Jobs and Salaries in Data Science: Trends, findings and predictions

Introduction:

In this article, we will share our journey of using Python to analyze “Data Science jobs” dataset, and discover valuable information. We will explain our approach, present our discoveries and some interesting ideas.

Our work is based on an Excel file called “jobs_in_data.csv” which is located on the “Kaggle” platform that we will analyze with the help of “jupyter notebook”.
Our chapters are:

1. Data cleaning.
2. Exploratory data analysis.
3. Learning Model.

Chapter I: Data Cleaning:

	work_year	job_title	job_category	salary_currency	salary	salary_in_usd	employee_residence	experience_level	employm
0	2023	Data DevOps Engineer	Data Engineering	EUR	88000	95012	Germany	Mid-level	Full-time
1	2023	Data Architect	Data Architecture and Modeling	USD	186000	186000	United States	Senior	Full-time
2	2023	Data Architect	Data Architecture and Modeling	USD	81800	81800	United States	Senior	Full-time
3	2023	Data Scientist	Data Science and Research	USD	212000	212000	United States	Senior	Full-time
4	2023	Data Scientist	Data Science and Research	USD	93300	93300	United States	Senior	Full-time

Figure 1 : Data before modifications

Our Excel file is made up of 12 columns and 9355 rows, here are more details:
Work year, job title, experience level, work setting, company location, and company size.

We have removed several duplications but it caused a lot of reduction of our data (from 9355 to 5341 rows), gratefully there are no missing values and the instruction df.describe () had showed many interesting results:

```
df.describe()
```

	work_year	salary	salary_in_usd
count	5341.000000	5341.000000	5341.000000
mean	2022.682082	145814.937839	146258.409099
std	0.608026	67025.469452	66594.117529
min	2020.000000	14000.000000	15000.000000
25%	2022.000000	97300.000000	98506.000000
50%	2023.000000	140000.000000	140000.000000
75%	2023.000000	186200.000000	186000.000000
max	2023.000000	450000.000000	450000.000000

Figure 2 : Description of the data

Chapter II: Exploratory Data Analysis:

1- Univariate analysis:

In our exploratory data analysis, we have found very interesting results like the visualizations below:

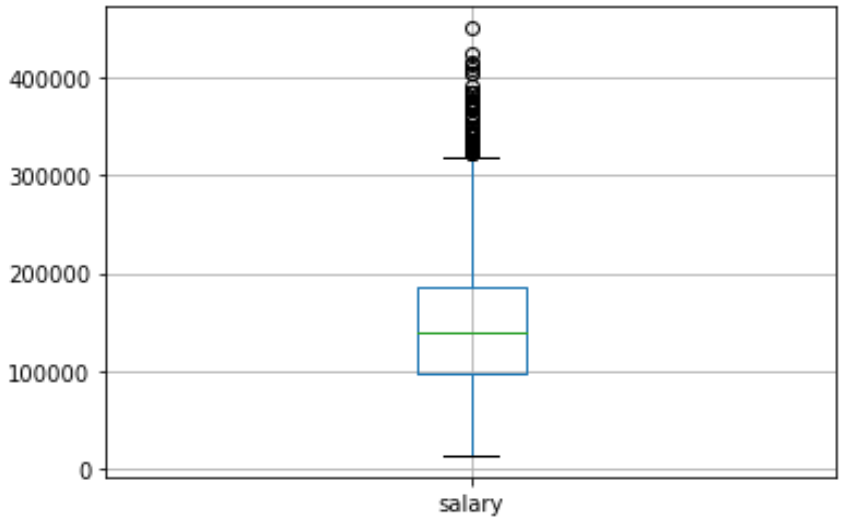


Figure 3 : Salary Variation in a boxplot

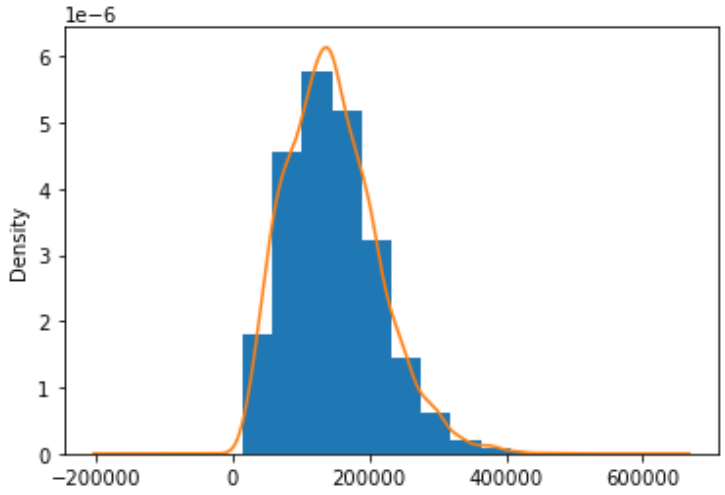


Figure 4 : variation of salary depending on density

2- Bivariate analysis:

The bivariate analysis is very important to find patterns and the variables the needs to be modified to apply our machine learning procedure, here's some visualizations:

