MAKERERE UNIVERSITY

SEMESTER ONE 2024/2025 ACADEMIC YEAR

SCHOOL COMPUTING AND IMFORMATICS TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

MASTER OF SCIENCE IN COMPUTER SCIENCE

MCS 7102

DATA SECURITY AND PRIVACY

ASSIGNMENT ONE

KHADIJA ATHMAN

2024/HD05/21918U

2400721918

Machine learning Exploratory Data Analysis Report

1. Where was the dataset sourced from and for what purpose?

This dataset was picked from UCI databases based on the research article "mining the productivity data for garment industry" by Abdullah AL Iman et al. The dataset consists of a number of variables used to predict productivity range of workers in a garment making factory either by regression or classification.

2. What is the nature of the data that is stored in the dataset?

I explore the representation and type of data stored in the dataframe using functions like .head(), .tail(), info(), describe(), shape. From the summary display of the table, i observe the following

```
#Displaying the columns and their data types
pdt.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1197 entries, 0 to 1196
Data columns (total 15 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   date                   1197 non-null   object
 1   quarter                1197 non-null   object
 2   department             1197 non-null   object
 3   day                    1197 non-null   object
 4   team                   1197 non-null   int64
 5   targeted_productivity  1197 non-null   float64
 6   smv                    1197 non-null   float64
 7   wip                    691 non-null    float64
 8   over_time              1197 non-null   int64
 9   incentive              1197 non-null   int64
 10  idle_time              1197 non-null   float64
 11  idle_men               1197 non-null   int64
 12  no_of_style_change     1197 non-null   int64
 13  no_of_workers          1197 non-null   float64
 14  actual_productivity    1197 non-null   float64
dtypes: float64(6), int64(5), object(4)
memory usage: 140.4+ KB
```

```
#checking the number of r
pdt.shape
```

```
(1197, 15)
```

```
[ ] #displaying the first 10 records in the dataframe
    pdt.head(10)
```

	date	quarter	department	day	team	targeted_productivity	smv	wip	over_time	incentive	idle_time	idle_men	no_of_style_change	no_of_workers	actual_productivity
0	1/1/2015	Quarter1	sweing	Thursday	8	0.80	26.16	1108.0	7080	98	0.0	0	0	59.0	0.940725
1	1/1/2015	Quarter1	finishing	Thursday	1	0.75	3.94	NaN	960	0	0.0	0	0	8.0	0.886500
2	1/1/2015	Quarter1	sweing	Thursday	11	0.80	11.41	968.0	3660	50	0.0	0	0	30.5	0.800570
3	1/1/2015	Quarter1	sweing	Thursday	12	0.80	11.41	968.0	3660	50	0.0	0	0	30.5	0.800570
4	1/1/2015	Quarter1	sweing	Thursday	6	0.80	25.90	1170.0	1920	50	0.0	0	0	56.0	0.800382
5	1/1/2015	Quarter1	sweing	Thursday	7	0.80	25.90	984.0	6720	38	0.0	0	0	56.0	0.800125
6	1/1/2015	Quarter1	finishing	Thursday	2	0.75	3.94	NaN	960	0	0.0	0	0	8.0	0.755167
7	1/1/2015	Quarter1	sweing	Thursday	3	0.75	28.08	795.0	6900	45	0.0	0	0	57.5	0.753683
8	1/1/2015	Quarter1	sweing	Thursday	2	0.75	19.87	733.0	6000	34	0.0	0	0	55.0	0.753098
9	1/1/2015	Quarter1	sweing	Thursday	1	0.75	28.08	681.0	6900	45	0.0	0	0	57.5	0.750428

From the summary display of columns and their data types, i observe the following

- i table that has 1197 rows and 15 columns
- the column types are both string and floating point

the following is a brief description of the columns in the dataset and their data types;

1. date - this is the date when the data was collected. Data type **Object**

2. Quarter - A portion of the month. A month was divided into four quarters. Data type **Object**

3. department - Associated department with the instance i.e finishing and sewing. Data type **Object**

