

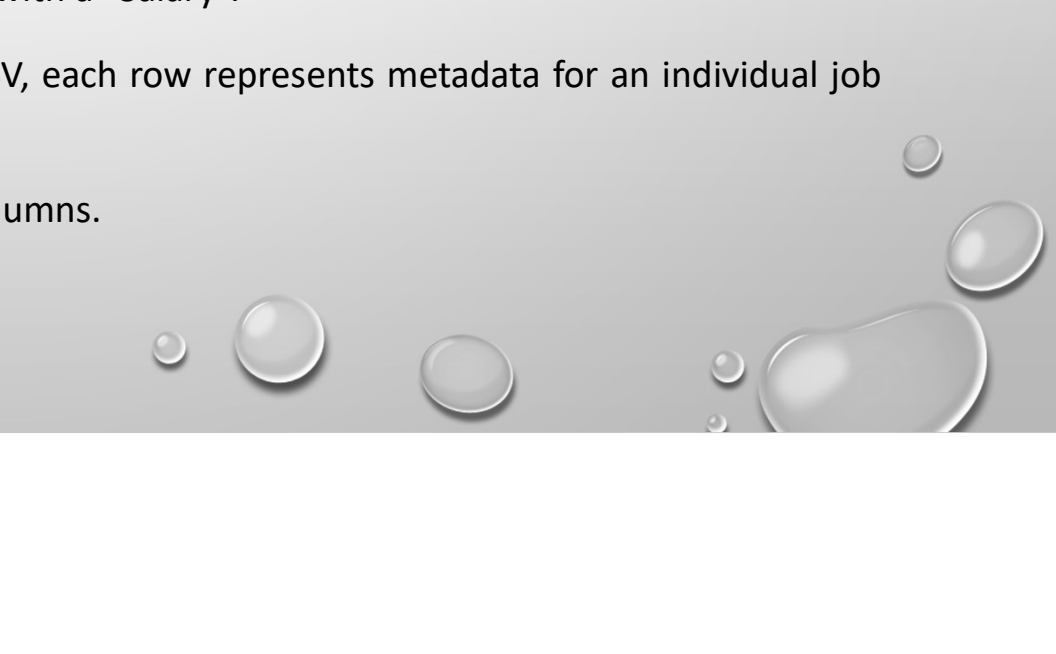
The background of the slide is a light gray gradient. It is decorated with numerous realistic water droplets of various sizes. Some droplets are at the top left, some are in the middle right, and others are at the bottom right. They have highlights and shadows, giving them a three-dimensional appearance.

SALARY PREDICTION BASED ON JOB DESCRIPTIONS

THIS PROJECT IS TO EXAMINE A SET OF JOB POSTINGS WITH SALARIES AND THEN PREDICT
SALARIES FOR A NEW SET OF JOB POSTINGS.




DATA SUPPLIED

- There are three CSV Data files:
 1. TRAIN_FEATURES.CSV: Each row represents metadata for an individual job posting.
 - The “jobId” column represents a unique identifier for the job posting. The remaining columns describe features of the job posting.
 2. TRAIN_SALARIES.CSV: Each row associates a “jobId” with a “Salary”.
 3. TEST_FEATURES.CSV: Similar to TRAIN_FEATURES.CSV, each row represents metadata for an individual job posting.
 - The first row of each file contains headers for the columns.
- 




