

**Name – Anshuman Behera**

**Registration Number – 12106905**

**ROLL no – RK21UPA17**

**SECTION - K21UP**

**Submitted to - Anjan R**

**Dataset name - PRODUCTIVITY PREDICTION OF GARMENT INDUSTRY EMPLOYEES**

**PRODUCTIVITY PREDICTION OF GARMENT INDUSTRY EMPLOYEES**

**Introduction**

The garment industry, a powerhouse in global commerce with a trillion-dollar valuation, specializes in the manufacturing of diverse clothing. Renowned for its labour- and capital-intensive nature, the industry's vitality hinges on optimizing workforce productivity. In practical terms, workforce productivity is a measure of the goods and services generated by a team of workers within a specific timeframe. The higher the productivity levels achieved by a team, the greater the potential for increased profits. This efficiency is indicative of optimal resource utilization throughout the production process. For instance, in real-time applications, leveraging advanced technologies such as data analytics and machine learning can enhance predictive maintenance strategies, ensuring machines operate efficiently and minimizing downtime. By maximizing worker productivity, the garment industry not only streamlines production processes but also fortifies its position in a highly competitive market.

**Reason for Choosing Dataset**

Low workforce productivity in a company can lead to reduced profitability, increased conflict, high employee turnover, and a lack of motivation. It is therefore important for a company to be able to track and predict its levels of productivity and investigate the factors that influence this.

**Metrics for success**

\* Building a model that can predict employee productivity (Best model should have the lowest RMSE of about 0.1 or less).

\* Identifying the top factors influencing the productivity level of employees.

**Domain knowledge**

**1.** Garment Industry Overview:

Types of Garment Manufacturing: Understand different types of garment manufacturing processes, such as mass production, bespoke tailoring, and ready-to-wear.

Supply Chain: Familiarity with the supply chain, including sourcing raw materials, design, production, and distribution processes.

Workforce Structure: Knowledge of the diverse roles in the industry, such as tailors, pattern makers, cutters, and sewing machine operators.

**2**. Productivity Metrics

Units of Measurement: Understanding productivity metrics like pieces produced per hour, garments sewn per day, or lead time required for a specific task.

Quality Control: Awareness of quality standards and how they impact productivity. Defects and rework significantly affect overall output.

**3.** Factors Influencing Productivity

Skill Levels: Differences in skill levels among workers greatly affect productivity. Experienced workers often perform tasks more efficiently.

Workflow Optimization: Efficient organization of workstations, tools, and materials to minimize downtime and streamline processes.

Training: Proper training programs to enhance skills and keep workers updated with new techniques and technologies.

Motivation and Morale: Employee satisfaction, motivation, and overall morale significantly impact productivity levels.

Health and Safety: Compliance with health and safety regulations to prevent accidents and maintain a healthy workforce.

**4.** Technology and Automation:

Sewing Machinery: Understanding different types of sewing machines, their capabilities, and how automation impacts productivity and quality.

CAD/CAM Systems: Knowledge of Computer-Aided Design (CAD) and Computer-Aided Manufacturing (CAM) systems used for pattern making and cutting.

5. Data Collection and Analysis

Data Sources: Identifying sources of data, including sensors on machinery, manual input, and historical records.

Data Preprocessing: Cleaning and preparing data for analysis, dealing with missing values, outliers, etc.

Feature Selection: Identifying relevant features affecting productivity, such as worker experience, machine type, order complexity, etc.

Model Selection: Choosing appropriate machine learning or statistical models for prediction, considering accuracy, interpretability, and scalability.

Evaluation Metrics: Using metrics like Mean Absolute Error (MAE), Root Mean Square Error (RMSE), or R-squared to evaluate the performance of predictive models.

6. Regulations and Compliance:

Labor Laws: Knowledge of labor laws and regulations related to working hours, wages, and conditions, which can affect workforce productivity.

7. Regulations and Compliance:

Fashion Trends: Awareness of current fashion trends impacting demand, which, in turn, affects production volume and workforce requirements.

Demand Forecasting: Techniques for predicting demand for specific garment types or designs, helping in resource allocation and workforce planning.

