave
import seaborn as sns
import matplotlib
import matplotlib.pyplot as plt
import plotly.express as px
import pycountry

%matplotlib inline
#display Matplotlib plots directly beneath the code
```

In [436…
```
# check on the current working directory
os.getcwd()
```

Out[436… '/Users/mollieharreys/Desktop/Merrimack/Python and SQL/Project_01'

In [437…
```
infile="project_1_data.csv"
#setting the provided data set as a variable

salary=pd.read_csv(infile)
#putting the dataset into a pandas data frame and naming the data frame salary
```

In [438…
```
salary.head(10)
#view the imported data in a padas dataframe - head shows the first 10 rows
```

Out[438…

	Unnamed: 0	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
0	0	2020	MI	FT	Data Scientist	70000	EUR	79833	DE	0	DE	L
1	1	2020	SE	FT	Machine Learning Scientist	260000	USD	260000	JP	0	JP	S
2	2	2020	SE	FT	Big Data Engineer	85000	GBP	109024	GB	50	GB	M
3	3	2020	MI	FT	Product Data Analyst	20000	USD	20000	HN	0	HN	S
4	4	2020	SE	FT	Machine Learning Engineer	150000	USD	150000	US	50	US	L
5	5	2020	EN	FT	Data Analyst	72000	USD	72000	US	100	US	L
6	6	2020	SE	FT	Lead Data Scientist	190000	USD	190000	US	100	US	S
7	7	2020	MI	FT	Data Scientist	11000000	HUF	35735	HU	50	HU	L
8	8	2020	MI	FT	Business Data Analyst	135000	USD	135000	US	100	US	L
9	9	2020	SE	FT	Lead Data Engineer	125000	USD	125000	NZ	50	NZ	S

The column "Unnamed: 0" is confusing
This seems to be a number coordinating to each case/individual
Let's rename this to the variable to "employee"

In [439…
```
salary = salary.rename(columns={'Unnamed: 0': 'employee'}) #rename column to employee
salary.head(10)
```

Out[439…

	employee	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
0	0	2020	MI	FT	Data Scientist	70000	EUR	79833	DE	0	DE	L
1	1	2020	SE	FT	Machine Learning Scientist	260000	USD	260000	JP	0	JP	S
2	2	2020	SE	FT	Big Data Engineer	85000	GBP	109024	GB	50	GB	M
3	3	2020	MI	FT	Product Data Analyst	20000	USD	20000	HN	0	HN	S
4	4	2020	SE	FT	Machine Learning Engineer	150000	USD	150000	US	50	US	L
5	5	2020	EN	FT	Data Analyst	72000	USD	72000	US	100	US	L
6	6	2020	SE	FT	Lead Data Scientist	190000	USD	190000	US	100	US	S
7	7	2020	MI	FT	Data Scientist	11000000	HUF	35735	HU	50	HU	L
8	8	2020	MI	FT	Business Data Analyst	135000	USD	135000	US	100	US	L
9	9	2020	SE	FT	Lead Data Engineer	125000	USD	125000	NZ	50	NZ	S

Should any of these variables be set as a factor? Lets see what each variable already is.

In [440…
```
#check the variable types
salary.dtypes
```

Out[440…
```
employee               int64
work_year              int64
experience_level      object
employment_type       object
job_title             object
salary                 int64
salary_currency       object
salary_in_usd          int64
employee_residence    object
remote_ratio           int64
company_location      object
company_size          object
dtype: object
```

We see above none are factors - They are integers or objects

Lets think - which of these are catagorical?

Catagorical variables -> work_year, experience_level, employment_type, job_title, salary_currency, employee_residence, company_location

In [441…
```
#set as catagorical variables -> work_year, experience_level, employment_type,
#job_title, salary_currency, employee_residence, company_location, remote_ratio, company_size
salary['work_year']=pd.Categorical(salary.work_year)
salary['experience_level']=pd.Categorical(salary.experience_level)
salary['employment_type']=pd.Categorical(salary.employment_type)
salary['job_title']=pd.Categorical(salary.job_title)
salary['salary_currency']=pd.Categorical(salary.salary_currency)
salary['employee_residence']=pd.Categorical(salary.employee_residence)
salary['company_location']=pd.Categorical(salary.company_location)
salary['remote_ratio']=pd.Categorical(salary.remote_ratio)
salary['company_size']=pd.Categorical(salary.company_size)

#check the types again to see if the changes worked
salary.dtypes
```

Out[441…
```
employee                 int64
work_year             category
experience_level      category
employment_type       category
job_title             category
salary                   int64
salary_currency       category
salary_in_usd            int64
employee_residence    category
remote_ratio          category
company_location      category
company_size          category
dtype: object
```

Above we now see that we have successfully changed the selected variables to categories - A majority of these are catagorical

---

3.) Do a basis exploratory data analysis

-use summary and comment on what you see for each variable

-use one other table, or summary, that produces a text output for each variable

-do at least one plot of some sort per variable

In [442…
```
#Dimensions
print(f"Rows and Columns: {salary.shape}")

#Missing Values (Total per column)
print(salary.isna().sum())

#Check missing values
print(f"Any missing values? {salary.isna().values.any()}")
```

```
Rows and Columns: (607, 12)
employee              0
work_year             0
experience_level      0
employment_type       0
job_title             0
salary                0
salary_currency       0
salary_in_usd         0
employee_residence    0
remote_ratio          0
company_location      0
company_size          0
dtype: int64
Any missing values? False
```

There are 607 rows and 12 Columns with no missing values in any column.

Below I am generating a statistical summary of only the numerical variables.
I chose to drop employee due to calulations on the numbering of these being irrelivant

In [443…
```
salary.drop(['employee'], axis=1).describe()
#only statistical variables removing employee column
```

Out[443…

	salary	salary_in_usd
count	6.070000e+02	607.000000
mean	3.240001e+05	112297.869852
std	1.544357e+06	70957.259411
min	4.000000e+03	2859.000000
25%	7.000000e+04	62726.000000
50%	1.150000e+05	101570.000000
75%	1.650000e+05	150000.000000
max	3.040000e+07	600000.000000

Above we see a statistical summery of each of the numerical variables - the salary_in_usd is what we shoudl pay attention to
Below I want to see what the distinct catagories that are in each catagorical variable

In [444…
```
# return unique values in each column
years_distinct = salary['work_year'].unique()
experience_distinct = salary['experience_level'].unique()
employment_distinct = salary['employment_type'].unique()
title_distinct = salary['job_title'].unique()
currency_distinct = salary['salary_currency'].unique()
residence_distinct = salary['employee_residence'].unique()
company_location_distinct = salary['company_location'].unique()
company_size_distinct = salary['company_size'].unique()
remote_ratio_distinct = salary['remote_ratio'].unique()
company_size_distinct = salary['company_size'].unique()


print("Years:",years_distinct)
print()
print("Experience Level:",experience_distinct)
print()
print("Employment Types:", employment_distinct)
print()
print("Titles",title_distinct)
print()
print("Currency Types:", currency_distinct)
print()
print("Employee Residences",residence_distinct)
print()
print("Company Locations:",company_location_distinct)
print()
print("Company Sizes:",company_size_distinct)
print()
print("Remote Ratios:",remote_ratio_distinct)
print()
print("Company Sizes:",company_size_distinct)

#print() inbetween results for ease of readability when referencing it later
```

```
Years: [2020, 2021, 2022]
Categories (3, int64): [2020, 2021, 2022]

Experience Level: ['MI', 'SE', 'EN', 'EX']
Categories (4, object): ['EN', 'EX', 'MI', 'SE']

Employment Types: ['FT', 'CT', 'PT', 'FL']
Categories (4, object): ['CT', 'FL', 'FT', 'PT']

Titles ['Data Scientist', 'Machine Learning Scientist', 'Big Data Engineer', 'Product Data Analyst', 'Machine Learning Engineer', ..., 'ETL Developer', 'Head of Machine
Learning', 'NLP Engineer', 'Lead Machine Learning Engineer', 'Data Analytics Lead']
Length: 50
Categories (50, object): ['3D Computer Vision Researcher', 'AI Scientist', 'Analytics Engineer', 'Applied Data Scientist', ..., 'Principal Data Scientist', 'Product Dat
a Analyst', 'Research Scientist', 'Staff Data Scientist']

Currency Types: ['EUR', 'USD', 'GBP', 'HUF', 'INR', ..., 'CLP', 'BRL', 'TRY', 'AUD', 'CHF']
Length: 17
Categories (17, object): ['AUD', 'BRL', 'CAD', 'CHF', ..., 'PLN', 'SGD', 'TRY', 'USD']

Employee Residences ['DE', 'JP', 'GB', 'HN', 'US', ..., 'EE', 'AU', 'BO', 'IE', 'CH']
Length: 57
Categories (57, object): ['AE', 'AR', 'AT', 'AU', ..., 'TR', 'UA', 'US', 'VN']

Company Locations: ['DE', 'JP', 'GB', 'HN', 'US', ..., 'DZ', 'EE', 'MY', 'AU', 'IE']
Length: 50
Categories (50, object): ['AE', 'AS', 'AT', 'AU', ..., 'TR', 'UA', 'US', 'VN']

Company Sizes: ['L', 'S', 'M']
Categories (3, object): ['L', 'M', 'S']

Remote Ratios: [0, 50, 100]
Categories (3, int64): [0, 50, 100]

