Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

MIT-WPU
।। विश्वशान्तिर्ध्रुवं ध्रुवा ।।

Mini Project Report

On

AI impact on jobs by 2030

Submitted by

PRN	Name of Student
1262253274	Tanishka Tiwari
1262252045	Pournami Roy
1262252891	Palak Soni
1262252177	Prachiti Suryawanshi
12622520376	Vrukshita Vinod

Under the Guidance of :

Prof. Vidya Patil

Abstract:

This Python mini project investigates the evolving dynamics of employment in the face of artificial intelligence and automation by the year 2030. Using a structured dataset encompassing job roles, skill requirements, and educational levels, the project models the probability of automation across professions. It employs machine learning techniques to predict automation risk, integrates explainable AI to interpret model behavior, and visualizes trends to uncover socioeconomic patterns.

Methodologies include:

- **Data preprocessing**: Cleaning, encoding, and normalizing features for model readiness.
- **Predictive modeling**: Logistic regression, decision trees, and ensemble methods to estimate automation probabilities.
- **Data visualization**: Interactive charts and dashboards to explore automation trends and job vulnerabilities.

This project serves as a foundation for understanding which professions are most susceptible to automation, why they are at risk, and how society might adapt through education, policy, and reskilling initiatives.

List of Abbreviations:

Abbreviation	Full Form
CSV	Comma Separated Values
DF	DataFrame
NaN / NA	Not a Number (Missing Value)
IDE	Integrated Development Environment

Figure No.	Figure Name
1	NEEDED MODULES
2	BASIC INSPECTION
3	DATA CLEANING
4	STATISTICS
5	FILTERING AND SORTING
6	GROUPING AND AGGREGATION
7	VISUALISATION

s

Table Of Contents :

Serial No.	Section title	Code/function
1	Needed Modules	import numpy as np import pandas as pd import matplotlib.pyplot as p import seaborn as s
2	Basic Inspection	pd.read_csv() df df.info() df.shape df.duplicated() df.head() df.tail() df.nunique() df['Years_Experience'].nunique() df['Years_Experience'].dtype

3	Data Cleaning	• df.isnull().sum() • df.dropna() • df.fillna(method='ffill') • df.drop_duplicates() • sns.heatmap(df.isnull(), ...)
4	Statistics	• df.describe() • df • df['Tech_Growth_Factor'].mean() • df['Average_Salary'].min() • df['Average_Salary'].max() • df[df['Average_Salary']==df['Average_Salary'].max()] • df[df['Average_Salary']==df['Average_Salary'].min()]
5	Filtering and sorting	• df['Total_skills_required'] = df['Skill_1'] + df['Skill_2'] + ... + df['Skill_10'] • df • def increaseBY2(x): return x + 2 • df['Average_Salary'] = df['Average_Salary'].apply(increaseBY2) • df.head() • df.sort_values(by='Average_Salary') • df.sort_values(by='Average_Salary', ascending=False)

6	Grouping and aggregation	df.groupby('Risk_Category')['Average_Salary'].agg(['mean', 'sum', 'count','min','max']
7	Visualisation	• fig, axes = p.subplots(3, 1, figsize=(8, 8)) • axes[0].hist(df['Risk_Category']) • axes[1].hist(df['Education_Level']) • axes[2].hist(df['Total_skills_required']) • s.countplot(x="Risk_Category", hue="Education_Level", data=df) • pd.crosstab(df["Education_Level"], df["Risk_Category"]) • s.heatmap(relation_table, annot=True, cmap="Blues") • df.groupby('Job_Title')['Years_Experience'].sum() with ax.plot(...) • p.pie(df['Years_Experience'].head(), labels=df['Job_Title'].head()) • p.pie(df['AI_Exposure_Index'].head(), labels=df['Job_Title'].head()) • s.scatterplot(data=df, x='Risk_Category', y='AI_Exposure_Index', hue='Job_Title') • s.scatterplot(data=df, x='Years_Experience', y='Automation_Probability_2030') • s.regplot(data=df, x='Years_Experience', y='Automation_Probability_2030', scatter=False) • s.scatterplot(data=df, x='Risk_Category', y='Total_skills_required', hue='Job_Title') • df.drop(...) to remove columns → df4.corr() • s.heatmap(corr, annot=True, fmt=".2f", cmap="coolwarm")

Introduction:

The rapid growth of Artificial Intelligence (AI) and automation technologies is reshaping industries across the world. As AI systems become more capable, many professions are expected to undergo major transformations by the year 2030. The Kaggle dataset **"AI Impact on Jobs 2030"** provides a structured and data-driven look into how different jobs, skills, and education levels may be affected by AI-driven automation in the near future.