SOFTWARE LANGUAGES AND LIBRARIES USED

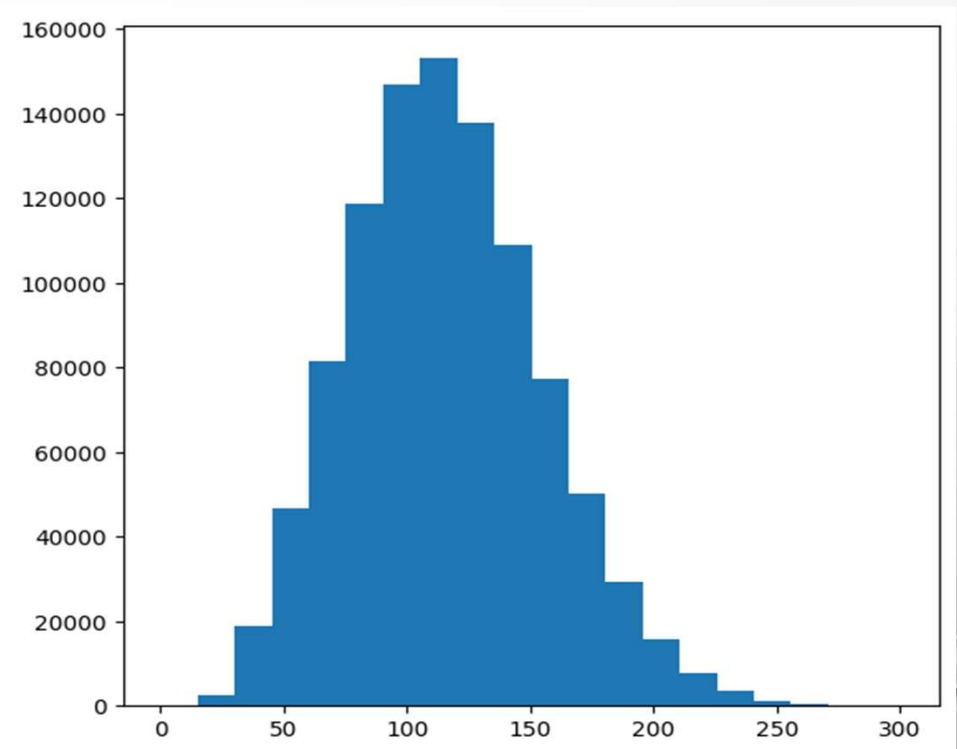
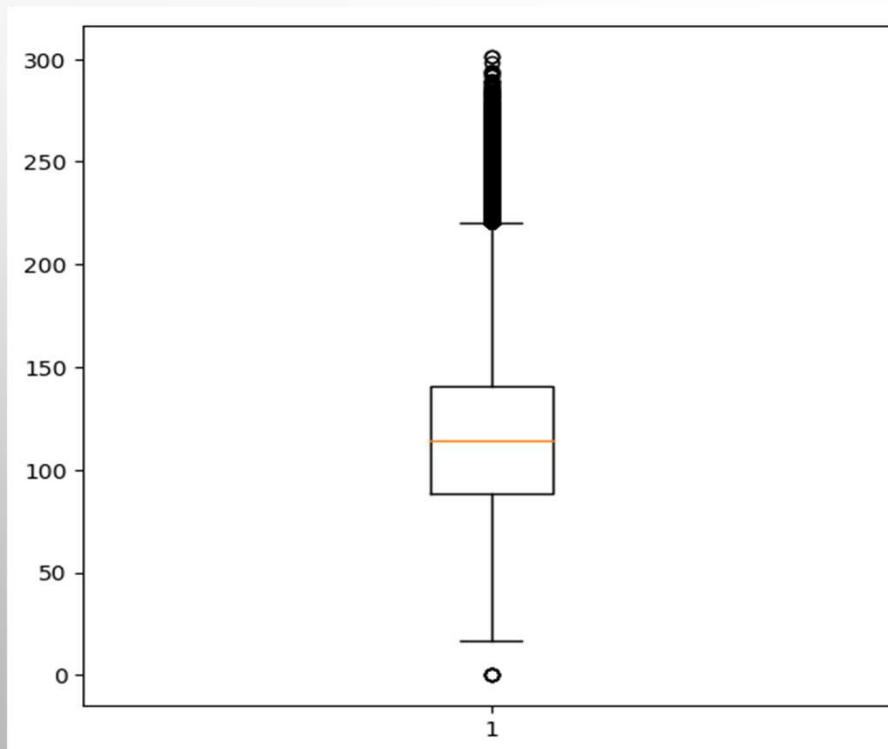
- Python programming language
 - Google Colab for a Notebook Environment
 - Pandas
 - NumPy
 - Scikit-Learn
 - Seaborn
 - Matplotlib
- 



STEPS TAKEN FOR THIS PROJECT

- PART 1: DEFINED THE PROBLEM
 - OUTLINED WHAT ARE THE FEATURES, THE TARGET VARIABLE?
 - IS IT A REGRESSION PROBLEM OR CLASSIFICATION? THEN DECIDED THE METRIC TO OPTIMIZE.
 - PART 2: DISCOVERED THE DATA
 - CHECKED MISSING, DUPLICATE DATA, AND OUTLIERS AND SUMMARIZED THE DATA.
 - VISUALIZED THE FEATURES WITH THE TARGET TO CHECK THEIR IMPACT AND RELATIONSHIP.
 - PART 3: DEVELOPED THE MODEL
 - BUILT LINEAR REGRESSION, GRADIENT BOOSTING, AND RANDOM FOREST REGRESSION MODEL.
 - FINE-TUNED THEM BY HAND, AND FIT THEM, SELECTED THE BEST ONE, FIT AND CHECKED THE PREDICTION.
- 