Company Sizes: ['L', 'S', 'M']
Categories (3, object): ['L', 'M', 'S']
```

What we can see about each variable based on the above and referencing the Meta Data Sheet

employee - This labels each case with a number. Seemingly irrelevant as the row titles already number each row. This column also starts at 0 throwing off the count so it no longer lines up with the row titles, why would there be a case 0? This is not an infection tracking dataset. We can disregard or remove this column
work_year - 3 distinct years - 2020, 2021, 2022
experience_level - 4 Distinct Experience Levels - 'MI', 'SE', 'EN', 'EX' possible values: EN Entry-level / Junior MI Mid-level / Intermediate SE Senior-level / Expert EX Executive-level / Director. *Director is not shown in this data set*
employment_type - 4 distinct types - PT Part-time FT Full-time CT Contract FL Freelance
job_title - 50 distinct types - Name of Job title held that year
salary - total gross salary - varies in terms of currency
salary_currency - 17 distinct values - Currency type
salary_in_usd - All salaries listed in US dollars
employee_residence - 57 distinct types - abreviations of countries
remote_ratio - 3 distinct values - 0 No remote work (less than 20%) 50 Partially remote 100 Fully remote (more than 80%)
company_location - 50 distinct values - abreviations of countries
company_size - 3 distinct values - S less than 50 employees (small) M 50 to 250 employees (medium) L more than 250 employees (large)

Data Exploration on Original Data

In [445…
```
salary.columns
```

Out[445…
```
Index(['employee', 'work_year', 'experience_level', 'employment_type',
       'job_title', 'salary', 'salary_currency', 'salary_in_usd',
       'employee_residence', 'remote_ratio', 'company_location',
       'company_size'],
      dtype='object')
```

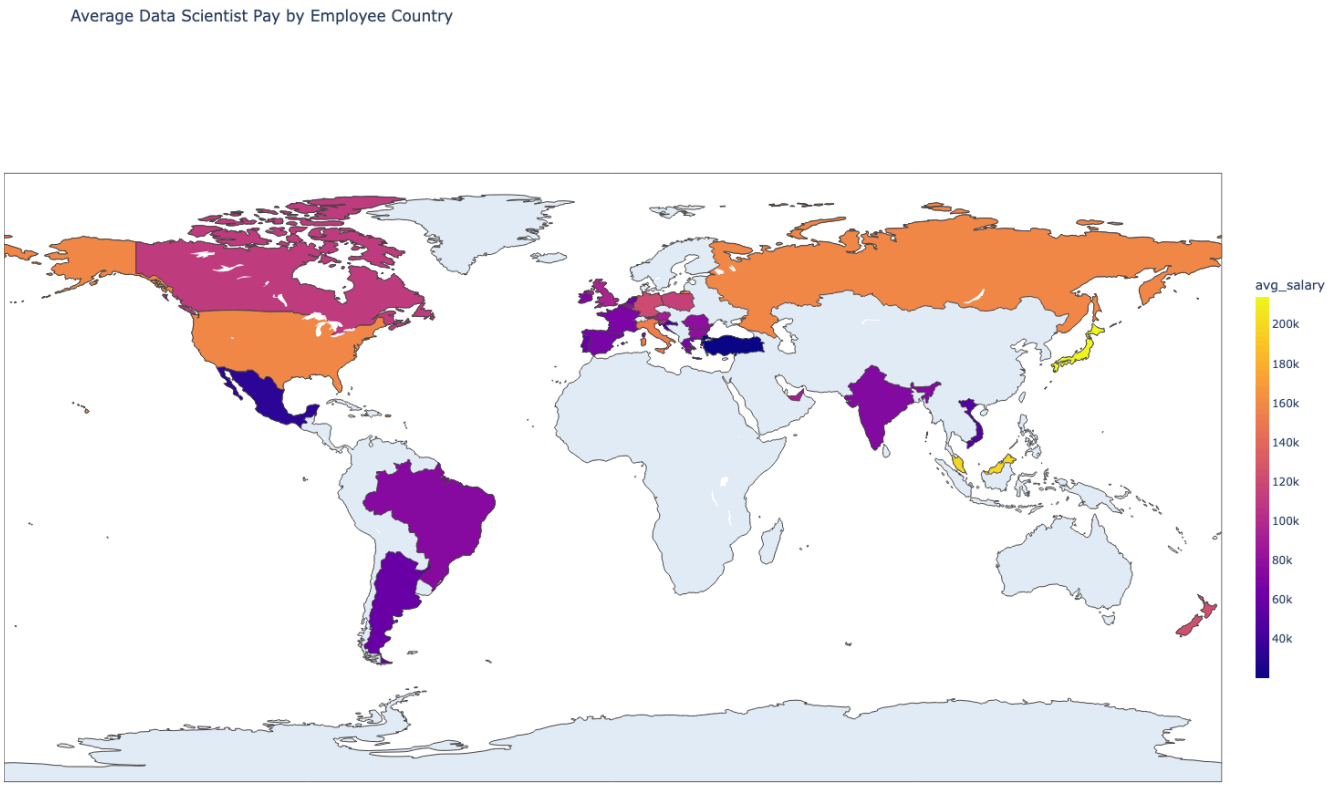
Plotting average pay by employee residence on an interactive map

In [446…
```
#function to find the corresponding ISO code for each coutry. We currently have a two letter abreviation
def to_iso3(name):
    try:
        return pycountry.countries.lookup(name).alpha_3
    except:
        return None

senior_fulltime_salary["iso_alpha"] = senior_fulltime_salary["employee_residence"].apply(to_iso3)
```

In [447…
```
#making a dataframe to use to plot the avergae pay - group by residence and the new ISO denotation for each country
avg_emp_pay = (
    senior_fulltime_salary
    .groupby(
        ["employee_residence", "iso_alpha"],
        as_index=False,
        observed=True
    )["salary_in_usd"]
    .mean()
    .rename(columns={"salary_in_usd": "avg_salary"})
)
```

In [448…
```
#creating a map made from an example on the plotly library site
#this is a choropleth map aka color coded map
#we will show the average salary on a map of only fulltime senior or expert employees

salary_map_emp = px.choropleth(
    avg_emp_pay,
    locations="iso_alpha",
    color="avg_salary",
    hover_name="employee_residence",
    color_continuous_scale=px.colors.sequential.Plasma,
    title="Average Data Scientist Pay by Employee Country"
)

# precise values in hover
salary_map_emp.update_traces(
    hovertemplate="<b>%{hovertext}</b><br>Average Salary: $%{z:,.2f}<extra></extra>"
```

```
)

#map was small - adjusting map frame result below
salary_map_emp.update_layout(
    height=1000,
    width=1000,
    margin={"r":50, "t":80, "l":0, "b":0}
)

salary_map_emp.update_coloraxes(
    colorbar=dict(
        thickness=15,   # width of legend
        len=0.5,        # height of legend
        x=1.02,         # pushes it outside map
        y=0.5
    )
)

salary_map_emp.show()
```



In [449…

```
avg_emp_pay[["employee_residence", "iso_alpha"]].head(5)
#had issues with conversions - used dataframe with converted titles - now ISO understands the full name vs the two letter acronym
```

Out[449…

	employee_residence	iso_alpha
0	United Arab Emirates	ARE
1	Argentina	ARG
2	Austria	AUT
3	Belgium	BEL
4	Bulgaria	BGR

In [450…

```
Employee_Residence_avg_pay_focused_data = average_salary_by(senior_fulltime_salary, 'employee_residence')
# comparing the average that shows up in the map to what the function returns
#previous version was showing wrong numbers - IDO conversion was the issue
Employee_Residence_avg_pay_focused_data
```

Out[450…

	employee_residence	salary_in_usd
16	Japan	214000.000000
18	Malaysia	200000.000000
22	Puerto Rico	160000.000000
28	United States	159403.468750
25	Russian Federation	157500.000000
15	Italy	153667.000000
20	New Zealand	125000.000000
7	Germany	120232.555556
21	Poland	114047.000000
6	Canada	110188.312500
26	Slovenia	102839.000000
10	United Kingdom	94477.000000
0	United Arab Emirates	92500.000000
2	Austria	91237.000000
3	Belgium	82744.000000
4	Bulgaria	80000.000000
24	Romania	76833.000000
5	Brazil	75726.750000
14	India	72710.714286
13	Ireland	71444.000000
9	France	68808.800000
11	Greece	68327.000000
8	Spain	67416.500000
19	Netherlands	62651.000000
23	Portugal	60757.000000
1	Argentina	60000.000000
29	Viet Nam	50000.000000
12	Croatia	45618.000000
17	Mexico	33511.000000
27	Türkiye	20171.000000

In [451…
```
#Employee_Residence_avg_pay_focused_data .to_csv("Employee_Residence_avg_pay_focused_data .csv", index=False)
```

In [452…
```
#salary_map_emp.write_html("salary_map_employee_pay_EX_SR_Fulltime.html")
```

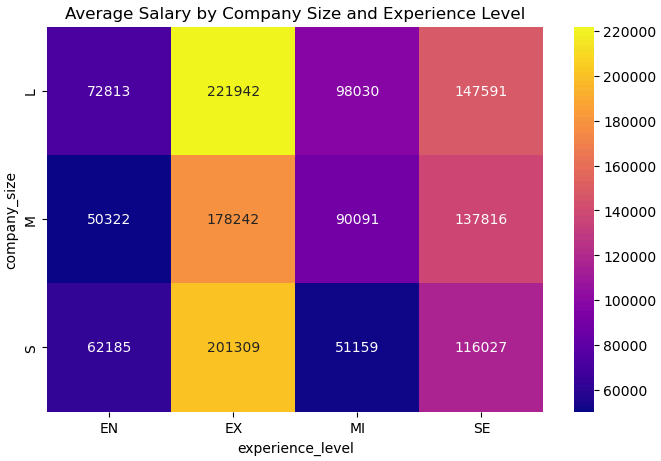
In [453…
```
#salary_map_emp.write_image("salary_map_employee_pay_EX_SR_Fulltime.svg")
```

Plotting Average Pay by Company Size and Experience Level

In [454…
```
#new data frame with just companysize experience level and finding the average pay for those
size_experience_avg_pay = (
    salary
    .groupby(["company_size", "experience_level"], as_index=False, observed=True)["salary_in_usd"]
    .mean()
)
```

In [455…
```
#pivot data to be able to use it in a heatmap style plot
heatmap_data = size_experience_avg_pay.pivot(
    index="company_size",
    columns="experience_level",
    values="salary_in_usd"
)
```

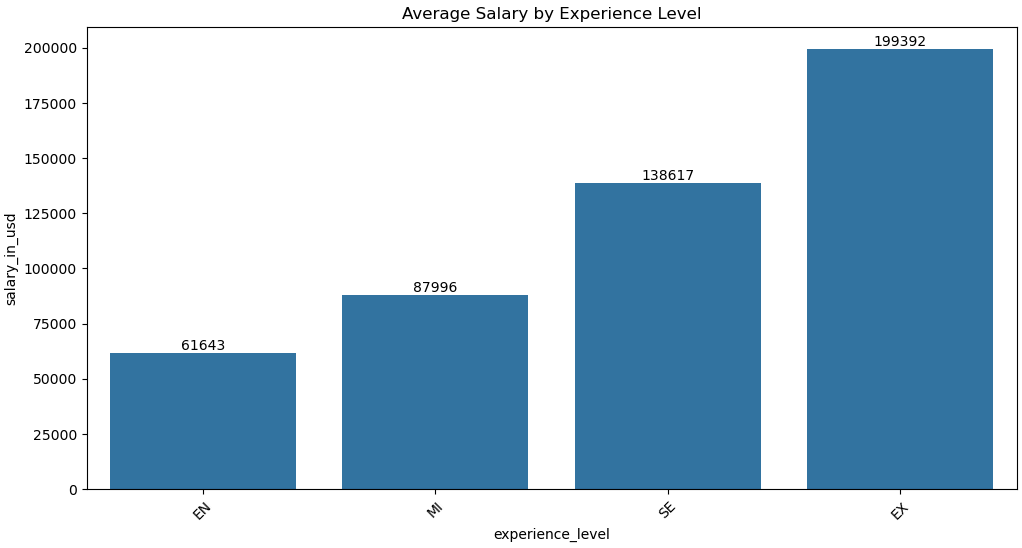
In [498…
```
#make the map with pvioted data
plt.figure(figsize=(8,5))