This dataset offers insights into **automation risk scores**, **future job demand**, **required skill shifts**, and the **overall vulnerability of various occupations**. By analyzing this information, researchers, students, and policymakers can better understand which job sectors are likely to shrink, which roles may evolve with new skill requirements, and which careers may see increased demand due to AI.

The dataset is especially valuable for data science projects, as it allows students to perform **exploratory data analysis (EDA)**, build **predictive models**, create **visualizations**, and study real-world implications of AI on the global workforce. Overall, it serves as a meaningful resource to explore the future of work and the socioeconomic impact of AI by 2030.

Problem Statement:

This project focuses on analysing the impact of Artificial Intelligence (AI) on various jobs by the year 2030. The aim is to understand how AI-driven automation may influence the job market and workforce requirements in the near future.

Objectives

*To analyse how different professions, skills, and education levels might be impacted by AI-driven automation by the year 2030.

*To understand which types of jobs are at the highest risk of AI-based automation.

Python Concepts Used and Explanation:

In this project, several Python concepts and libraries were used to analyse the *AI Impact on Jobs 2030* dataset:

a) Libraries Used

Pandas – for loading, cleaning, and analysing the dataset.

NumPy – for numerical operations and array handling.

Matplotlib,Seaborn – for data visualization and plotting graphs.

b) Functions Used

`head()` – to preview the first few rows of the dataset.

`info()` – to check data types and missing values.

`describe()` – to view statistical summaries.

`dropna()` – to remove missing values if required.

`drop_duplicates()` – to remove duplicate rows.

`value_counts()` – to analyse categorical data.

c) Operations Performed

Data cleaning (handling missing and duplicate values)

Data summarization

Filtering and sorting values

Plotting graphs like bar charts, scatter plots, and correlation heatmaps

Analyzing automation risk across job roles

Screenshots of output

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000 entries, 0 to 2999
Data columns (total 18 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   Job_Title                   3000 non-null   object
 1   Average_Salary              3000 non-null   int64
 2   Years_Experience            3000 non-null   int64
 3   Education_Level             3000 non-null   object
 4   AI_Exposure_Index           3000 non-null   float64
 5   Tech_Growth_Factor          3000 non-null   float64
 6   Automation_Probability_2030 3000 non-null   float64
 7   Risk_Category               3000 non-null   object
 8   Skill_1                     3000 non-null   float64
 9   Skill_2                     3000 non-null   float64
 10  Skill_3                     3000 non-null   float64
 11  Skill_4                     3000 non-null   float64
 12  Skill_5                     3000 non-null   float64
 13  Skill_6                     3000 non-null   float64
 14  Skill_7                     3000 non-null   float64
 15  Skill_8                     3000 non-null   float64
 16  Skill_9                     3000 non-null   float64
 17  Skill_10                    3000 non-null   float64
dtypes: float64(13), int64(2), object(3)
memory usage: 422.0+ KB
Dimension of the dataset :  (3000, 18)
presence on duplicate values in rows [True - if found, False - if not found]
1st 5 data from the dataset :
```