**Column definitions:**

1. date : Date in MM-DD-YYYY

2. day : Day of the Week

3. quarter : A portion of the month. A month was divided into four quarters

4. department : Associated department with the instance

5. team no : Associated team number with the instance

6. no\_of\_workers : Number of workers in each team

7. no\_of\_style\_change : Number of changes in the style of a particular product

8. targeted\_productivity : Targeted productivity set by the Authority for each team for each day.

9. smv : Standard Minute Value, it is the allocated time for a task

10. wip : Work in progress. Includes the number of unfinished items for products

11. over\_time : Represents the amount of overtime by each team in minutes

12. incentive : Represents the amount of financial incentive (in BDT) that enables or motivates a particular course of action.

13. idle\_time : The amount of time when the production was interrupted due to several reasons

14. idle\_men : The number of workers who were idle due to production interruption

15. actual\_productivity : The actual % of productivity that was delivered by the workers.

**Libraries used and approaches.**

**1. Pandas**: Pandas is a powerful Python library for data manipulation and analysis, providing data structures like Data Frame and Series for efficient data handling, cleaning, and transformation.

**2.NumPy**: NumPy is a fundamental package for scientific computing in Python, enabling efficient handling of large, multi-dimensional arrays and matrices, along with a variety of mathematical functions to operate on these arrays.

**3. Seaborn**: Seaborn is a statistical data visualization library based on Matplotlib. It provides a high-level interface for creating attractive and informative statistical graphics, simplifying the process of visualizing complex datasets and relationships.

**4. Matplotlib**: Matplotlib is a comprehensive plotting library in Python, used to create static, interactive, and animated visualizations in various formats. It offers fine-grained control over the appearance of plots, charts, and figures for data visualization.

Top of Form

**Steps OF EDA**

**Step 1: Understand the Data**

Objective: Understand the context of the data and the problem you're trying to solve.

Approach:

Read and load the dataset into a data structure (e.g., Pandas DataFrame).

Look at the first few rows of the dataset (df.head()) to understand the structure.

Identify the features (columns) and their data types.

Check for any data dictionary or documentation to understand the meaning of each feature.

**Step 2: Clean the Data**

Objective: Handle missing values, outliers, and any inconsistencies in the dataset.

Approach:

Check for missing values using df.isnull().sum(). Decide whether to remove or fill missing values based on the context.

Identify and handle outliers using statistical methods like IQR (Interquartile Range) or visualizations like box plots.

Handle duplicate records if any.

Convert data types if necessary (e.g., converting object types to numeric types).

**Step 3: Univariate Analysis**

Objective: Understand the characteristics of individual features.

Approach:

For numerical features:

Calculate basic statistics (mean, median, standard deviation, etc.) using df. describe().

Visualize the distribution using histograms (df. hist()) or kernel density plots.

For categorical features:

Count the unique values (df['column\_name'].value\_counts()) and visualize using bar plots (sns.countplot()).

**Step 4: Bivariate Analysis**

Objective: Explore relationships between pairs of features.

Approach:

For numerical vs. numerical features:

Use scatter plots (plt.scatter()) or pair plots (sns.pairplot()) to visualize correlations.

Calculate correlation coefficients using df.corr() and visualize as a heatmap.

For categorical vs. numerical features:

Use box plots (sns.boxplot()) to compare distributions across different categories.

Calculate group-wise statistics and visualize using bar plots.

**Step 5: Multivariate Analysis**

Objective: Explore relationships involving more than two features.

Approach:

Utilize advanced visualizations like 3D scatter plots or create interaction features to explore multivariate relationships.

Use techniques like PCA (Principal Component Analysis) for dimensionality reduction and visualization.

**Step 6: Draw Insights and Conclusions**

Objective: Summarize the patterns, relationships, and insights gained from the analysis.

Approach:

Document notable findings, correlations, outliers, and trends.

Identify variables that significantly impact the target variable.

Formulate hypotheses for further testing or modeling.

**Step 7: Communicate Results**

Objective: Present the findings and insights to stakeholders.