sns.heatmap(
    heatmap_data,
    annot=True, #write value on each cube
    fmt=".0f", #Show the number as a whole integer (no decimals).
    cmap="plasma" #map stype from seaborn
)


plt.title("Average Salary by Company Size and Experience Level")
#plt.savefig("heatmap_avg_salary.svg", bbox_inches="tight")
plt.show()
```



Higher pay is correlated with higher experience levels more than company size

In [499…
```
#new data frame with just experience level and finding the mean for each
experience_level_avg_pay = (
    salary
    .groupby(["experience_level"], as_index=False, observed=True)["salary_in_usd"]
    .mean()
)
```

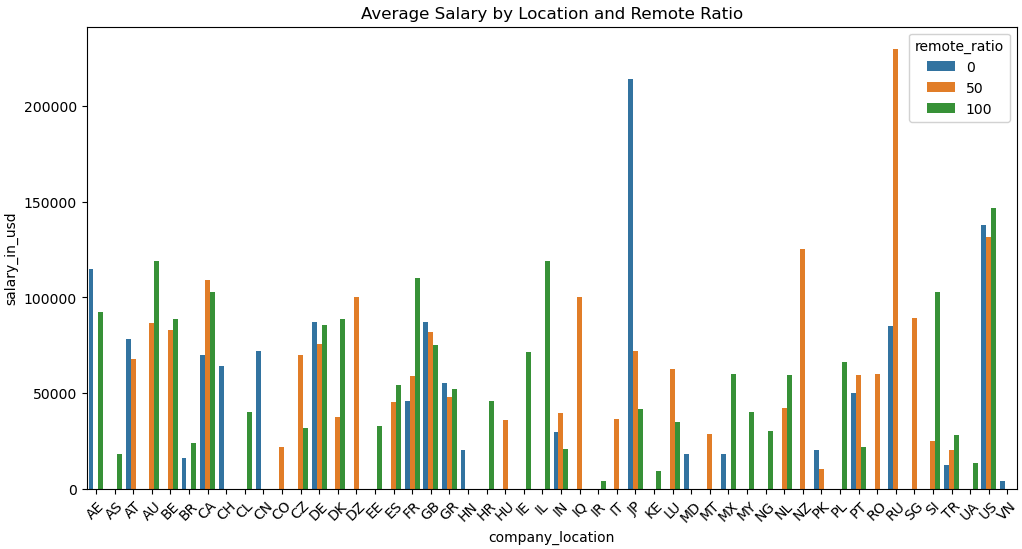
In [500…
```
order = ["EN", "MI", "SE", "EX"] #put in order of experience

plt.figure(figsize=(12,6))

pay_level = sns.barplot(
    data=experience_level_avg_pay,
    x="experience_level",
    y="salary_in_usd",
    order=order
)
pay_level.bar_label(pay_level.containers[0], fmt="%.0f")#display number above each column
plt.xticks(rotation=45)
plt.title("Average Salary by Experience Level")

#plt.savefig("experience_level_salary.svg", bbox_inches="tight") #save fig as svg - commented out so it doesn't save every time I rerun the file
plt.show()
```



Company Location and Remote Ratio Average Salary

In [501…
```
#new data frame with just company location and remote ratio and finding the average pay for those
location_remote_ratio_avg_pay = (
    salary
    .groupby(["company_location", "remote_ratio"], as_index=False, observed=True)["salary_in_usd"]
    .mean()
)
```

In [502…
```
plt.figure(figsize=(12,6))

sns.barplot(
    data=location_remote_ratio_avg_pay,
    x="company_location",
    y="salary_in_usd",
    hue="remote_ratio"
)

plt.xticks(rotation=45)#rotate labels so they dont overlap
plt.title("Average Salary by Location and Remote Ratio")

plt.show()
```



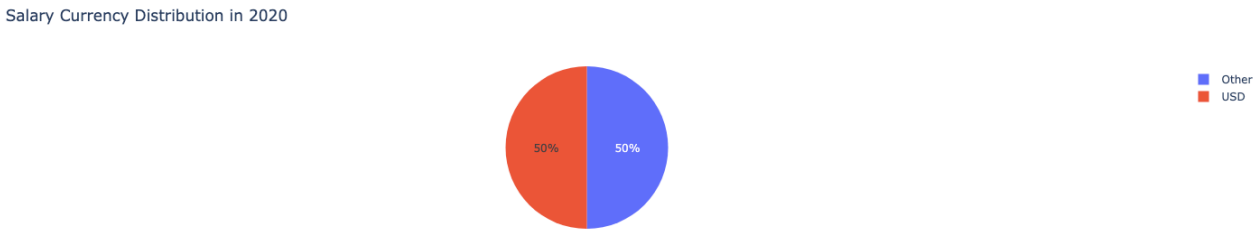
In [503…
```
#plt.savefig("Bar_Location_remote_ratio_avg_pay_all.svg", format="svg")
```

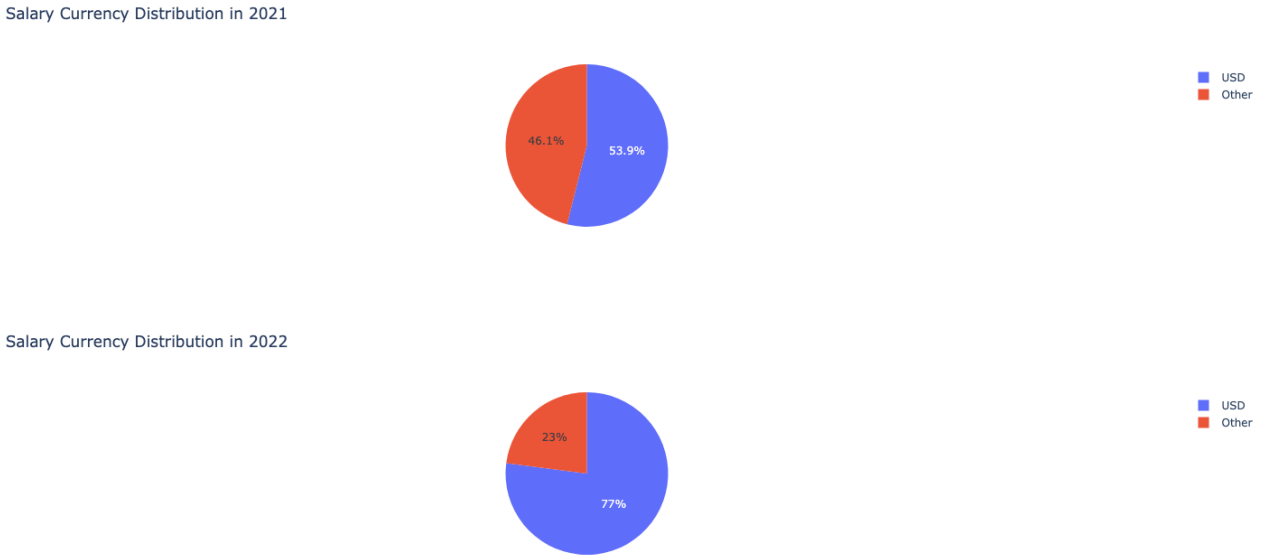
Comparing how many pay currencies are in USD vs others and its changes over 3 years

In [504…
```
#counting how many times each currency appears in each year
currency_counts = (
    salary
    .groupby(["work_year", "salary_currency"], observed=True)
    .size()
    .reset_index(name="count"))# converts that result from a series into a dataframe
currency_counts.columns = ["work_year", "currency", "count"]#renames colunmns for legibility
```

In [505…
```
#catagoizes each currency as either USD or other
pie_data = salary.copy()

pie_data["currency_group"] = pie_data["salary_currency"].apply(
    lambda x: "USD" if x == "USD" else "Other"
)
#group into year and salary type then count the number of salary types in each year
yearly = (
    pie_data.groupby(["work_year", "currency_group"], observed=True)
    .size()
    .reset_index(name="count")
)
```

In [506…
```
#take the categorized currencies and plots them into a pie chart for each yea
#now we can see if there has been a change in what currency
for year in sorted(yearly["work_year"].unique()):
    temp = yearly[yearly["work_year"] == year]

    salary_pie = px.pie(
        temp,
        names="currency_group",
        values="count",
        title=f"Salary Currency Distribution in {year}"
    )
    #salary_pie.write_image(f"salary_pie_{year}.svg") #indenting both of these keeps them in the loop
    salary_pie.show()
```





In [507…
```
pip install -U kaleido
```