```
Number of unique values in Years_Experience column :  30
Type of data in Years_Experience :  int64
Minimum average salary : Rs. 30030
Maximum average salary : Rs. 149798
Increased salaries :
```

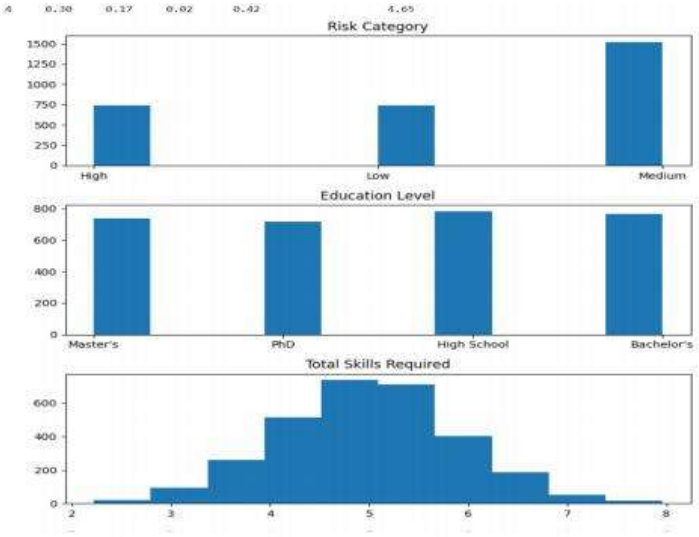
	Job_Title	Average_Salary	Years_Experience	Education_Level
0	Security Guard	45797	28	Master's
1	Research Scientist	133357	20	PhD
2	Construction Worker	146218	2	High School
3	Software Engineer	136532	13	PhD
4	Financial Analyst	70399	22	High School

	AI_Exposure_Index	Tech_Growth_Factor	Automation_Probability_2030
0	0.18	1.28	0.85
1	0.62	1.11	0.05
2	0.86	1.18	0.81
3	0.39	0.68	0.60
4	0.52	1.46	0.64

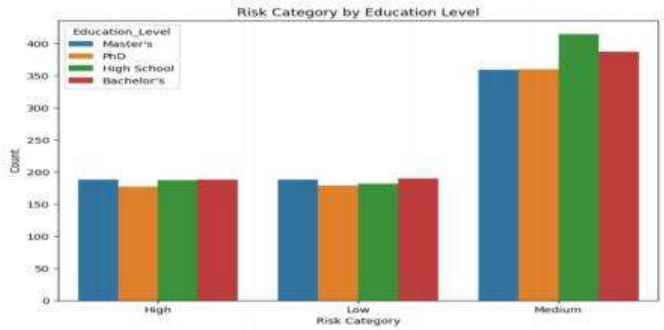
	Risk_Category	Skill_1	Skill_2	Skill_3	Skill_4	Skill_5	Skill_6
0	High	0.45	0.10	0.46	0.33	0.14	0.65
1	Low	0.02	0.52	0.40	0.05	0.97	0.23
2	High	0.01	0.94	0.56	0.39	0.02	0.23
3	Medium	0.43	0.21	0.57	0.03	0.84	0.45
4	Medium	0.75	0.54	0.59	0.97	0.61	0.28

	Skill_7	Skill_8	Skill_9	Skill_10	Total_skills_required
0	0.06	0.72	0.94	0.00	3.85
1	0.09	0.62	0.38	0.98	4.26
2	0.24	0.68	0.61	0.83	4.51

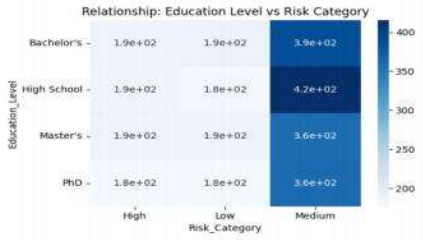
4 0.38 0.17 0.02 0.42 4.65



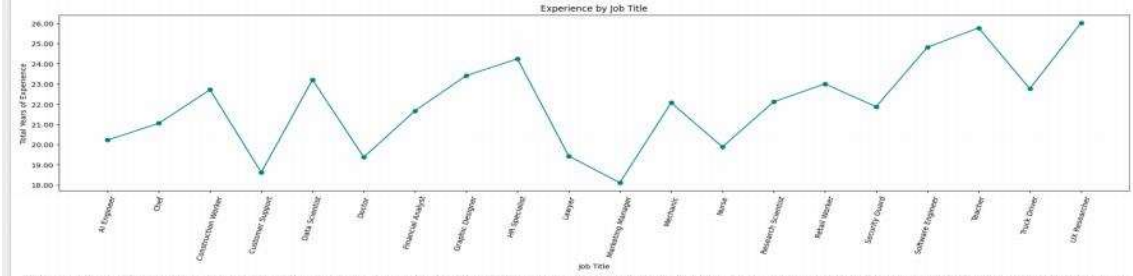
The histogram of Risk Category shows that most individuals fall into the Medium-risk group for AI-driven job automation by 2030. The High-risk group is the smallest, suggesting limited roles face full automation. The Education Level distribution shows that High School and Bachelor's degrees are the most common, while PhD holders are few. The Total Skills Required histogram reveals that most people possess around 4.5–6 skills, indicating moderate skill diversity. A higher number of skills generally reduces the likelihood of AI replacement because multi-skill roles involve more complex, non-routine tasks.



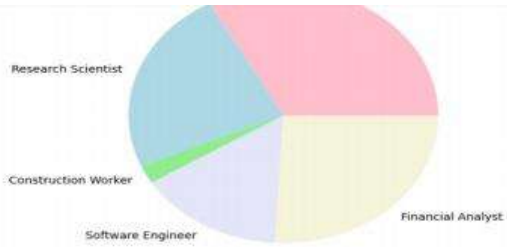
This bar chart shows how many jobs at each education level fall into different AI automation risk categories by 2030. For all four groups (High School, Bachelor's, Master's, PhD), most jobs are in the Medium-risk category, which means they are likely to be reshaped but not fully automated. High- and low-risk jobs are fewer and distributed across education levels, suggesting that no qualification band is completely safe or uniquely exposed to extreme automation risk.