BOXPLOT DENOTES OUTLIERS WHICH NEED TO BE FURTHER INVESTIGATED



OUTLIERS ARE BELOW 8.5 AND ABOVE 220.5

```
▶ stats = train_df.salary.describe()
print(stats)
IQR = stats["75%"] - stats["25%"]
upper = stats["75%"] + 1.5 * IQR
lower = stats["25%"] - 1.5 * IQR
print(f"The upper and lower bounds for suspected outliers are {upper} and {lower}")
```

```
⇒ count    1000000.000000
   mean         116.061818
   std          38.717936
   min           0.000000
   25%           88.000000
   50%          114.000000
   75%          141.000000
   max          301.000000
Name: salary, dtype: float64
The upper and lower bounds for suspected outliers are 220.5 and 8.5
```

THE SALARY IS 0 WHICH LOOKS CORRUPT DATA SO BEST TO DROP THESE RECORDS

```
# Check potential outliers below lower bounds  
train_df[train_df.salary < 8.5]
```



	jobId	companyId	jobType	degree	major	industry	yearsExperience	milesFromMetropolis	salary
30559	JOB1362684438246	COMP44	JUNIOR	DOCTORAL	MATH	AUTO	11	7	0
495984	JOB1362684903671	COMP34	JUNIOR	NONE	NONE	OIL	1	25	0
652076	JOB1362685059763	COMP25	CTO	HIGH_SCHOOL	NONE	AUTO	6	60	0
816129	JOB1362685223816	COMP42	MANAGER	DOCTORAL	ENGINEERING	FINANCE	18	6	0
828156	JOB1362685235843	COMP40	VICE_PRESIDENT	MASTERS	ENGINEERING	WEB	3	29	0

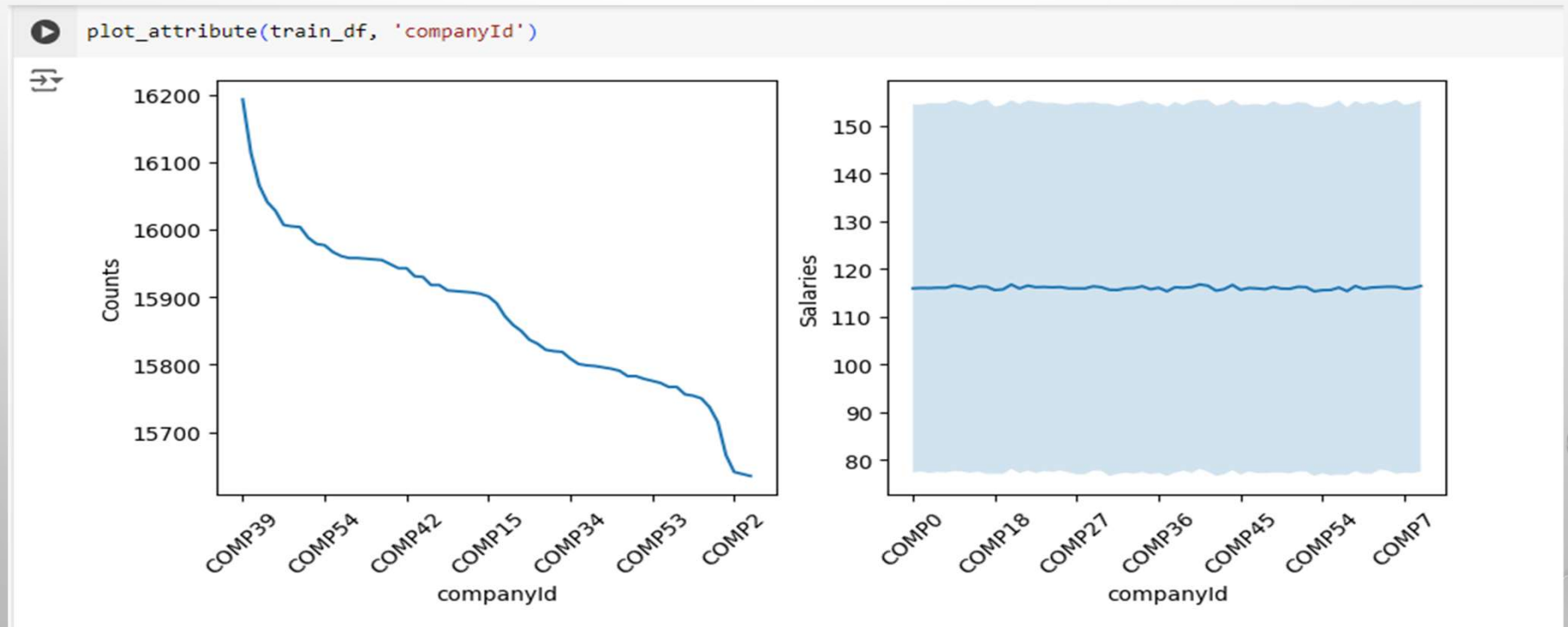
DEGREE, MAJOR, AND EXPERIENCE JUSTIFIES THE SALARY THOUGH THE JOB TYPE IS JUNIOR

```
# Check potential outliers above upper bounds
train_df[(train_df.salary > 222.5) & (train_df.jobType == "JUNIOR")]
```