Approach:

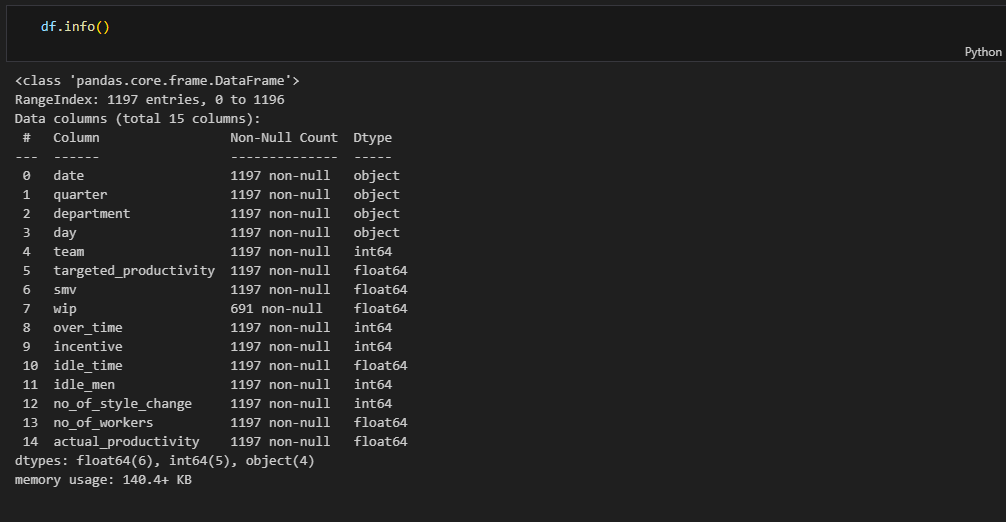
Create clear and informative visualizations to illustrate key points.

Prepare a summary report or presentation, explaining the analysis, findings, and recommendations.

Engage with

**Questions**

1. What are the names and data types of the columns?



df.info() is a Pandas function that prints a concise summary of a Data Frame. This summary includes information about the number of rows and columns, the data types of the columns, and the number of non-null values in each column.

2.What are the basic summary statistics?

A screenshot of a computer

Description automatically generated

df.describe() provides summary statistics for each numeric column in a data -frame , including the mean, median, mode, standard deviation, and percentiles. This information can be used to understand the central tendency and dispersion of the data, which can be helpful for identifying outliers and anomalies

3.Are there any categorical variables and missing values ? If so print it .

A black rectangular object with blue lines

Description automatically generated

* Identifying categorical variables that need to be encoded
* Identifying categorical variables that can be used to create features
* Identifying categorical variables that can be used to group data

A screen shot of a computer

Description automatically generated

* Use the isnull() function to identify all missing values in the DataFrame.
* Use the sum() function to count the number of missing values in each column.
* Use the dropna() function to drop rows or columns with missing values.
* Use the fillna() function to fill in missing values with estimated values.

4.Are there any outliers in the data? If so use box plots, histograms and visualize .

A screenshot of a computer program

Description automatically generated

1. Create a box plot of the data.
2. Identify the upper and lower quartiles (Q1 and Q3).
3. Calculate the interquartile range (IQR) by subtracting the lower quartile from the upper quartile (IQR = Q3 - Q1).
4. Draw whiskers from the upper and lower quartiles that extend to 1.5 times the IQR.
5. Any data points that fall outside of the whiskers are considered to be outliers

**4.Any data points that fall outside of the whiskers are considered to be outliers**

A screenshot of a computer

Description automatically generated

5. Is the data balanced or imbalanced? Visualize.

A screenshot of a computer

Description automatically generated

This code creates a count plot of the wip variable in the df DataFrame. The plot is colored using the Set1 color palette. The title of the plot is "Distribution of employee is working or not", and the x- and y-axis labels are "wip" and "Count", respectively.