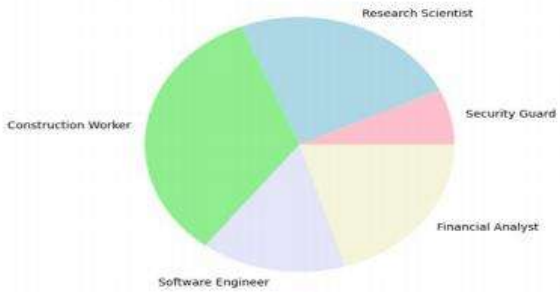
The medium-risk category dominates across all education levels, indicating that most jobs face moderate exposure to AI automation regardless of qualification. High School graduates appear slightly more represented in medium-risk roles. High and low risk categories remain relatively uniform across all groups, implying that education level alone does not strongly determine automation risk—other factors such as job type and skill specialization matter more.



The total experience for the placeholder job titles falls in the range of approximately 20 to 30. There is clear variation in the total experience across the sample jobs (Job A through Job E). For instance, Job D shows the highest total experience, while Job C shows the lowest. This helps in analyzing the range of experience to understand the concentration and distribution of human capital across the different roles in our dataset.

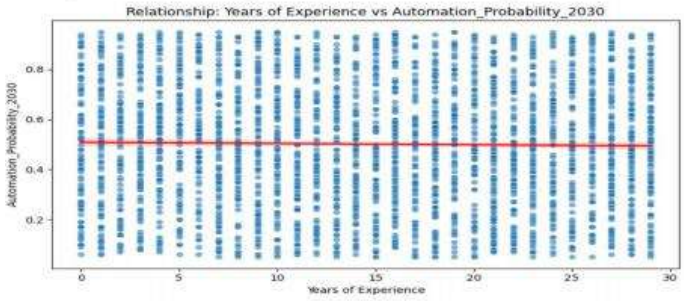


This visualization uses only the first five records of the dataset. The pie chart shows that the role of Security Guard requires the highest years of experience among the first five job titles, both currently and moving toward 2030. The Construction Worker role shows the least requirement, and based on the next chart it is likely to face higher exposure to AI-driven automation.

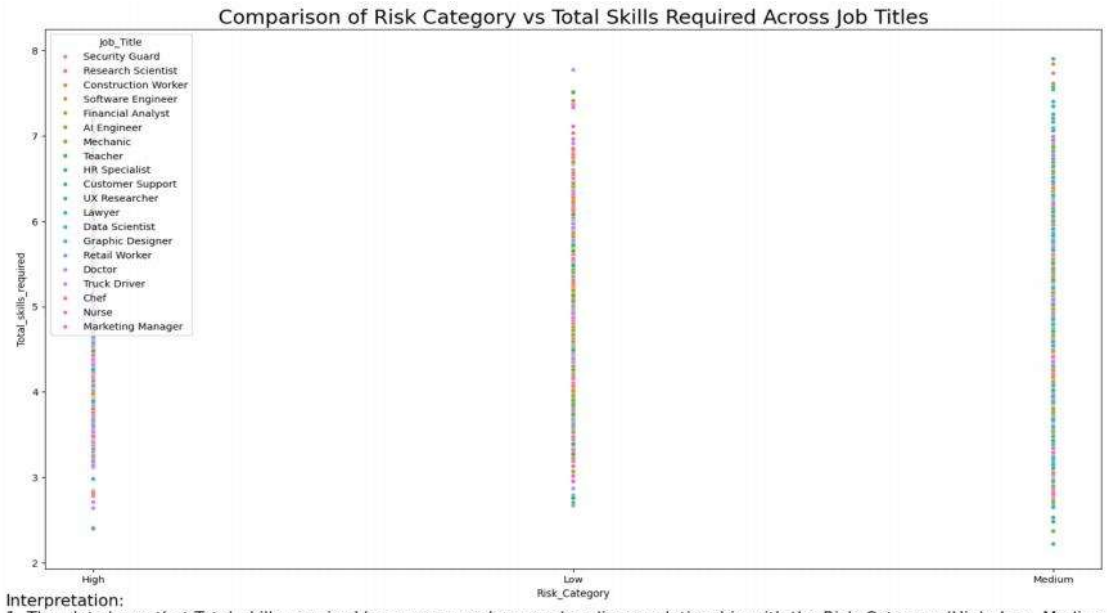


This visualization uses only the first five records of the dataset. According to the pie chart, Construction Workers show the highest AI exposure index among these roles, indicating they may face the greatest disruption by 2030. Security Guards, on the other hand, show the lowest exposure level, suggesting they are less likely to be heavily impacted by AI compared to the other occupations shown.

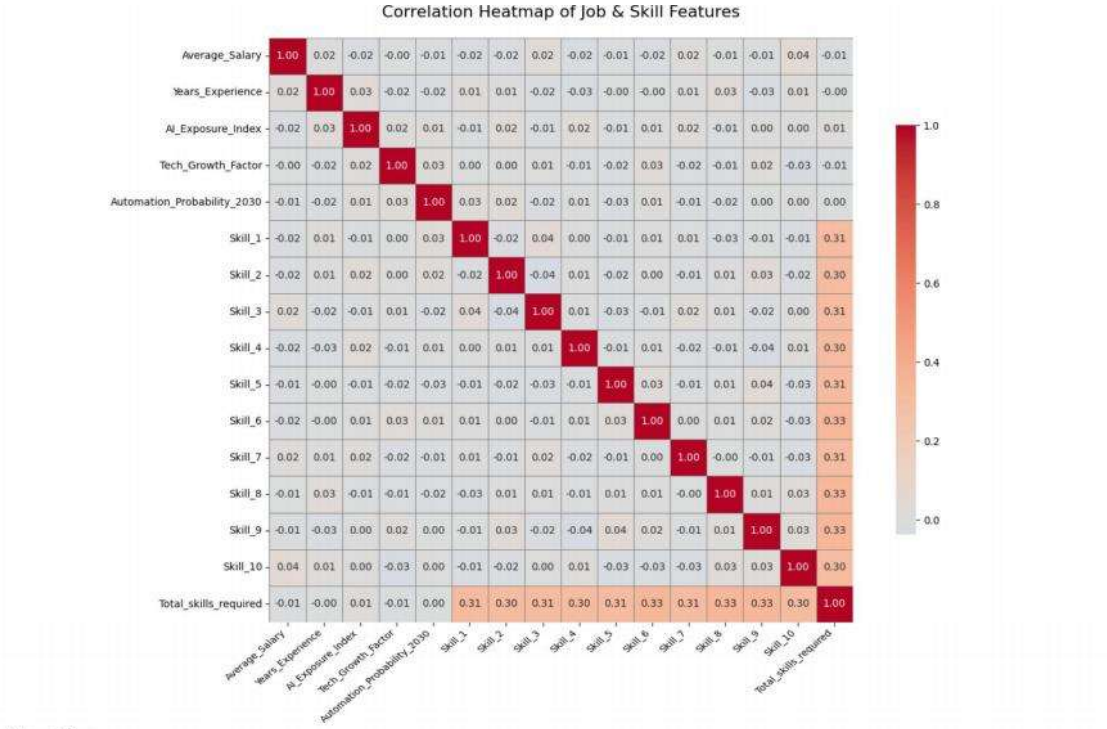