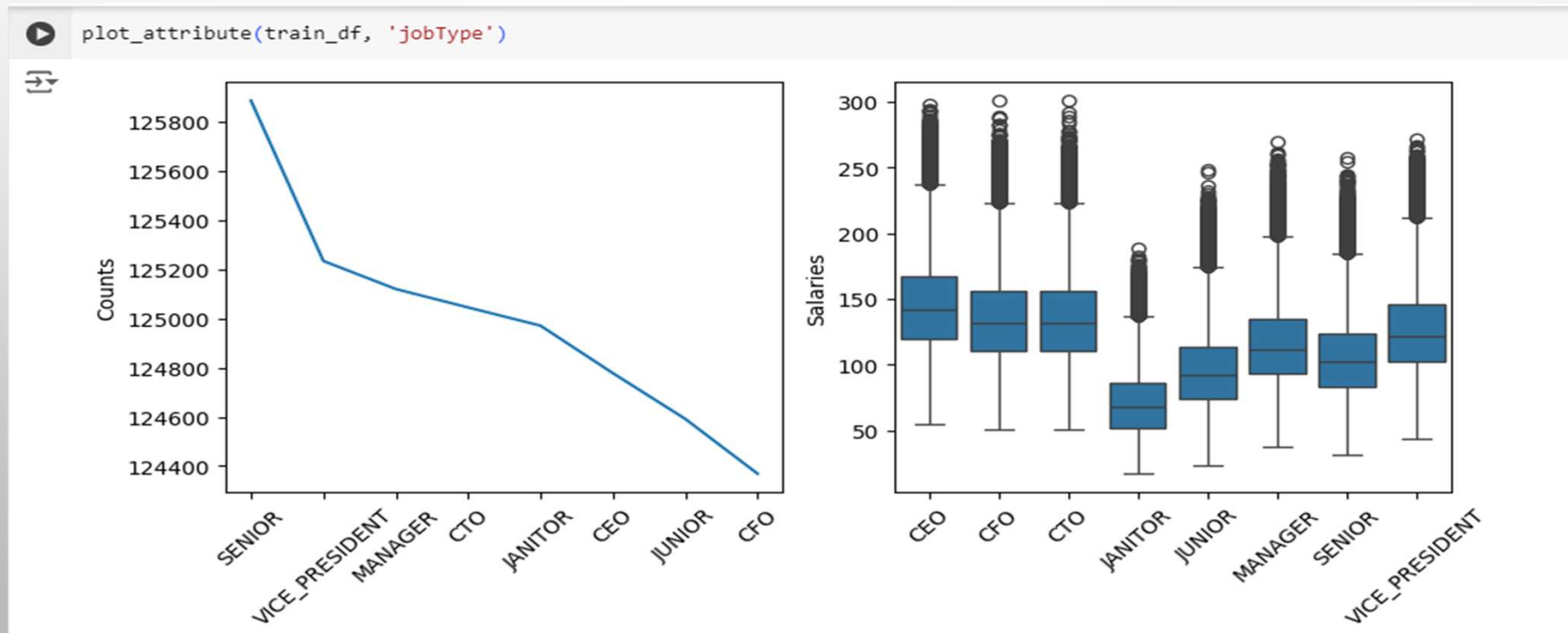


	jobId	companyId	jobType	degree	major	industry	yearsExperience	milesFromMetropolis	salary
1222	JOB1362684408909	COMP40	JUNIOR	MASTERS	COMPSCI	OIL	24	5	225
27710	JOB1362684435397	COMP21	JUNIOR	DOCTORAL	ENGINEERING	OIL	24	3	246
31355	JOB1362684439042	COMP45	JUNIOR	DOCTORAL	COMPSCI	FINANCE	24	0	225
100042	JOB1362684507729	COMP17	JUNIOR	DOCTORAL	BUSINESS	FINANCE	23	8	248
160333	JOB1362684568020	COMP18	JUNIOR	DOCTORAL	BUSINESS	FINANCE	22	3	223
303778	JOB1362684711465	COMP51	JUNIOR	MASTERS	ENGINEERING	WEB	24	2	226
348354	JOB1362684756041	COMP56	JUNIOR	DOCTORAL	ENGINEERING	OIL	23	25	226
500739	JOB1362684908426	COMP40	JUNIOR	DOCTORAL	ENGINEERING	OIL	21	0	227
627534	JOB1362685035221	COMP5	JUNIOR	DOCTORAL	ENGINEERING	OIL	24	29	230
645555	JOB1362685053242	COMP36	JUNIOR	DOCTORAL	BUSINESS	FINANCE	24	1	225
685775	JOB1362685093462	COMP38	JUNIOR	BACHELORS	ENGINEERING	OIL	24	13	225
743326	JOB1362685151013	COMP14	JUNIOR	DOCTORAL	BUSINESS	FINANCE	19	0	236
787674	JOB1362685195361	COMP43	JUNIOR	DOCTORAL	BUSINESS	FINANCE	18	15	232
796956	JOB1362685204643	COMP30	JUNIOR	MASTERS	BUSINESS	OIL	24	2	228
855219	JOB1362685262906	COMP13	JUNIOR	MASTERS	ENGINEERING	OIL	22	26	225
954368	JOB1362685362055	COMP11	JUNIOR	DOCTORAL	BUSINESS	OIL	24	26	223

