Is Data Science still the sexist job of the 21st Century?

Michael Zelaya

Mzelaya@bellarmine.edu

1/15/2024

Executive Summary:

Nowadays, several roles/categories are involved in the data-related job market. For example, a few include 'Data Engineer,' 'Data Analyst,' 'Data Architect,' etc. Since there are many job titles in the data field, I want to predict which job roles/categories will most likely be in need and the most reliable in the future. The owner of the dataset, Hummaam Qaasim, calls the dataset that I will be using *Jobs and Salaries in Data Science,* which I found on Kaggle. In total, the dataset has 12 columns and 9,356 rows. As for modeling, I plan on using three potential predictive models: the nearest neighbor, random forest, and gradient boosting algorithms. In addition, as a baseline for evaluating predictive models, I am interested in using linear/logistic regression to compare the accuracy scores with the other three predictive models I will use with my data. As for tools, I plan on using the Python programming language. In addition, as for libraries in Python, I plan to use numpy, pandas, matplotlib, seaborn, and sklearn. Lastly, as for visualization tools, I plan on using Python and Tableau. I firmly believe these steps can push me in the right direction to clean the data, perform explanatory data analysis (EDA), and ultimately produce the most accurate predictive model. All in all, technology and the data industry are constantly evolving rapidly. Should data science as a career be a cause for worry?

Project idea:

My project idea is to find a trend and accurately predict which data-related job role/category is on the rise and the most secure for the near future. Many unique data-related careers exist today, such as data analyst, data engineer, data architect, etc. Meanwhile, a couple of years ago, data science was the only career that existed and was in need, and it continues to do so. In addition, I'm also curious if artificial intelligence will affect data-related job markets in the upcoming years.

Background:

Since there are many different careers related to data, I want to predict which job categories will be in need and most stable in the future. The dataset I will be using is called jobs_in_data.csv, and I found it on Kaggle. It has

12 columns and 9,356 rows in total. A bit about the dataset: the **work_year** column is the year it was recorded and is essential for understanding salary trends over time. Next, **job_title** is the specific title of the job role. It is also crucial for understanding salary distribution across various roles within the data field. Then, **job_category** is a broader category of the job role for more straightforward analysis. For example, it can include areas like "Data Analysis", "Data Engineering," etc. Following is the **salary_currency,** which is the currency in which the salary is payable—for instance, USD, EUR, etc. It can be essential to understanding the actual value of wages in a global context. Then, **salary** is the annual gross salary of the role in the local currency. In addition, **salary_in_usd** is the gross yearly salary converted to U.S. dollars (USD). It aids in global salary comparisons and analyses. Next is **employee_residence,** which is the employee's country of residence. It can be helpful to explore geographical salary differences and cost of living. After that, **experience_level** classifies the professional experience level of the employee. For instance, it can include "Entry-level," "Mid-level," "Senior," and "Executive." It can provide insight into how experience influences salary in data-related roles. Next, **employment_type** specifies the type of employment, like "Full-time," "Part-time," "Contract," etc. It can help in analyzing how different employment can affect salary. Then, **work_setting** is the work setting or environment, such as "Remote," "In-person," or "Hybrid." It can reflect the impact of work settings on salary levels in the data industry. The following is the **company_location,** which is the country the company is in. It can help in analyzing how the location of the company affects salary. Lastly is the **company_size,** the size of the employer's company. For example, it can split up into small (S), medium (M), and large (L) sizes. This column allows for an analysis of how a company's size can influence salary.

Modeling:

At first, as a baseline for evaluating predictive models, I am interested in using linear/logistic regression with my data to compare the accuracy scores with the other three predictive models I will also be using. The other three potential predictive models I'm interested in implementing with my dataset are the nearest neighbor, random forest, and gradient boosting algorithms. I first want to talk about the nearest neighbor algorithm, one of the first algorithms almost to solve the traveling salesman problem, which is quite impressive. According to Yse, in simple terms, the nearest neighbor algorithm predicts responses for the testing data based on its similarity with the training data. In addition, the algorithm assumes that data with similar characters are linked and uses distance measures at its

core. Next, I want to discuss and further explain the random forest algorithm for my data. According to R from Analytics Vidhya, random forest uses a collection of decision trees to make predictions. In addition, each decision tree trains a different portion of the data, and the forecasts of all trees are an average to produce the final prediction. Lastly, the final predictive model I want to focus on for my project is the gradient boosting algorithm. In simple terms, gradient boosting is an algorithm that "…repeatedly selects a function that leads in the direction of a weak hypothesis or negative gradient so that it can minimize a loss function. Gradient boosting classifier combines several weak learning models to produce a powerful predicting model" (Simplilearn, 2023). In other words, the gradient boosting algorithm minimizes the overall prediction error by relying on the best possible next model combined with previous models.

Tools:

To complete this project, I plan on using just the Python programming language in Jupyter Notebook from Anaconda-Navigator. I chose Python instead of other programming languages like R or SQL because I feel more experienced and comfortable using it than the other two languages. As for libraries in Python, I plan on using quite a few, such as numpy, pandas, matplotlib, seaborn, and sklearn. I chose these libraries because these packages would help me clean the data, perform explanatory data analysis (EDA), and ultimately perform predictive analysis using machine learning. Lastly, as for visualization tools, I plan on using only Tableau. I chose Tableau instead of other visualization tools like Power BI or Google Charts because I have more experience and feel more comfortable and confident with Tableau than other visualization tools in the market.

Conclusion:

Lastly, many new job roles exist in today's data field industry, and salary ranges depend on experience level, work setting, company size, and country in which an employer lives. I want to figure out which job roles/categories in the future would most likely still exist and be most stable in the long run. I plan to clean the data, perform explanatory data analysis (EDA), and implement predictive models to determine an outcome. Would data science still be on the rise?

References

Davenport, T. H., & Patil, D. (2022, July 21). *Is data scientist still the sexiest job of the 21st Century?*. Harvard Business Review. https://hbr.org/2022/07/is-data-scientist-still-the-sexiest-job-of-the-21st-century

Qaasim, H. (2023, December 25). *Jobs and salaries in Data Science*. Kaggle. https://www.kaggle.com/datasets/hummaamqaasim/jobs-in-data/data

R, S. E. (2024, January 3). *Understand random forest algorithms with examples (updated 2024)*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/#:~:text=Random%20forest%20is%20an%20ensemble,to%20produce%20the%20final%20prediction

Yse, D. L. (n.d.). *K-Nearest Neighbor (KNN) explained*. Pinecone. https://www.pinecone.io/learn/k-nearest-neighbor/