There is no strong relationship between Risk Category and AI Exposure Index in this dataset. Jobs across High, Medium, and Low risk groups show a similar spread of AI exposure values. This suggests that the impact of AI automation is more dependent on the nature of each job role and the skills involved, rather than simply the risk label alone.



People with more work experience are not necessarily safer from automation. The risk level remains almost constant, regardless of whether someone has 1 year or 30 years of experience. The automation probability appears randomly scattered, indicating weak or no correlation.



Interpretation:
1. The plot shows that Total_skills_required has a very weak or no clear linear relationship with the Risk_Category (High, Low, Medium).
2. Regardless of the risk category, the majority of observations fall within a narrow range (~2.5 to 7.5 skills).
3. Job Titles appear across all risk categories and skill levels, indicating no job is restricted to a single risk or skill bracket.



Interpretation:
1. This heatmap shows Pearson correlation coefficients (-1 to 1) for job-related factors and skills.
2. Total_skills_required has a strong positive relationship with individual skills (Skill 2, 3, 4, 10, etc.).
3. Average_Salary, Years_Experience, AI_Exposure_Index, and Tech_Growth_Factor show weak or near-zero correlations with specific skills.
4. The only noticeable non-skill correlation is between Average_Salary and Years_Experience, though it remains low.
5. The key takeaway: Skills are independent of salary, experience, or AI exposure, but they collectively shape Total_skills_required.

5.Conclusion:

In this mini project, we successfully orchestrated a complete **data pipeline** — from importing essential libraries (numpy, pandas, matplotlib, seaborn) to executing **data inspection, cleaning, transformation, and visualization routines**. By leveraging functions such as df.info(), df.shape, df.describe(), and df.groupby(), we ensured that the dataset was not only **syntactically valid** but also **semantically meaningful**. Through operations like dropna(), fillna(), and drop_duplicates(), we enforced **data integrity constraints**, thereby eliminating noise and redundancy.