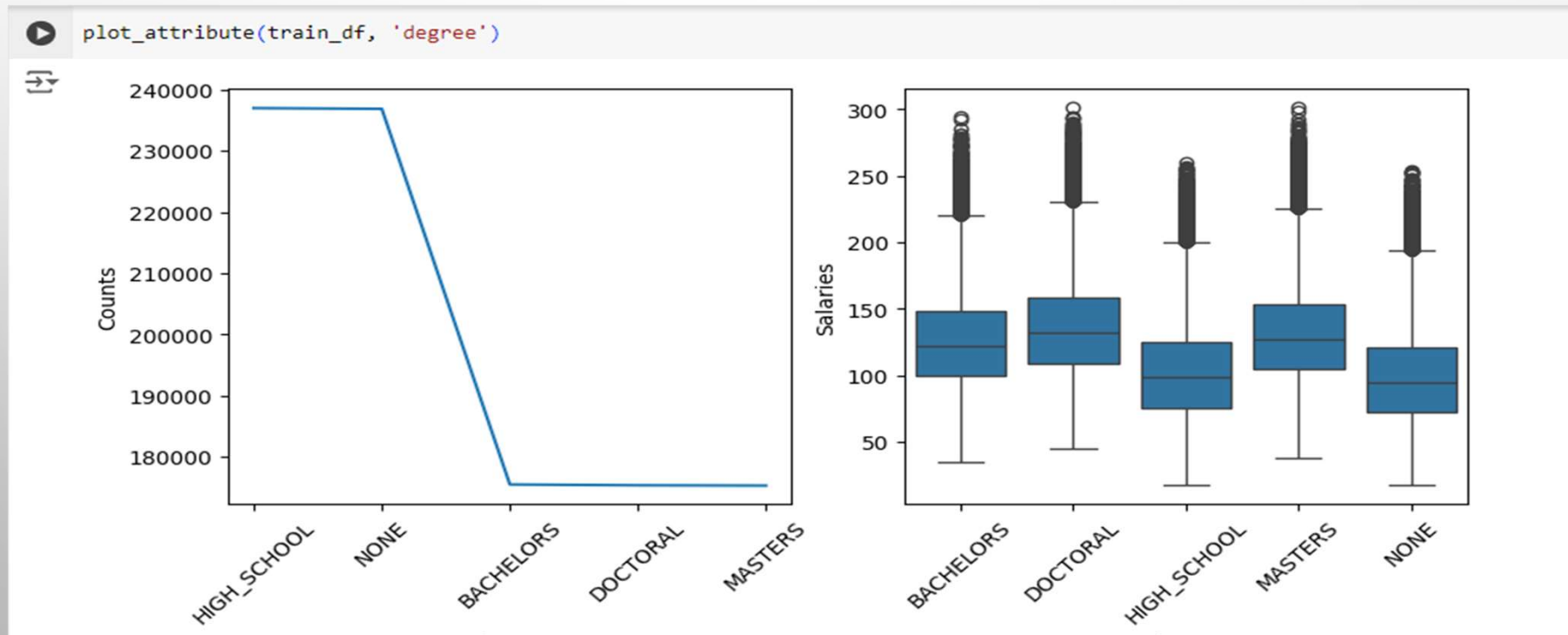
COMPANYID DOES NOT GIVE ANY INSIGHT



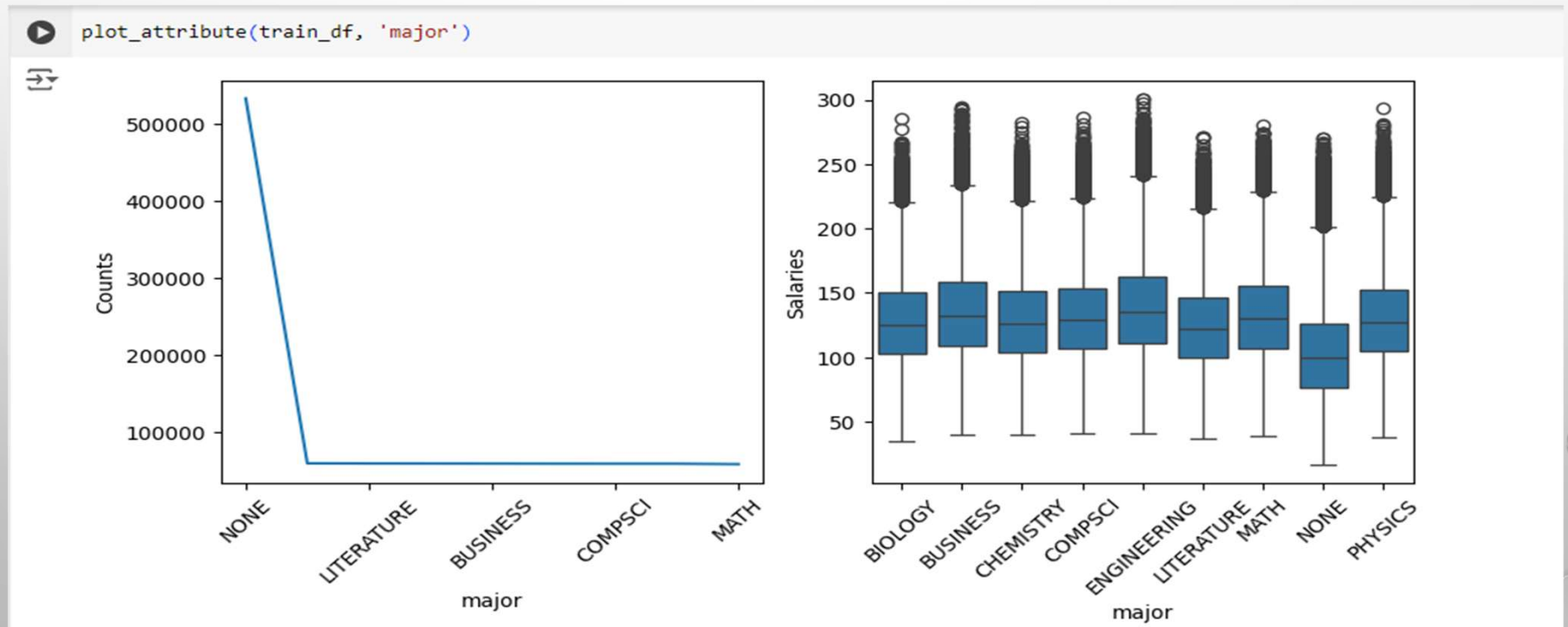
AS THE LEVEL OF SENIORITY GOES UP, THE SALARY INCREASES.