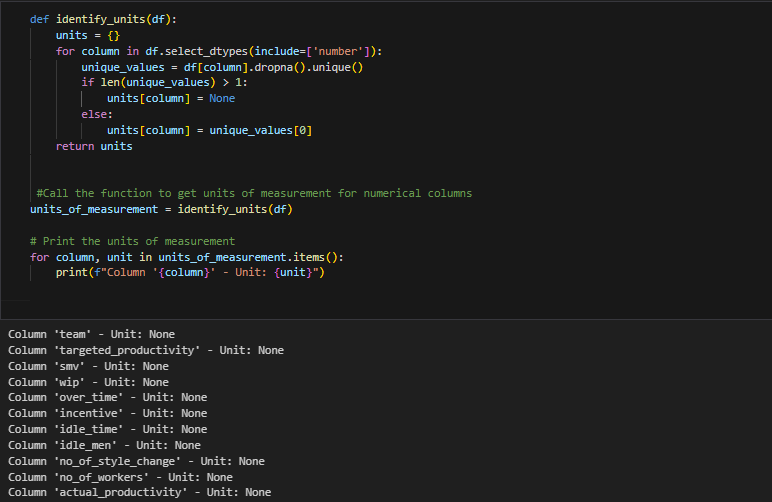
What is the target variable (if any)

A screenshot of a computer program

Description automatically generated

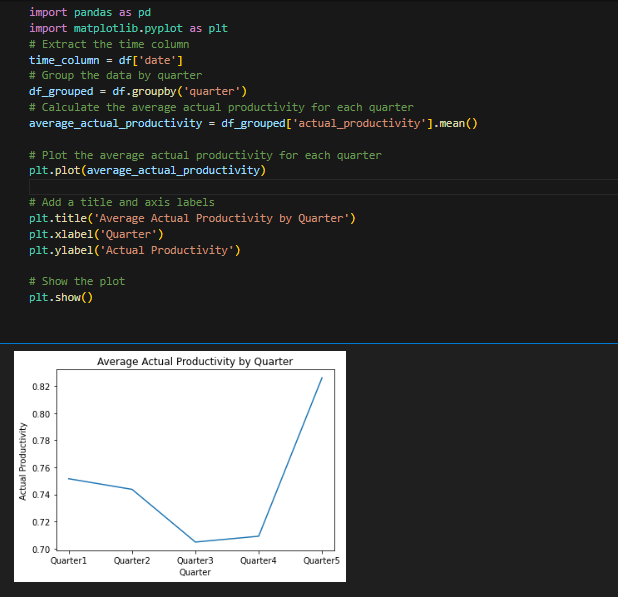
Here we are able to find the target variable

7. What are the units of measurement for numerical columns



The identify\_units() function can be used for a variety of purposes, such as:

* To identify columns with missing units of measurement: The function can be used to identify columns that contain measurements in different units or columns that have missing units of measurement. This information can be used to clean the data or to standardize the units of measurement in the DataFrame.
* To troubleshoot DataFrames: If you are having problems with a machine learning algorithm or another data analysis tool, the identify\_units() function can be used to check to see if the units of measurement in the DataFrame are consistent.
* To document the data: The function can be used to generate a list of the units of measurement for the columns in a DataFrame. This information can be used to document the data and to make it easier for others to understand the data.

9. Are there any time-based trends and pattern ?

The code you provided is a useful tool for visualizing and analyzing data on productivity. By generating a line plot of the average actual productivity for each quarter, you can identify trends in productivity and make informed decisions about how to improve productivity.

10. Are there any correlations between variables? Calculate correlation

A screenshot of a computer program

Description automatically generated

Overall, the correlation matrix is a valuable tool for building machine learning models. By using the correlation matrix to identify and remove highly correlated variables, identify features that are important for prediction, and understand the relationships between variables, you can improve the performance of your models.