```
Requirement already satisfied: kaleido in /opt/anaconda3/lib/python3.13/site-packages (1.3.0)
Requirement already satisfied: choreographer>=1.3.0 in /opt/anaconda3/lib/python3.13/site-packages (from kaleido) (1.3.0)
Requirement already satisfied: logistro>=1.0.8 in /opt/anaconda3/lib/python3.13/site-packages (from kaleido) (2.0.1)
Requirement already satisfied: orjson>=3.10.15 in /opt/anaconda3/lib/python3.13/site-packages (from kaleido) (3.11.9)
Requirement already satisfied: packaging in /opt/anaconda3/lib/python3.13/site-packages (from kaleido) (25.0)
Requirement already satisfied: platformdirs>=4.3.6 in /opt/anaconda3/lib/python3.13/site-packages (from choreographer>=1.3.0->kaleido) (4.5.0)
Requirement already satisfied: simplejson>=3.19.3 in /opt/anaconda3/lib/python3.13/site-packages (from choreographer>=1.3.0->kaleido) (4.1.1)
Note: you may need to restart the kernel to use updated packages.
```

In [508…
```
#salary_pie.write_image(f"salary_pie_by_year.svg") #exporting this way only return the last item in the loop
```

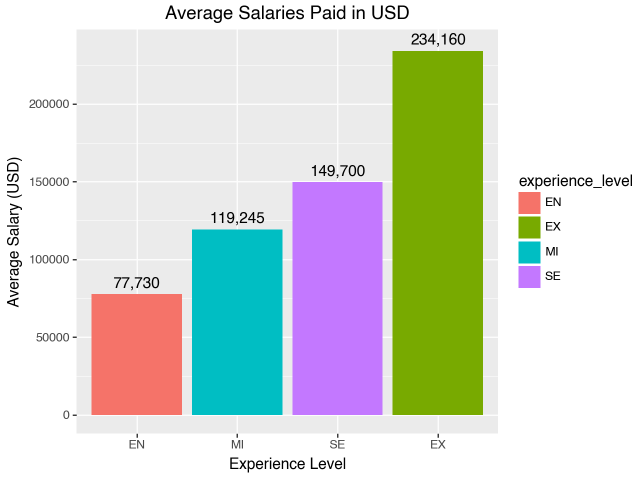
We can see that there has been a large increase in USD paid salaries from 2020 to 2022 **a 27% increase in USD pay over 2 years**

How does only USD pay imact pay averages? Since we see above USD pay is becoming more prevalent

In [509…
```
usd_only = (
    salary[salary["salary_currency"] == "USD"]
    .groupby("experience_level", observed=True, as_index=False)["salary_in_usd"]
    .mean()
)
```

In [510…
```
order = ["EN", "MI", "SE", "EX"] #ordered experience levels

average_pay_in_USD_Only_Experience = (
    ggplot(
        usd_only,
        aes(
            x="experience_level",
            y="salary_in_usd",
            fill="experience_level"
        )
    )
    + geom_col()
    + geom_text(
        aes(label="round(salary_in_usd, 0)"), #whole numbers
        va="bottom", #alignment of text
        nudge_y=2000,
        format_string="{:,.0f}" #write average above each column with a comma so it's more readable
    )
    + scale_x_discrete(limits=order)
    + labs(
        title="Average Salaries Paid in USD",
        x="Experience Level",
        y="Average Salary (USD)"
    )
)

average_pay_in_USD_Only_Experience
```

Out [510…



In [511…

```
#ggsave(average_pay_in_USD_Only_Experience, "average_pay_in_USD_Only_Experience.svg")
```

When considering the prevalence of pay being in USD, which is mosl likley what we will pay. Viewing the averages of the pay that originate in USD gives us a perspective on what would be likely expected from a US company.

Top Job titles in each Job Type

In [512…

```
#find the most frequesntly occuring job titles in each employment type
top_jobs = (
    salary
    .groupby(["employment_type", "job_title"], observed=True)
    .size()
    .reset_index(name="count")
)

#return the top 3 of each job title in each employment type
top_3 = (
    top_jobs
    .sort_values(["employment_type", "count"], ascending=[True, False])
    .groupby("employment_type", observed=True)
    .head(3)
)

top_3
```

Out [512…

	employment_type	job_title	count
0	CT	Applied Machine Learning Scientist	1
1	CT	Business Data Analyst	1
2	CT	ML Engineer	1
5	FL	Computer Vision Engineer	1
6	FL	Data Engineer	1
7	FL	Data Scientist	1
30	FT	Data Scientist	140
25	FT	Data Engineer	129
20	FT	Data Analyst	96
58	PT	AI Scientist	2
61	PT	Data Engineer	2
62	PT	Data Scientist	2

In [513…

```
#top_3.to_csv("top_3_jobs_by_employment_type.csv", index=False)
```

Deeper Data exploration

Let's think through what variables can be compared and computed to weed out important and unimportant data.

Ideas

- Seperate out only the employees that are full time - we want a full time employees - Maybe make this a new data frame to compute/graph questions below
- Job titles that are full time
- compare pay accross expereince levels
- View titles within each expereince level - see what is similar to the titles we want
- Find average pay of all other counties compare to US average
- Pay as compares to "remoteness" of company
- See how pay changes on average over the 3 years - "rate of increase"
- Currency type and level of pay
- Residence vs pay
- Residence vs remote ratio - not crucial but could be interesting
- Company size vs company pay
- medium companies pay us vs international (since we are small but rapidly growing)

In [514…

```
#Seperate out only the employees that are full time - we want a full time employees - Maybe make this a new data frame to compute/graph questions below
fulltime_salary = salary[salary['employment_type'] == 'FT'].copy()
```

```
fulltime_salary.head(10)
```

Out[514…

	employee	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
0	0	2020	MI	FT	Data Scientist	70000	EUR	79833	DE	0	DE	L
1	1	2020	SE	FT	Machine Learning Scientist	260000	USD	260000	JP	0	JP	S
2	2	2020	SE	FT	Big Data Engineer	85000	GBP	109024	GB	50	GB	M
3	3	2020	MI	FT	Product Data Analyst	20000	USD	20000	HN	0	HN	S
4	4	2020	SE	FT	Machine Learning Engineer	150000	USD	150000	US	50	US	L
5	5	2020	EN	FT	Data Analyst	72000	USD	72000	US	100	US	L
6	6	2020	SE	FT	Lead Data Scientist	190000	USD	190000	US	100	US	S
7	7	2020	MI	FT	Data Scientist	11000000	HUF	35735	HU	50	HU	L
8	8	2020	MI	FT	Business Data Analyst	135000	USD	135000	US	100	US	L
9	9	2020	SE	FT	Lead Data Engineer	125000	USD	125000	NZ	50	NZ	S

In [515…

```
FT_title_distinct = fulltime_salary['job_title'].unique()
print("Distinct Full-time Titles",FT_title_distinct)
```

```
Distinct Full-time Titles ['Data Scientist', 'Machine Learning Scientist', 'Big Data Engineer', 'Product Data Analyst', 'Machine Learning Engineer', ..., 'ETL Developer', 'Head of Machine Learning', 'NLP Engineer', 'Lead Machine Learning Engineer', 'Data Analytics Lead']
Length: 48
Categories (50, object): ['3D Computer Vision Researcher', 'AI Scientist', 'Analytics Engineer', 'Applied Data Scientist', ..., 'Principal Data Scientist', 'Product Data Analyst', 'Research Scientist', 'Staff Data Scientist']
```

There are only two fewer titles in full time positions compared to all types of potitions. Lets look at the job titles of full time employees that are Senior-level and Expert/Executive-level. Since we want someone who can lead data science for the whole company and possible recruit and lead a team in the future.

In [516…

```
senior_fulltime_salary = fulltime_salary[fulltime_salary['experience_level'].isin(['SE', 'EX'])].copy()
senior_fulltime_salary.head(10)
```

Out[516…

	employee	work_year	experience_level	employment_type	job_title	salary	salary_currency	salary_in_usd	employee_residence	remote_ratio	company_location	company_size
1	1	2020	SE	FT	Machine Learning Scientist	260000	USD	260000	JP	0	JP	S
2	2	2020	SE	FT	Big Data Engineer	85000	GBP	109024	GB	50	GB	M
4	4	2020	SE	FT	Machine Learning Engineer	150000	USD	150000	US	50	US	L
6	6	2020	SE	FT	Lead Data Scientist	190000	USD	190000	US	100	US	S
9	9	2020	SE	FT	Lead Data Engineer	125000	USD	125000	NZ	50	NZ	S
17	17	2020	SE	FT	Big Data Engineer	100000	EUR	114047	PL	100	GB	S
22	22	2020	SE	FT	Data Engineer	42000	EUR	47899	GR	50	GR	L
25	25	2020	EX	FT	Director of Data Science	325000	USD	325000	US	100	US	L
27	27	2020	SE	FT	Data Engineer	720000	MXN	33511	MX	0	MX	S
29	29	2020	SE	FT	Machine Learning Manager	157000	CAD	117104	CA	50	CA	L

Now we have a dataset that contains fulltime Senior and executive empoyee data.

In [517…

```
senior_FT_title_distinct = senior_fulltime_salary['job_title'].unique()
print("Distinct Full-time Senior or Excecutive Titles",senior_FT_title_distinct)
```

```
Distinct Full-time Senior or Excecutive Titles ['Machine Learning Scientist', 'Big Data Engineer', 'Machine Learning Engineer', 'Lead Data Scientist', 'Lead Data Engineer', ..., 'Head of Machine Learning', 'Lead Machine Learning Engineer', 'Machine Learning Developer', 'Applied Data Scientist', 'Data Analytics Lead']
Length: 41
Categories (50, object): ['3D Computer Vision Researcher', 'AI Scientist', 'Analytics Engineer', 'Applied Data Scientist', ..., 'Principal Data Scientist', 'Product Data Analyst', 'Research Scientist', 'Staff Data Scientist']
```

Now we have 41 distinct titles compared to the original 50 - not a giant differance. We now know title vary widley accross all data science positions and levels

In [518…