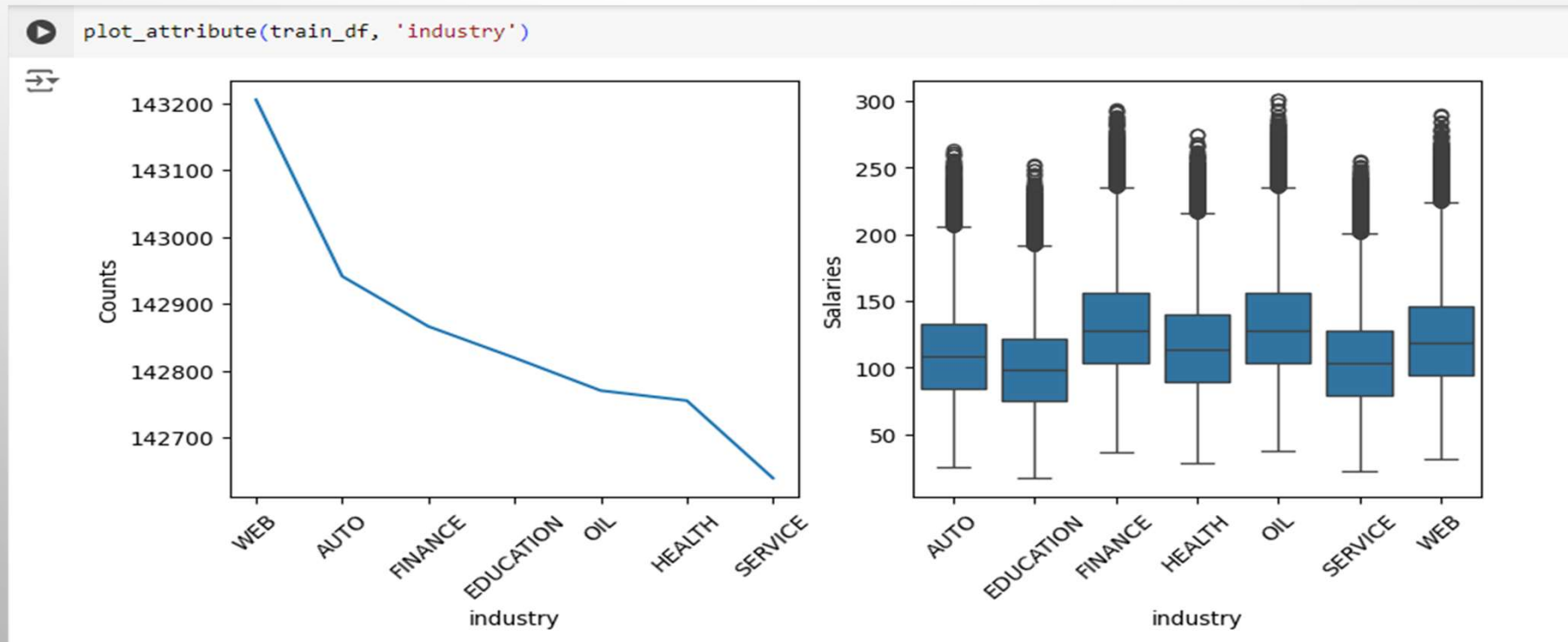
PEOPLE WITHOUT DEGREE OR WITH A HIGH SCHOOL
OBTAIN THE LOWER SALARIES IN COMPARISON WITH
PEOPLE WHO HAVE BACHELORS, MASTERS OR DOCTORAL.