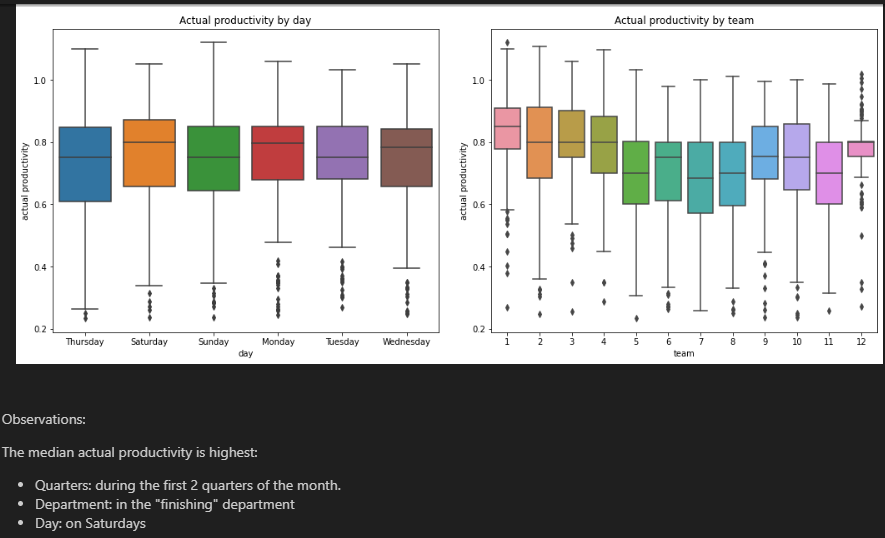
11. What is the specific objective or goal of predicting productivity in the garment industry?

A screen shot of a computer

Description automatically generated

* To track productivity over time: You can use this code to calculate the average productivity for each month, quarter, or year. This information can be used to track productivity over time and to identify trends.
* To compare productivity between different groups: You can use this code to calculate the average productivity for different groups of employees, such as employees in different departments or employees with different job titles. This information can be used to identify groups where productivity is low and to take steps to improve productivity.
* To set performance goals: You can use this code to calculate the average productivity for a group of employees and then set performance goals for the group. For example, you might set a goal of increasing average productivity by 5% in the next quarter.

12. Actual productivity by day of week, department, quarter of the month, team

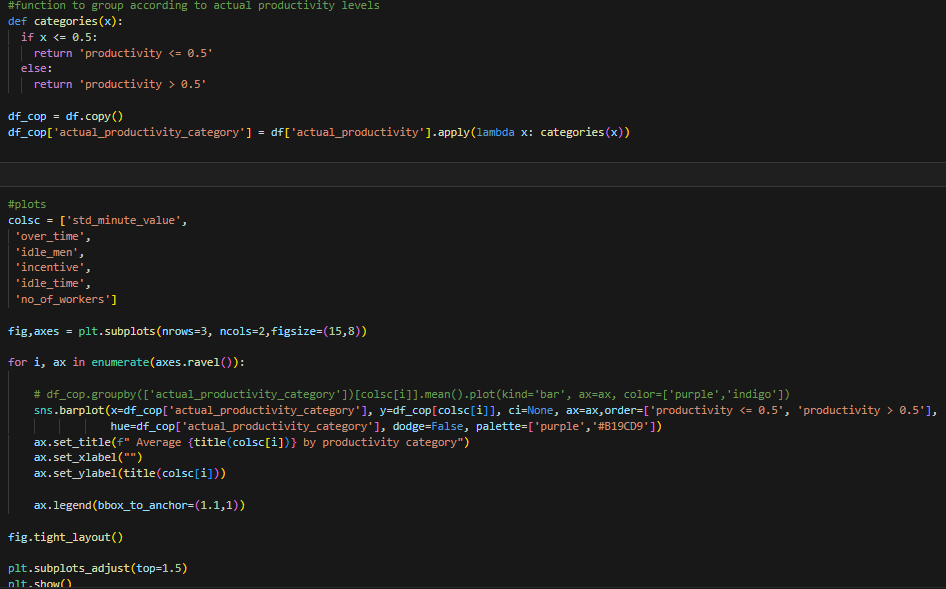


* A screenshot of a computer screen

  Description automatically generated

the code you provided creates a 2x2 grid of box plots that show the distribution of the actual\_productivity variable for each category in the cat list. This information can be used to identify categories where productivity is low and to investigate the reasons for this.

13. Average standard minute value, over time, idle men,incentive, idle time and no of workers by productivity category (>0.5 or not)



A close-up of a graph

Description automatically generated

The code appears to analyze productivity metrics over time, categorizing them based on a threshold (likely 0.5). Metrics like standard minute value, idle men, incentive, idle time, and the number of workers are used to assess productivity levels. The output offers insights into how these metrics fluctuate over time and whether there's a correlation with productivity, specifically when it exceeds or falls below the specified threshold. Without the exact code or output, a more detailed explanation is challenging. If you provide more details, I can offer further assistance.