```
#lets try a this previous code on our newly filtered dataset, only viewing the US currency.
fulltime_stats = senior_fulltime_salary.drop(['employee','salary'], axis=1).describe()
fulltime_stats
```

Out[518…

	salary_in_usd
count	303.000000
mean	143287.244224
std	65012.187251
min	18907.000000
25%	100400.000000
50%	137141.000000
75%	174500.000000
max	600000.000000

In [519…
```
fulltime_stats.to_csv("fulltime_stats_senior_exec.csv", index=True)
```

In [520…
```
#in the below the removed experience levels were still showing up - adding this in here makes sure to remove them from the catagorical option
senior_fulltime_salary['experience_level'] = (senior_fulltime_salary['experience_level'].cat.remove_unused_categories())
```

In [521…
```
#checking that there are only two catagories
senior_fulltime_salary['experience_level'].unique()
```

Out[521…
```
['SE', 'EX']
Categories (2, object): ['EX', 'SE']
```

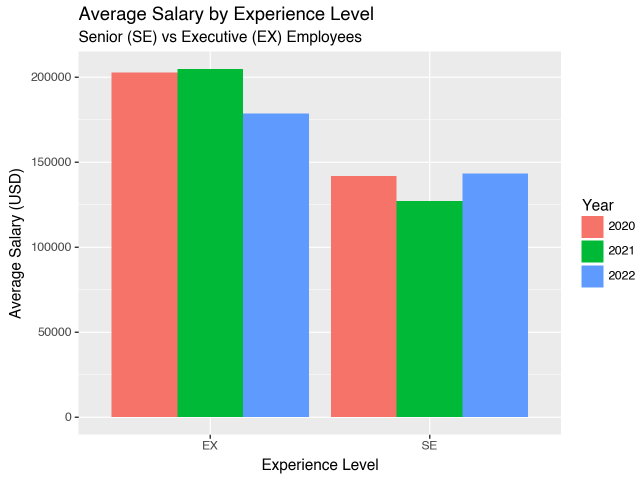
In [522…
```
avg_salary_year_experience = (
    senior_fulltime_salary
    .groupby(['experience_level', 'work_year'], observed = True)['salary_in_usd']
    .mean()
    .reset_index()
)

avg_salary_year_experience.head()
```

Out[522…

	experience_level	work_year	salary_in_usd
0	EX	2020	202416.500000
1	EX	2021	204528.000000
2	EX	2022	178313.846154
3	SE	2020	141784.058824
4	SE	2021	126913.779412

In [523…
```
(ggplot( avg_salary_year_experience, aes(
            x='experience_level',
            y='salary_in_usd',
            fill='factor(work_year)'
        )) + geom_col(position='dodge')+ labs(
        title='Average Salary by Experience Level',
        subtitle='Senior (SE) vs Executive (EX) Employees',
        x='Experience Level',
        y='Average Salary (USD)',
        fill='Year'
    ))
```

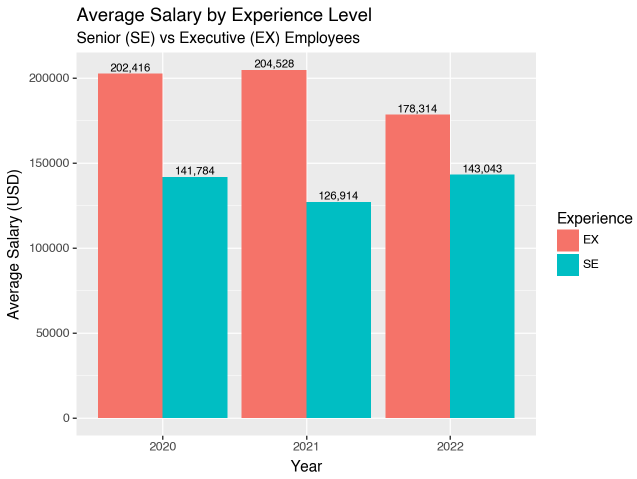
Out[523…



Another way of plotting below switching the work_year and experience_level and adding the number of the average above each bar

In [524…
```
(ggplot(avg_salary_year_experience, aes(
            x='work_year',
            y='salary_in_usd',
            fill='experience_level'
        )) + geom_col(position='dodge')  + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=8,
        format_string='{:,.0f}'
    )+ labs(
        title='Average Salary by Experience Level',
        subtitle='Senior (SE) vs Executive (EX) Employees',
        x='Year',
        y='Average Salary (USD)',
        fill='Experience'
    ))
```

Out[524…



Seems to be that EX level average pay slightly increases then unltimely decreased and the SE level did the opposite.
EX levels stay firmly higher than SE levels even as their average reduced in 2022 Their average pay gap shinks in 2022

Lets try making a function to define the average salary by any variable we choose

In [525…
```
def average_salary_by(df, variable):
    return (
        df.groupby(variable, observed=True)['salary_in_usd']
        .mean()
        .reset_index()
        .sort_values('salary_in_usd', ascending=False)
    )
```

In [526…
```
average_salary_by(senior_fulltime_salary, 'experience_level')
```

Out[526…

	experience_level	salary_in_usd
0	EX	190727.720000
1	SE	139021.014388

In [527…
```
!pip install pycountry
import pycountry
#installing a package called pycountry to turn country abbreviations to full names
```

```
Requirement already satisfied: pycountry in /opt/anaconda3/lib/python3.13/site-packages (26.2.16)
```

In [528…
```
def code_to_country(code):#country code abbreviation into full country name
    try:
        return pycountry.countries.get(alpha_2=code).name
        #compare codes in the library to those in the data frame and return the corresponding full name to replace country abbreviation
    except:
        return code #otherwise return the same abbrevirion input
```

In [529…
```
senior_fulltime_salary['employee_residence'] = (senior_fulltime_salary['employee_residence'].apply(code_to_country))
#apply the function to the employee residence column
```

In [530…
```
average_salary_by(senior_fulltime_salary, 'employee_residence')
#now we can easily tell which country has which pay
```

Out[530…

	employee_residence	salary_in_usd
16	Japan	214000.000000
18	Malaysia	200000.000000
22	Puerto Rico	160000.000000
28	United States	159403.468750
25	Russian Federation	157500.000000
15	Italy	153667.000000
20	New Zealand	125000.000000
7	Germany	120232.555556
21	Poland	114047.000000
6	Canada	110188.312500
26	Slovenia	102839.000000
10	United Kingdom	94477.000000
0	United Arab Emirates	92500.000000
2	Austria	91237.000000
3	Belgium	82744.000000
4	Bulgaria	80000.000000
24	Romania	76833.000000
5	Brazil	75726.750000
14	India	72710.714286
13	Ireland	71444.000000
9	France	68808.800000
11	Greece	68327.000000
8	Spain	67416.500000
19	Netherlands	62651.000000
23	Portugal	60757.000000
1	Argentina	60000.000000
29	Viet Nam	50000.000000
12	Croatia	45618.000000
17	Mexico	33511.000000
27	Türkiye	20171.000000

In [531…
```
#we can use this same function on the company country!
senior_fulltime_salary['company_location'] = (senior_fulltime_salary['company_location'].apply(code_to_country))
```

In [532…
```
average_salary_by(senior_fulltime_salary, 'company_location')
#now we can easily tell where each company is, along with their average salary
```

Out[532…

	company_location	salary_in_usd
14	Japan	214000.000000
20	Russian Federation	157500.000000
23	United States	157367.528139
18	Poland	153667.000000
17	New Zealand	125000.000000
5	Germany	120232.555556
4	Canada	116941.941176
21	Slovenia	102839.000000
8	France	94075.750000
0	United Arab Emirates	92500.000000
1	Austria	91237.000000
9	United Kingdom	90931.083333
6	Denmark	88654.000000
2	Belgium	82744.000000
12	Ireland	71444.000000
7	Spain	68191.333333
16	Netherlands	62651.000000
13	India	60976.200000
19	Portugal	60757.000000
10	Greece	47899.000000
15	Mexico	46755.500000
11	Croatia	45618.000000
3	Brazil	21453.500000
22	Türkiye	20171.000000

Do the high paid company locations correspond to high paid employee residence?

Lets adjust our function to take more variables

In [533…
```
def average_salary_by_2(df, variable1, variable2):
    return (
        df.groupby([variable1, variable2], observed=True)['salary_in_usd']
```