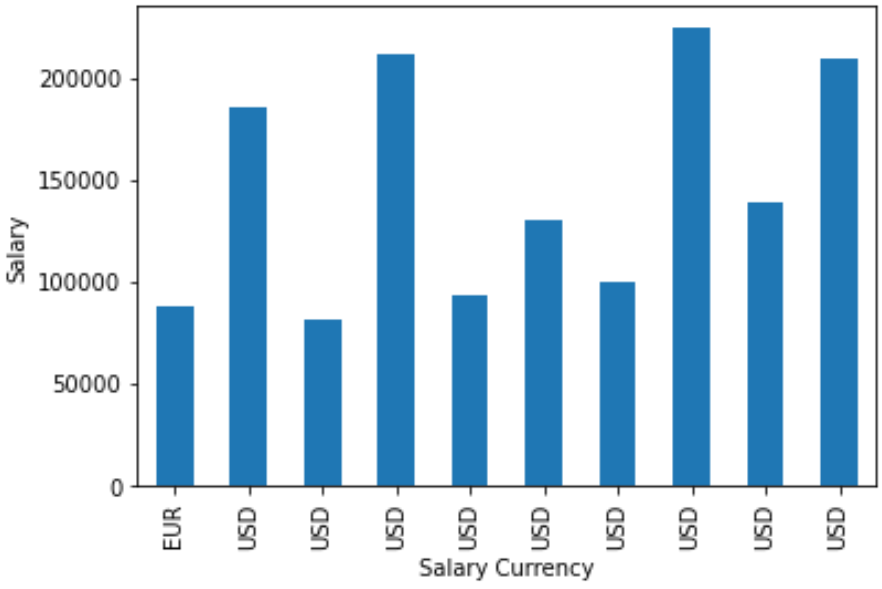


Figure 5 : Salary currency of 10 different employees

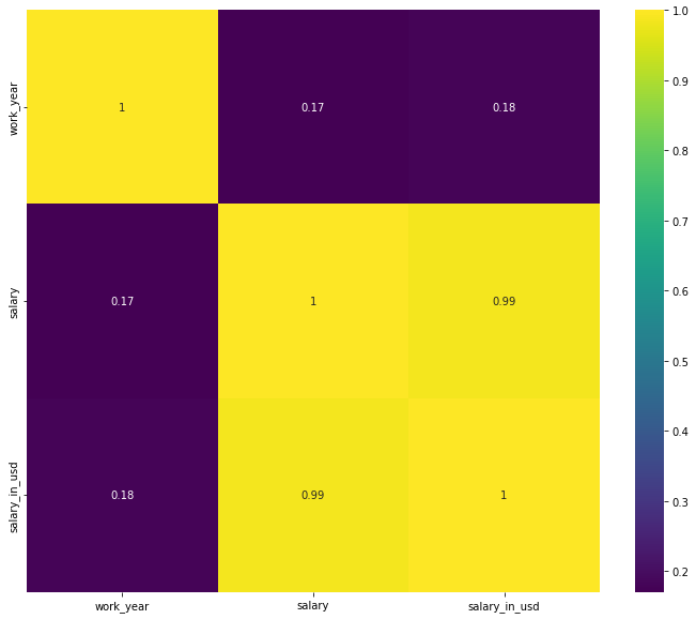
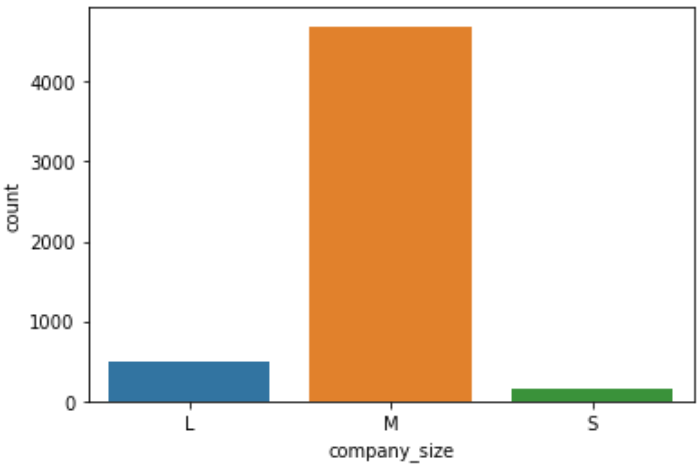


Figure 6 : Heatmap

Our dataset have various columns and we have chosen company size as our target to predict in the machine learning process, indeed we distinguish 3 types of companies and as they are object variables we did some modification:



We will assign large companies with 3, medium with 2 and small with one.

Chapter III: Machine Learning Model:

We have chosen salary, salary in usd, company size and work year as features, and the target variable will be company size.

We will try 4 models in our work then we will compare the accuracy.

	precision	recall	f1-score	support
1	0.57	0.23	0.33	99
2	0.91	0.98	0.94	943
3	0.50	0.11	0.18	27
accuracy			0.89	1069
macro avg	0.66	0.44	0.49	1069
weighted avg	0.87	0.89	0.87	1069

Figure 7: KNN

	precision	recall	f1-score	support
1	0.00	0.00	0.00	99
2	0.88	1.00	0.94	943
3	0.00	0.00	0.00	27
accuracy			0.88	1069
macro avg	0.29	0.33	0.31	1069
weighted avg	0.78	0.88	0.83	1069

Figure 8: Naive Bayes

	precision	recall	f1-score	support
1	0.60	0.03	0.06	99
2	0.88	1.00	0.94	943
3	0.00	0.00	0.00	27
accuracy			0.88	1069
macro avg	0.49	0.34	0.33	1069
weighted avg	0.84	0.88	0.83	1069

Figure 9: Logistic Regression

	precision	recall	f1-score	support
1	0.60	0.03	0.06	99
2	0.88	1.00	0.94	943
3	0.00	0.00	0.00	27
accuracy			0.88	1069
macro avg	0.49	0.34	0.33	1069
weighted avg	0.84	0.88	0.83	1069

Figure 10: SVM

The model that have the best accuracy is KNN, which have 0.89, all the other models have 0.88.