The project further demonstrated the power of **vectorized operations** and **custom function application** (apply() with increaseBY2) to manipulate features such as Average_Salary, while **derived attributes** like Total_skills_required showcased the flexibility of **feature engineering**. Sorting algorithms (sort_values()) and filtering conditions (df[df['AI_Risk'] > 0.7]) allowed us to dynamically query subsets of the dataset, mimicking real-world **ETL workflows**.

On the analytical front, descriptive statistics (mean, min, max, std) provided a **quantitative snapshot** of the dataset, while **correlation matrices** and **heatmaps** revealed underlying relationships between variables. Visualizations (barplot, pie, pairplot) transformed raw data into **actionable insights**, bridging the gap between **code execution** and **decision-making**.

Ultimately, this project exemplifies how **Pythonic paradigms** — clean syntax, modular functions, and reproducible workflows — can be harnessed to decode the complex narrative of AI's impact on jobs by 2030. The seamless integration of **dataframes, aggregation pipelines, and visualization layers** reflects not just a technical exercise, but a **scalable blueprint** for future research in data science.

6.References:

- Nexford University. *How will Artificial Intelligence Affect Jobs 2026–2030.* Published October 20, 2025. Available at: Nexford Insights
- EY India. *Unlocking productivity gains: GenAI to transform 38 million jobs by 2030.* Press release, January 14, 2025. Available at: EY India Newsroom
- IBEF (India Brand Equity Foundation). *Artificial Intelligence set to reshape 38 million jobs in India by 2030.* Published January 15, 2025. Available at: IBEF News
- (links below)

https://www.kaggle.com/datasets/khushikyad001/ai -impact-on-jobs-2030
https://papers.ssrn.com/sol3/Delivery.cfm/5213331.pdf?abstractid=5213331&mirid=1