```
        .mean()
        .reset_index()
        .sort_values('salary_in_usd', ascending=False)
    )
```

In [534…
```
#since we are focusing on the top preformers I filtered by pay above 100,000
Locations_Average_Pay = average_salary_by_2(senior_fulltime_salary, 'company_location', 'employee_residence')
Locations_Average_Pay
```

Out[534…

	company_location	employee_residence	salary_in_usd
5	Canada	United States	225000.000000
20	Japan	Japan	214000.000000
35	United States	Malaysia	200000.000000
36	United States	Puerto Rico	160000.000000
37	United States	United States	159141.337838
27	Russian Federation	Russian Federation	157500.000000
25	Poland	Italy	153667.000000
11	France	United States	152000.000000
31	United States	Brazil	130000.000000
24	New Zealand	New Zealand	125000.000000
6	Germany	Germany	120232.555556
13	United Kingdom	Poland	114047.000000
4	Canada	Canada	110188.312500
28	Slovenia	Slovenia	102839.000000
34	United States	India	102047.000000
12	United Kingdom	United Kingdom	94477.000000
0	United Arab Emirates	United Arab Emirates	92500.000000
1	Austria	Austria	91237.000000
7	Denmark	Greece	88654.000000
2	Belgium	Belgium	82744.000000
30	United States	Bulgaria	80000.000000
14	United Kingdom	Romania	76833.000000
10	France	France	74767.666667
18	Ireland	Ireland	71444.000000
9	Spain	France	69741.000000
33	United States	Greece	68428.000000
8	Spain	Spain	67416.500000
23	Netherlands	Netherlands	62651.000000
19	India	India	60976.200000
26	Portugal	Portugal	60757.000000
21	Mexico	Argentina	60000.000000
15	United Kingdom	Viet Nam	50000.000000
32	United States	France	50000.000000
16	Greece	Greece	47899.000000
17	Croatia	Croatia	45618.000000
22	Mexico	Mexico	33511.000000
3	Brazil	Brazil	21453.500000
29	Türkiye	Türkiye	20171.000000

In [535…
```
Level_Residence_Average_over1000000 = average_salary_by_2(senior_fulltime_salary[senior_fulltime_salary['salary_in_usd'] > 100000], 'experience_level', 'employee_reside
Level_Residence_Average_over1000000
```

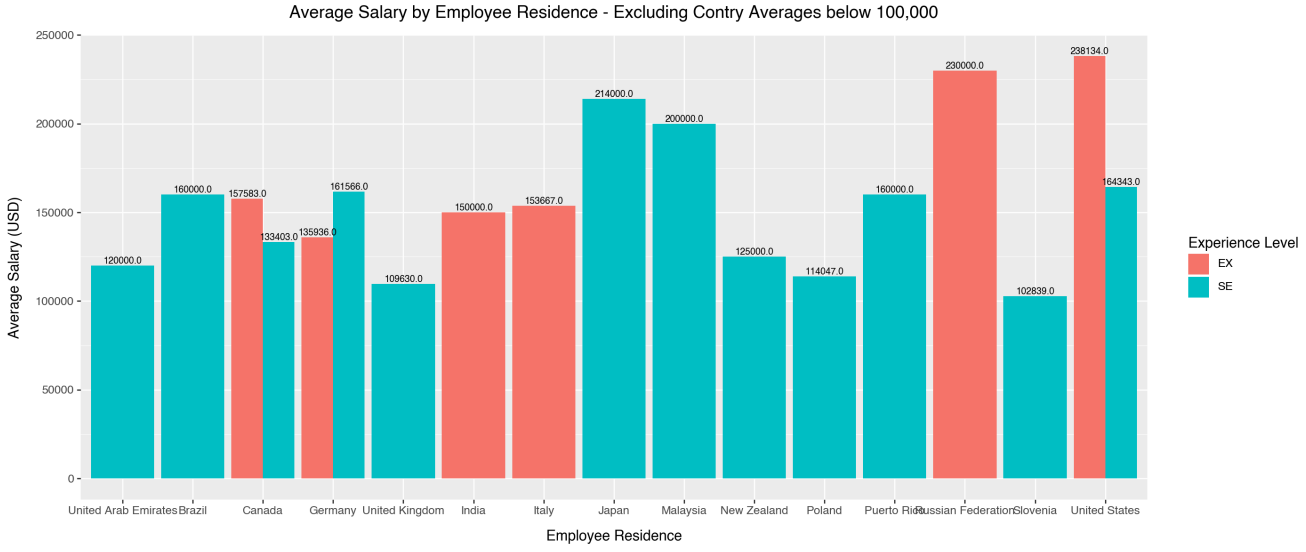
Out [535…

	experience_level	employee_residence	salary_in_usd
5	EX	United States	238133.928571
4	EX	Russian Federation	230000.000000
11	SE	Japan	214000.000000
12	SE	Malaysia	200000.000000
17	SE	United States	164343.224044
9	SE	Germany	161565.666667
7	SE	Brazil	160000.000000
15	SE	Puerto Rico	160000.000000
0	EX	Canada	157583.000000
3	EX	Italy	153667.000000
2	EX	India	150000.000000
1	EX	Germany	135936.000000
8	SE	Canada	133402.666667
13	SE	New Zealand	125000.000000
6	SE	United Arab Emirates	120000.000000
14	SE	Poland	114047.000000
10	SE	United Kingdom	109630.200000
16	SE	Slovenia	102839.000000

In [536…

```
Level_Residence_Average_over1000000 = (
    ggplot(Level_Residence_Average_over1000000, aes(
        x='employee_residence',
        y='salary_in_usd',
        fill='experience_level'
    ))
    + geom_col(position='dodge')
    + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=7
    )
    + theme(figure_size=(14, 6))
    + labs(
        title='Average Salary by Employee Residence - Excluding Contry Averages below 100,000',
        x='Employee Residence',
        y='Average Salary (USD)',
        fill='Experience Level'
    )
)
Level_Residence_Average_over1000000
```

Out [536…



In [310…

```
#ggsave(Level_Residence_Average_over1000000, "Level_Residence_Average_over1000000.svg")
```

Getting Specific - US vs International: Work and Company Combinations with Experience Level Averages

We are interested in pay where the united states is involved as either the company location or as the employee location to get a better sense of the reality of pay that we want to know.

In [538…

```
us_filtered_comp = senior_fulltime_salary[
    (senior_fulltime_salary['employee_residence'] == 'United States') &
    (senior_fulltime_salary['company_location'] == 'United States') &
    (senior_fulltime_salary['experience_level'].isin(['SE', 'EX']))
]

company_salary_us = average_salary_by_2(
    us_filtered_comp,
    'company_location',
    'experience_level'
)
```

```
company_salary_us
#we are filtering the df of senior/executive level fulltime US based employees with a us company location
#and finding the average pay for each experience level
```

Out[538…

	company_location	experience_level	salary_in_usd
0	United States	EX	238133.928571
1	United States	SE	153824.528846

In [544…
```
238133.9 - 153824.5 #difference ebtween the two averages
```

Out[544… 84309.4

In [545…
```
us_filtered_emp = senior_fulltime_salary[
    (senior_fulltime_salary['employee_residence'] == 'United States') &
    (senior_fulltime_salary['company_location'] != 'United States') &
    (senior_fulltime_salary['experience_level'].isin(['SE', 'EX']))
]

employee_salary_us= average_salary_by_2(
    us_filtered_emp,
    'company_location',
    'experience_level'
)

employee_salary_us
#we are filtering the df of senior/executive level fulltime employees that live in the US and
#work for an internationally based company and finding the average pay for each experience level
```

Out[545…

	company_location	experience_level	salary_in_usd
0	Canada	SE	225000.0
1	France	SE	152000.0

In [546…
```
Avg_US_int_Pay = (
    employee_salary_us
    .groupby("experience_level", observed=True)["salary_in_usd"]
    .mean()
    .reset_index()
)

Avg_US_int_Pay

#overall mean pay of all countries with a US employee and International Company
```

Out[546…

	experience_level	salary_in_usd
0	SE	188500.0

Average salary of an employee in the US and what a US company pays any employee (US or International) is around the same amount about $158,000

In [547…
```
int_filtered_comp = senior_fulltime_salary[
    (senior_fulltime_salary['employee_residence'] != 'United States')&
    (senior_fulltime_salary['company_location'] != 'United States') &
    (senior_fulltime_salary['experience_level'].isin(['SE', 'EX']))
]

company_salary_int = average_salary_by_2(
    int_filtered_comp,
    'company_location',
    'experience_level'
)

company_salary_int
#we are filtering the df of senior/executive level fulltime employees by the non-us company locations,
#non-US based employees finding the average pay for each country and experience level
```

Out[547…

	company_location	experience_level	salary_in_usd
18	Japan	SE	214000.000000
4	Canada	EX	157583.000000
24	Russian Federation	EX	157500.000000
22	Poland	EX	153667.000000
6	Germany	EX	135936.000000
21	New Zealand	SE	125000.000000
7	Germany	SE	115745.857143
5	Canada	SE	103417.642857
25	Slovenia	SE	102839.000000
0	United Arab Emirates	SE	92500.000000
1	Austria	SE	91237.000000
12	United Kingdom	SE	90931.083333
8	Denmark	SE	88654.000000
2	Belgium	SE	82744.000000
16	India	EX	79039.000000
9	Spain	EX	74787.000000
11	France	SE	74767.666667
15	Ireland	SE	71444.000000
20	Netherlands	SE	62651.000000
23	Portugal	SE	60757.000000
17	India	SE	56460.500000
10	Spain	SE	55000.000000
13	Greece	SE	47899.000000
19	Mexico	SE	46755.500000
14	Croatia	SE	45618.000000
3	Brazil	SE	21453.500000
26	Türkiye	SE	20171.000000

In [543…

```
#calculating averages for non-us companys with non-us employees grouped by experience level
Avg_int_int_Pay = (
    company_salary_int
    .groupby("experience_level", observed=True)["salary_in_usd"]
    .mean()
    .reset_index()
)

Avg_int_int_Pay
```

Out[543…

	experience_level	salary_in_usd
0	EX	126418.666667
1	SE	79525.988095

In [549…

```
126418.6 - 79525.9 #difference between the two averages
```

Out[549… 46892.70000000001

In [550…

```
int_filtered_comp = senior_fulltime_salary[
    (senior_fulltime_salary['employee_residence'] != 'United States') &
    (senior_fulltime_salary['company_location'] == 'United States') &
    (senior_fulltime_salary['experience_level'].isin(['SE', 'EX']))
]

emp_salary_int = average_salary_by_2(
    int_filtered_comp,
    'employee_residence',
    'experience_level'
)

emp_salary_int
#we are filtering the df of senior/executive level fulltime employees where the
#company is US based but the employee is international and finding the average pay by country
```

Out[550…

	employee_residence	experience_level	salary_in_usd
6	Malaysia	SE	200000.0
7	Puerto Rico	SE	160000.0
4	India	EX	150000.0
1	Brazil	SE	130000.0
0	Bulgaria	SE	80000.0
3	Greece	SE	68428.0
5	India	SE	54094.0
2	France	SE	50000.0

In [319…

```
#finding average of all non us companies with us employees grouped by experience level
Avg_int_US_Pay = (
    emp_salary_int
    .groupby("experience_level", observed=True)["salary_in_usd"]
    .mean()
    .reset_index()
)