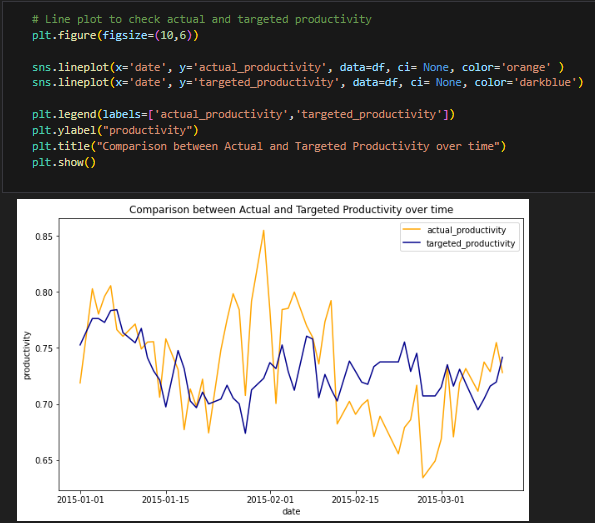
13. Comparison between Actual Productivity and incentives over time

A screen shot of a computer screen

Description automatically generated

The code analyzes the temporal trends of actual productivity and incentives up to March 8, 2015. It uses Seaborn to create a line plot with dates on the x-axis and the corresponding values for actual productivity (in orange) and incentives (in dark green) on the y-axes. The dual-axis setup allows for a clear comparison between the two trends. The legend distinguishes between the lines, and the plot is titled "Incentives and actual productivity over time." This visualization provides a concise representation of how actual productivity and incentives have changed over the specified period.

**14. Comparison between Actual and Targeted productivity out come of it**



The code generates a line plot comparing the trends of actual and targeted productivity over time. It uses Seaborn to create two lines on the same plot: one in orange representing actual productivity and another in dark blue representing targeted productivity. The legend distinguishes between the two lines, and the y-axis is labeled "productivity." The resulting visualization provides a quick overview of how actual and targeted productivity have evolved over the dataset's timeframe. This type of plot is useful for identifying patterns, deviations, or similarities between the actual and targeted productivity values.

16.Actual Productivity per Department over time





The provided code generates a line plot to examine the variation in actual productivity across different departments over time. Using Seaborn, the x-axis represents dates ('date'), the y-axis represents 'actual\_productivity', and different departments are distinguished by color and style. Each department is assigned a unique color, and the line style differentiates them further. Markers are added to emphasize individual data points. The resulting plot provides a visual representation of how actual productivity varies across different departments throughout the specified time period. The title, "Actual Productivity per Department over time," summarizes the focus of the visualization.

17. Idle time by department

A screen shot of a computer

Description automatically generated

The code generates a bar plot using Seaborn to illustrate the distribution of idle time across different departments, with additional differentiation based on teams. The x-axis represents the departments, the y-axis represents the 'idle\_time' values, and different colors within each bar indicate different teams. The height of the bars represents the amount of idle time, and the plot provides a visual comparison of idle time across departments and teams. The title, "Idle time by department," succinctly conveys the focus of the visualization. This type of plot is useful for quickly identifying which departments and teams experience higher or lower levels of idle time.

**18 . Idle men by department**A screenshot of a computer

Description automatically generated

The code generates a bar plot using Seaborn to display the total count of idle men in different departments, with further differentiation based on teams. The x-axis represents departments, the y-axis represents the total count of 'idle\_men,' and different colors within each bar represent various teams. The plot provides a visual comparison of the cumulative idle men across departments and teams. The title, "Idle men by department," succinctly describes the main focus of the visualization. This type of plot is effective for quickly identifying which departments and teams contribute more to the total count of idle men in the dataset.