EMPLOYEES WHO ARE MAJOR IN ANY SUBJECT GET NEARLY SAME SALARY. HOWEVER, THOSE WITHOUT ANY MAJOR OBTAIN 20% LOWER WAGES THAN THEIR PEERS.

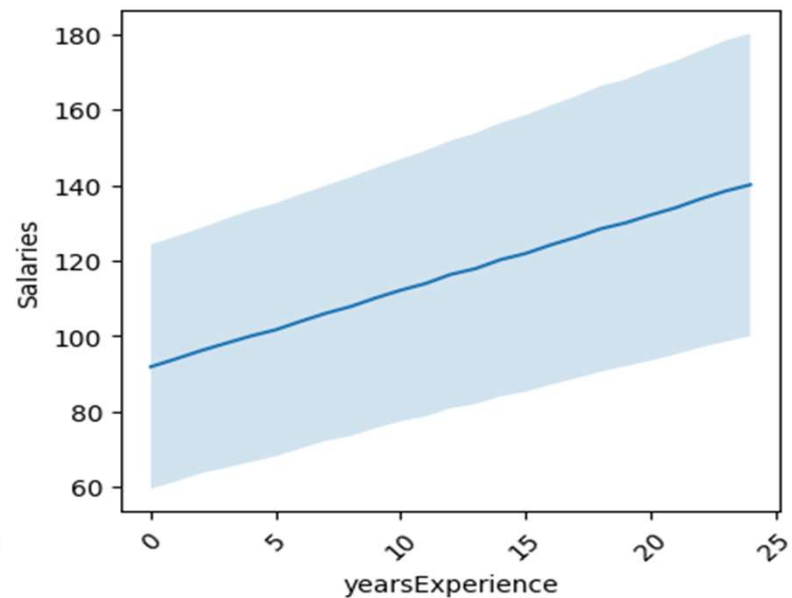
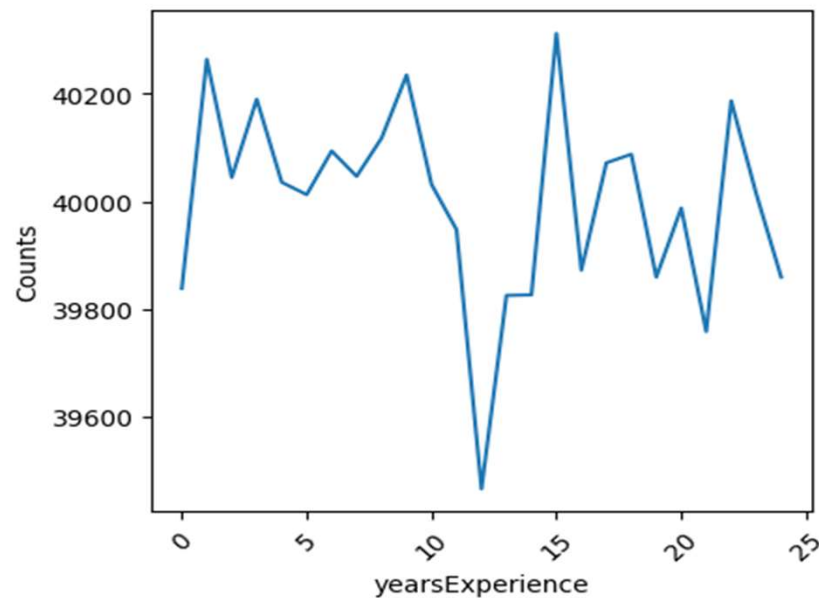


THE HIGHEST WAGES ARE PAID IN OIL AND FINANCE, WHEREAS EDUCATION SECTOR PAYS THE LOWEST.