Avg_int_US_Pay
```

Out[319…

	experience_level	salary_in_usd
0	EX	150000.000000
1	SE	106074.571429

In [551…
```
150000.0 - 106074.5 #difference ebtween the two averages
```

Out[551… 43925.5

Observations
Range for US empoyee and US company - **153,824 - 238,133**
SR(153,824) and EX(238,133) difference - 84,309.4

Range for US employee and International company - **152,000 - 225,000**
SR(188,500) and EX(N/A) difference - SR only refer to revious range

Range for International employee and International company - **20,171 - 214,000**
SR(79,525) and EX(126,418) difference - 46,892

Range for International employee and US company - **50,000 = 200,000**
SR(106,074) and EX(150,000) difference - 43,925

Range for Intermediate or Senior Employee - 79,525 - 188,500 *note the 79,000 is an outlier average - if wishing to be competative the range would be **106,074 - 188,500**
Range for Expert or Excecutive Employee - 150,000 - 238,133

Reasonable estimate of a pay range for a Intermediate employee on their way towards an executive level with a team with competative pay based on only this chart would be **150,000 - 200,000**

Averages by other Factors

In [552…
```
#average salary of a fulltime senior emplye at a large medium and smallco company International and Domestic
average_salary_by(senior_fulltime_salary, 'company_size')
```

Out[552…

	company_size	salary_in_usd
0	L	157444.783133
1	M	140444.680203
2	S	116544.173913

In [553…
```
#since we are focusing on the top preformers I filtered by pay above 100,000
average_salary_by_2(senior_fulltime_salary[senior_fulltime_salary['salary_in_usd'] > 100000],'company_size','employee_residence')
```

Out[553…

	company_size	employee_residence	salary_in_usd
4	L	Russian Federation	230000.000000
16	S	Japan	214000.000000
6	L	United States	200072.549020
10	M	Malaysia	200000.000000
20	S	United States	170250.000000
13	S	Brazil	160000.000000
19	S	Puerto Rico	160000.000000
11	M	United States	158619.612676
8	M	Germany	153680.750000
3	L	Italy	153667.000000
7	M	Canada	150800.000000
2	L	India	150000.000000
1	L	Germany	141846.000000
0	L	Canada	136248.750000
17	S	New Zealand	125000.000000
12	S	United Arab Emirates	120000.000000
14	S	Canada	118187.000000
18	S	Poland	114047.000000
9	M	United Kingdom	111247.750000
15	S	United Kingdom	103160.000000
5	L	Slovenia	102839.000000

In [554…
```
#does company size and level of remote work effect salary?
avg_salary_remote_size = (
    senior_fulltime_salary
    .groupby(['company_size', 'remote_ratio'], observed=True)['salary_in_usd']
    .mean()
    .reset_index()
)

avg_salary_remote_size.head(10)
```

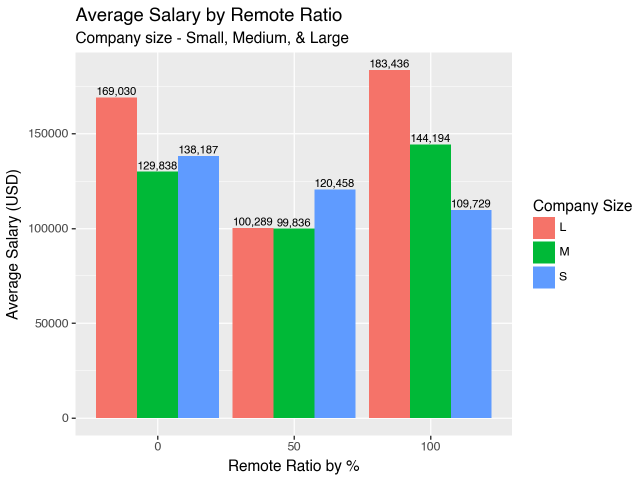
Out[554…

	company_size	remote_ratio	salary_in_usd
0	L	0	169030.058824
1	L	50	100288.565217
2	L	100	183436.488372
3	M	0	129838.500000
4	M	50	99836.200000
5	M	100	144193.814103
6	S	0	138187.000000
7	S	50	120458.250000
8	S	100	109729.000000

In [555…

```
#graphing the above chart
chart_avg_salary_remote_size = (ggplot(avg_salary_remote_size, aes(
            x='remote_ratio',
            y='salary_in_usd',
            fill='company_size'
        )) + geom_col(position='dodge')  + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=8,
        format_string='{:,.0f}'
    )+ labs(
        title='Average Salary by Remote Ratio',
        subtitle='Company size - Small, Medium, & Large',
        x='Remote Ratio by %',
        y='Average Salary (USD)',
        fill='Company Size'
    ))

chart_avg_salary_remote_size
```

Out[555…



In [556…

```
#ggsave(chart_avg_salary_remote_size, "chart_avg_salary_remote_size.svg")
```

Observations

Up to 50% remote seem to have on average lower pay across all comany sizes with a slighly higher pay for larger companies.

The largest pay is by large moslty remote companies while the lowest is semi remote medium-sized companies

Possible reasons

- lower pay if having to maintain office space
- higher pay due to larger company size

We do not specify our level of remote work but do allow for this position to be offshore - this chart does not effect much except giving possible ranges for in person and remote pay. Medium and small companies have a similar range regardless of remot work ratio -

- Large: 100,289 to 183,436
- Medium: 99,836 to 144,194
- Small: 109,729 to 138,187

Reasonable estimate of a pay range for a small company on it way to a medium company with competative pay based on only this chart would be 120,000 - 150,000

Graphs and Tables

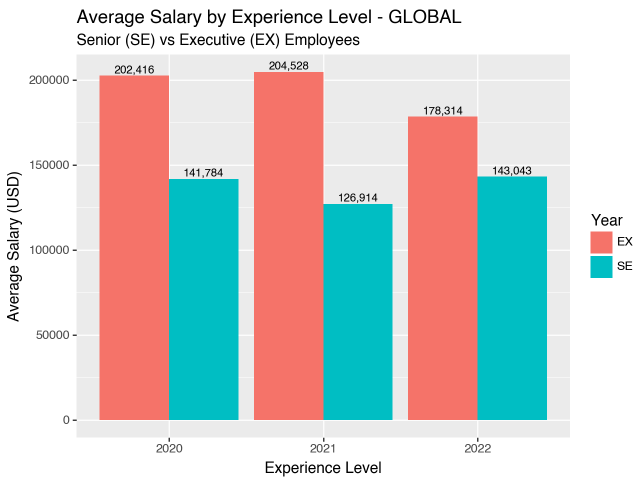
Assembled from above for ease of analysis

In [557…

```
chart_avg_salary_experience_over_time_GLOBAL = (ggplot(avg_salary_year_experience, aes(
            x='work_year',
            y='salary_in_usd',
            fill='experience_level'
        )) + geom_col(position='dodge')  + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=8,
        format_string='{:,.0f}'
    )+ labs(
        title='Average Salary by Experience Level - GLOBAL',
        subtitle='Senior (SE) vs Executive (EX) Employees',
```

```
        x='Experience Level',
        y='Average Salary (USD)',
        fill='Year'
    ))

chart_avg_salary_experience_over_time_GLOBAL
```

Out [557…



In [558…

```
#ggsave(chart_avg_salary_experience_over_time, "chart_avg_salary_experience_over_time_GLOBAL.svg")
```

Reasonable estimate of a pay range for a Intermediate employee on their way towards an executive level with a team with competative pay and based off of based on only this chart with the anticipation salaries are going up would be **140,000 - 185,000**

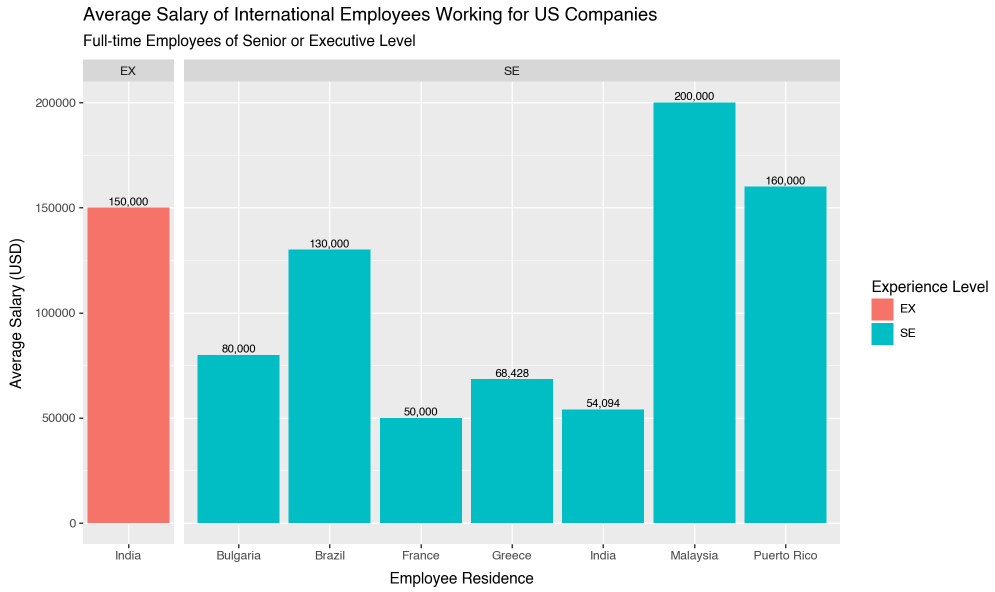
In [559…

```
chart_avg_salary_remote_size = (ggplot(avg_salary_remote_size, aes(
            x='remote_ratio',
            y='salary_in_usd',
            fill='company_size'
        )) + geom_col(position='dodge')  + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=8,
        format_string='{:,.0f}'
    )+ labs(
        title='Average Salary by Remote Ratio - GLOBAL',
        subtitle='Company size - Small, Medium, & Large',
        x='Remote Ratio by %',
        y='Average Salary (USD)',
        fill='Company Size'
    ))
```

In [560…