19. Department and team with more style changes

**A screen shot of a computer

Description automatically generated**

The provided code generates a bar plot using Seaborn to depict the total count of style changes in different departments, with further differentiation based on teams. The x-axis represents departments, the y-axis represents the total count of 'no\_of\_style\_change,' and different colors within each bar represent various teams. The plot visually compares the cumulative style changes across departments and teams, allowing for a quick assessment of which departments and teams contribute more to the total count of style changes in the dataset. The title, "Department and team with more style changes," succinctly conveys the primary focus of the visualization. This type of plot is useful for identifying patterns and variations in style changes within organizational units

**20. Average Productivity of each team**

**A screenshot of a computer screen

Description automatically generated**

The provided code utilizes Seaborn's catplot to create a categorical bar plot, illustrating the average actual productivity across different departments, with additional differentiation based on teams. The x-axis represents departments, the y-axis represents the 'actual\_productivity,' and different colors within each bar denote various teams. The plot visually compares the average actual productivity levels, allowing for insights into both departmental and team-based variations. The title, "Average actual productivity per department," succinctly communicates the primary focus of the visualization. This type of plot is effective for gaining a quick overview of how actual productivity varies within and between departments, as well as among different teams within those departments.

**hypothesis testing**

Performing hypothesis testing involves a series of steps to assess whether a claim about a population parameter is supported by sample data. In the context of your dataset on garment industry productivity, let's outline the general steps for hypothesis testing:

**Formulate the Hypotheses**:

**Null Hypothesis (H\_0**):Typically, the null hypothesis states that there is no effect or no difference. For example, in the context of predicting employee productivity, the null hypothesis might state that there is no significant relationship between certain factors and productivity.

**Alternative Hypothesis** :This is the hypothesis that you want to test. It could suggest a difference or an effect. In the example, the alternative hypothesis might propose that specific factors significantly influence employee productivity.

**2. Choose a Significance Level :**

- This is the probability of rejecting the null hypothesis when it is true. Common choices are 0.05, 0.01, or 0.10.

**3. Select the Test Statistic:**

- Depending on your hypothesis and the type of data you have, choose an appropriate statistical test. Common tests include t-tests, ANOVA, chi-square tests, regression analysis, etc.

**4. Collect and Analyze Data:**

- Use the dataset to calculate the test statistic. This involves summarizing the data, calculating means, standard deviations, and other relevant statistics depending on the chosen test.

**5. Determine the Critical Region:**

- Based on the significance level (\(\alpha\)), determine the critical region — the values of the test statistic that would lead to rejection of the null hypothesis.

**6. Make a Decision:**

- Compare the calculated test statistic to the critical value. If it falls into the critical region, reject the null hypothesis. Otherwise, fail to reject the null hypothesis.

**7. Draw Conclusions:**

- Based on the decision in step 6, draw conclusions about the null hypothesis and what it means in the context of your study.

**8. Report Results:**

- Communicate the results of your hypothesis test, including the test statistic, critical values, p-values, and your conclusion.

**Conclusion:**

In conclusion, the chosen dataset holds significant relevance in addressing the critical challenges faced by the garment industry in optimizing workforce productivity. As highlighted, the garment industry's success is intricately tied to the efficient utilization of labour and capital resources. This dataset becomes instrumental in unlocking insights and solutions to enhance productivity, streamline production processes, and ultimately bolster the industry's competitive edge.

The imperative of understanding and predicting workforce productivity in real-time is underscored by the potential consequences of low productivity, including reduced profitability, increased conflict, high employee turnover, and diminished motivation. The dataset serves as a valuable resource for building predictive models that can accurately forecast employee productivity, with the defined metric of achieving a low Root Mean Square Error (RMSE) of 0.1 or less. This precision in prediction is pivotal for companies aiming to proactively address productivity issues and implement targeted strategies for improvement.

Furthermore, the dataset facilitates the identification of top factors influencing employee productivity. Through advanced technologies like data analytics and machine learning, companies can gain insights into the nuanced dynamics affecting workforce efficiency. Armed with this knowledge, organizations can tailor interventions and strategies to optimize these influential factors, leading to a more productive and motivated workforce.

In essence, the chosen dataset not only aligns with the industry's overarching goals of maximizing productivity but also provides a robust foundation for data-driven decision-making. By leveraging the insights derived from this dataset, the garment industry can foster innovation, implement targeted improvements, and navigate the competitive landscape with agility. The pursuit of enhanced workforce productivity, guided by the findings of this dataset, emerges as a strategic imperative for companies seeking sustained success in the ever-evolving global market.