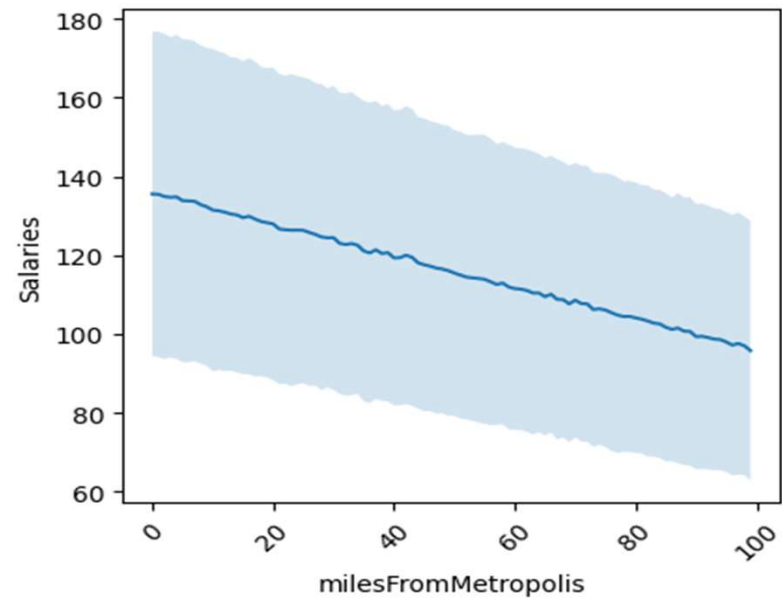
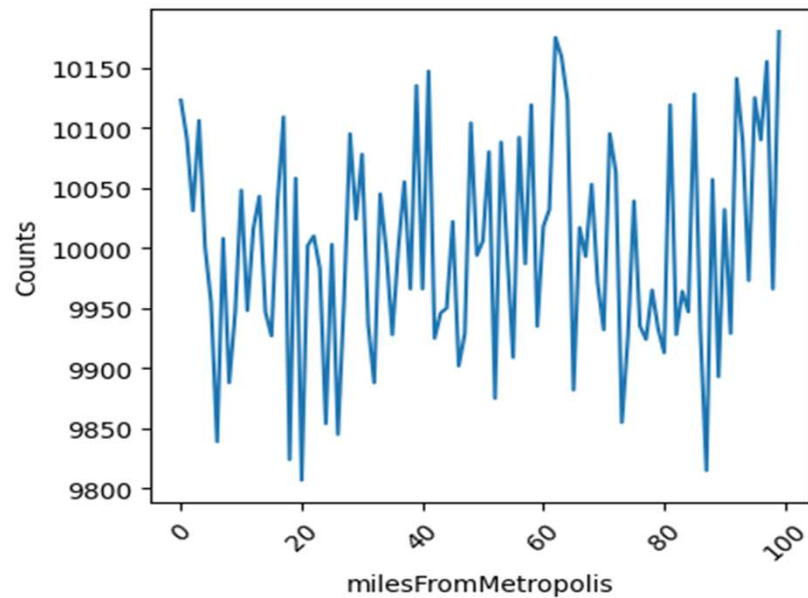
AS EXPERIENCE INCREASES, THE AMOUNT OF SALARY GOES UP.

```
plot_attribute(train_df, 'yearsExperience')
```



AS YOU LIVE FAR FROM METRO CITIES, YOUR SALARY GOES DOWN.

```
plot_attribute(train_df, 'milesFromMetropolis')
```



MODEL SUMMARY

▶ `models.print_summary()`



Model Summaries:

`LinearRegression()` - MSE: 358.1689429533021

`RandomForestRegressor(max_depth=15, max_features=8, min_samples_split=80,
n_estimators=60, n_jobs=-1)` - MSE: 313.31678084441927

`GradientBoostingRegressor(max_depth=7, n_estimators=40)` - MSE: 313.0541452486368

Best Model:

`GradientBoostingRegressor(max_depth=7, n_estimators=40)`

MSE of Best Model
313.0541452486368

FEATURE IMPORTANCES

Feature Importances

feature	importance
group_mean	0.690169
yearsExperience	0.152483
milesFromMetropolis	0.104514
group_min	0.015269
group_std	0.013361
group_max	0.011826
group_median	0.010032
jobType	0.001527
industry	0.000591
major	0.000193
degree	0.000028
companyId	0.000007

FEATURE IMPORTANCES PLOT