```
emp_salary_int
#we are filtering the df of senior/executive level fulltime employees where the company is US based but the employee is international and finding the average pay by cou

chart_avg_salary_Int_emp_US_comp = (
    ggplot(emp_salary_int, aes(
        x='employee_residence',
        y='salary_in_usd',
        fill='experience_level'
    ))
    + geom_col(position='dodge')
    + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=8,
        format_string='{:,.0f}'
    )
    + labs(
        title='Average Salary of International Employees Working for US Companies',
        subtitle='Full-time Employees of Senior or Executive Level',
        x='Employee Residence',
        y='Average Salary (USD)',
        fill='Experience Level'
    )
    + facet_grid(
        '. ~ experience_level',
        scales='free_x',
        space='free_x'
    )
    + theme(
        figure_size=(10, 6)
    )
)

chart_avg_salary_Int_emp_US_comp
```

Out[560…



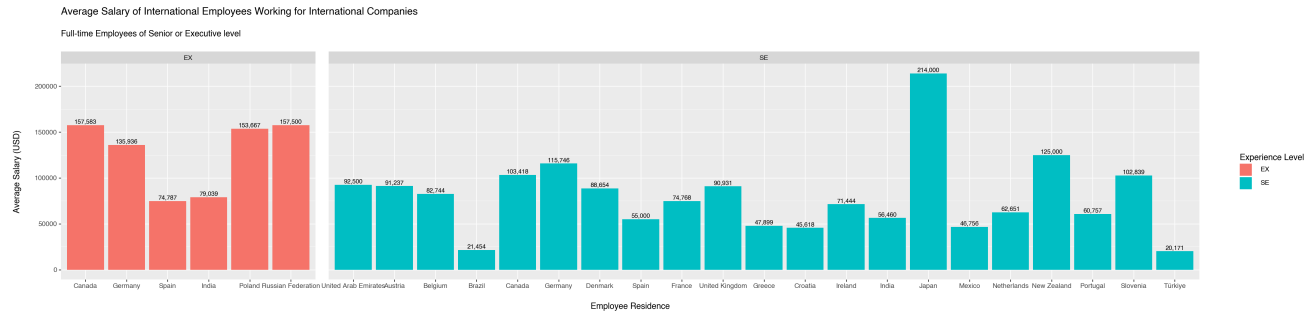
In [561…

```
#ggsave(chart_avg_salary_Int_emp_US_comp, "chart_avg_salary_Int_emp_US_comp.svg")
```

In [562…

```
company_salary_int
#we are filtering the df of senior/executive level fulltime employees by the non-us company locations, non-US based employees finding the average pay

chart_avg_salary_Int_emp_Int_comp = (
    ggplot(company_salary_int, aes(
        x='company_location',
        y='salary_in_usd',
        fill='experience_level'
    ))
    + geom_col(position='dodge')
    + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=8,
        format_string='{:,.0f}'
    )
    + labs(
        title='Average Salary of International Employees Working for International Companies',
        subtitle = 'Full-time Employees of Senior or Executive level',
        x='Employee Residence',
        y='Average Salary (USD)',
        fill='Experience Level'
    )
    + theme(
    figure_size=(24, 6))
    + facet_grid('. ~ experience_level', scales='free_x', space='free_x'
)

)

chart_avg_salary_Int_emp_Int_comp
```

Out[562…



In [563…

```
#ggsave(chart_avg_salary_Int_emp_Int_comp , "chart_avg_salary_Int_emp_Int_comp .svg")
```

In [564…

```
employee_salary_us
#we are filtering the df of senior/executive level fulltime employees that live in the US and work for an internationally based companyand finding the average pay

chart_avg_salary_US_emp_Int_comp  = (
    ggplot(employee_salary_us, aes(
        x='company_location',
        y='salary_in_usd',
        fill='experience_level'
    ))
    + geom_col(position='dodge')
    + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=8,
        format_string='{:,.0f}'
    )
    + labs(
```

```
        title= 'Average Salary of US Employees Working for International Companies',
        subtitle = 'Full-time Employees of Senior or Executive level',
        x='Employee Residence',
        y='Average Salary (USD)',
        fill='Experience Level'
    )


)

chart_avg_salary_US_emp_Int_comp
```

Out[564…

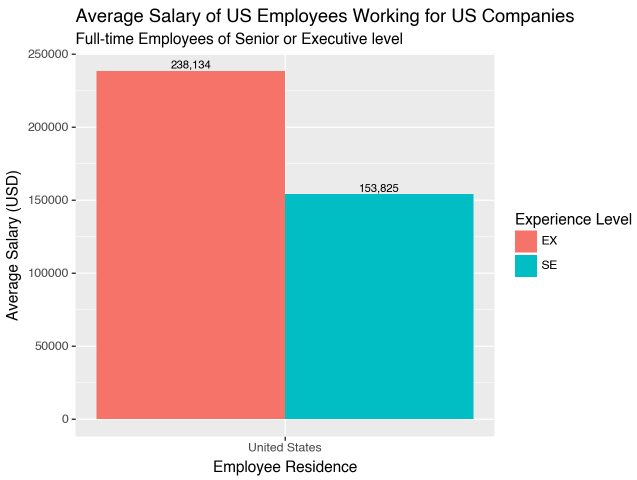
In [565…

```
#ggsave(chart_avg_salary_US_emp_Int_comp, "chart_avg_salary_US_emp_Int_comp .svg")
```

In [566…

```
company_salary_us
#we are filtering the df of senior/executive level fulltime US based employees with a US company location and finding the average pay
chart_avg_salary_US_emp_US_comp = (
    ggplot(company_salary_us, aes(
        x='company_location',
        y='salary_in_usd',
        fill='experience_level'
    ))
    + geom_col(position='dodge')
    + geom_text(
        aes(label='round(salary_in_usd, 0)'),
        position=position_dodge(width=0.9),
        va='bottom',
        size=8,
        format_string='{:,.0f}'
    )
    + labs(
        title= 'Average Salary of US Employees Working for US Companies',
        subtitle = 'Full-time Employees of Senior or Executive level',
        x='Employee Residence',
        y='Average Salary (USD)',
        fill='Experience Level'
    )


)

chart_avg_salary_US_emp_US_comp
```

Out[566…



In [567…

```
#ggsave(chart_avg_salary_US_emp_US_comp, "chart_avg_salary_US_emp_US_comp.svg")
```

C.) Prepare at least two possible tables and two possible graphs that would answer the question posed by your CEO. Using PowerPoint, do the following

D.) Prepare a deck that answers the CEO's question clearly. You are allowed a title page, and then three other pages. Your CEO likes short but complete presentations. Your presentation needs to tell a story that the CEO can make sense of. Reports on data projects always need to tell a story, most people structure the world as stories, you gotta have a story.

Write down a short paragraph in your RMD file that summarizes your story. Prepare high quality graphics in R, and/or tables as needed, paste them into PowerPoint. Let me know if you haven't used PowerPoint before. You should not have a presentation with no graphics- but it could all be graphics if you want. Just not all text

---

Observations

Comparing Average pay of Employees and Companies both Domestic and International

Observations - Senior Full time Salary Data

Overall Range for US empoyee and US company - **153,824 - 238,133**
average by experience - SR(153,824) and EX(238,133) difference - 84,309.4

Overall Range for US employee and International company - **152,000 - 225,000**
* average by experience - SR(188,500) and EX(N/A) difference - SR only refer to revious range*

Overall Range for International employee and International company - **20,171 - 214,000**
average by experience - SR(79,525) and EX(126,418) difference - 46,892

Overall Range for International employee and US company - **50,000 - 200,000**
average by experience - SR(106,074) and EX(150,000) difference - 43,925

Range for Intermediate or Senior Employee - 79,525 - 188,500
note the 79,000 is an outlier average (applying to non us comany and employee) - if wishing to be competative the range would be* **106,074 - 188,500
Range for Expert or Excecutive Employee - 150,000 - 238,133

Reasonable estimate of a pay range for a Intermediate employee on their way towards an executive level with a team with competative pay based on only this chart would be **160,000 - 210,000**

Comparing Average pay of Companies with Differing Sizes and Differing Remote Ratio

Observations - Senior Full time Salary Data

Up to 50% remote seem to have on average lower pay across all comany sizes with a slighly higher pay for larger companies.

The largest pay is by large moslty remote companies while the lowest is semi remote medium-sized companies

Possible reasons

- lower pay if having to maintain office space
- higher pay due to larger company size

We do not specify our level of remote work but do allow for this position to be offshore - this chart does not effect much except giving possible ranges for in person and remote pay. Medium and small companies have a similar range regardless of remote work ratio -

- Large: 100,289 to 183,436
- Medium: 99,836 to 144,194
- Small: 109,729 to 138,187

Reasonable estimate of a pay range for a small company on it way to a medium company with competative pay based on only this chart would be 120,000 - 150,000

Comparing average pay Globally of Senior and Executive level employees

Observations - Senior Full time Salary Data

Reasonable estimate of a pay range for a Intermediate employee on their way towards an executive level with a team with competative pay and based off of based on avergae SR and EX over past 3 years pay with the anticipation salaries are going up would be **140,000 - 185,000**

Summary Paragraph

I have viewed the data, graphed and compared by multiple variables and focused only on the variables related to our end goal. I can come to the conclusion that based on this data avergae salaries of our target experience level and employment type have most recently gone down between 2021 and 2022. We know the salaries are going up on average due to the recession. Referring to the higher salary levels in 2021 and 2020 would be prudent. The range for senior to executive for 2020 is 141,784 to 202,416. Comparing domestic and internationally based companies and employee residences, when the US is in either of those catagories the pay is higher. On average, non-US countries with non-us employees have lower pay due. The outliers to both of these, on the high end is Japan and Myanmar. The statistics for all full-time Senior and Executive employees show a mean of 143,287 and a median of 137,141. There are lower values of lower-paying countries pulling the median down but it is not drastic. The high average of small companies over 3 years sits in the median-mean range listed above at a value of 138,187, but this is below the range we want to refer to (senior to executive for 2020 is 141,784 to 202,416.) We know that we are an expanding company, so moving above the small company range is prudent, which, combined with wanting top talent that has current and future experience between the levels of a senior and executive, I would reccomend a this US company to offer 150,000 to 200,000. A low of 150,000 to be competitive for senior-level employees and be above the overall mean pay. High of 200,000 to be competitive with executive-level employees. This pay range is competitive for US employees working for international companies and US employees working for US companies, when keeping in mind our company size of small, growing into medium. It is especially competitive for internationally based employees since this position can be remote.