4. day - Day of the Week i.e monday, tuesday, wednesday…Data type **Object**

5. team - Associated team number with the instance. Data type **Integer**

6. targeted_productivity - Targeted productivity set by the Authority for each team for each day represented between a range of 0 to 1. Data type **Float**

7. smv - Standard Minute Value, it is the allocated time for a task. Data type **Float**

8. wip - Work in progress. Includes the number of unfinished items for products . Data type **Float**

9. over_time - extra time worked in terms of hours .Data type **Float**

10. incentive - how much an employee will be compensated for assigned work. Data type **Float**

11. idle_time - the time in which the employee was not doing anything work related for many reasons .Data type **Float**

12. idle_men - Number of workers who were idle due to production interruption. Data type **Integer**

13. no_of_style_change - Number of changes in the style of a particular product represented in 0, 1 and 2. Data type **Integer**

14. no_of_workers - number of workers assigned a given task.Data type **Float**

15. actual_productivity - actual productivity of the employees on a given day for a given team represented between a range of 0 to 1. Data type **Float**

3. Is the dataframe clean

Here i will check for any duplicate values, missing values, wrong representations

- Doing a check for missing data, it is shown that the data has missing values in the 'wip' (Work in progress) column. The number of missing values is 506 which contributes to 49.746 % of the total data for the column.

```
[ ] #checking for any missing values in the dataset
    pdt.isnull().sum()
```

	0
date	0
quarter	0
department	0
day	0
team	0
targeted_productivity	0
smv	0
wip	506
over_time	0
incentive	0
idle_time	0
idle_men	0
no_of_style_change	0
no_of_workers	0
actual_productivity	0

dtype: int64

- It's also evident that there was a data entry problem in the column 'department' in that the number of unique departments is two but the unique() function shows 3.

That is ‘sweing’ which is misspelled ,‘finishing’ and ‘finishing ’ . The figure below shows that there was a spelling error and an extra white space most likely during data entry.

```
# filtering out unique values of the column department in an array
pdt['department'].unique()
```

```
array(['sweing', 'finishing ', 'finishing'], dtype=object)
```

- The quarter column has 5 unique quarters instead of 4.

- The data type of date is reprepresented as a string rather than datetime

4. Is the data accurately represented

Are there any outliers or miss representations in the dataset

THe accuracy of the data can be checked through visualizing and understanding the statistical summary of the data, that is to say the mean, standard deviation, minimum value and maximum value.

```
[ ] pdt.describe()
```

	team	targeted_productivity	smv	wip	over_time	incentive	idle_time	idle_men	no_of_style_change	no_of_workers	actual_productivity
count	1197.000000	1197.000000	1197.000000	691.000000	1197.000000	1197.000000	1197.000000	1197.000000	1197.000000	1197.000000	1197.000000
mean	6.426901	0.729632	15.062172	1190.465991	4567.460317	38.210526	0.730159	0.369256	0.150376	34.609858	0.735091
std	3.463963	0.097891	10.943219	1837.455001	3348.823563	160.182643	12.709757	3.268987	0.427848	22.197687	0.174488
min	1.000000	0.070000	2.900000	7.000000	0.000000	0.000000	0.000000	0.000000	0.000000	2.000000	0.233705
25%	3.000000	0.700000	3.940000	774.500000	1440.000000	0.000000	0.000000	0.000000	0.000000	9.000000	0.650307
50%	6.000000	0.750000	15.260000	1039.000000	3960.000000	0.000000	0.000000	0.000000	0.000000	34.000000	0.773333
75%	9.000000	0.800000	24.260000	1252.500000	6960.000000	50.000000	0.000000	0.000000	0.000000	57.000000	0.850253
max	12.000000	0.800000	54.560000	23122.000000	25920.000000	3600.000000	300.000000	45.000000	2.000000	89.000000	1.120437

1. Team is properly represented in that the maximum is 12, minimum is 1 standard deviation of 3.64 and a mean of 6. Meaning there are no outliers

2. Smv is fairly represented with a mean of 15.06, standard deviation of 10.94, minimum of 2.9 and maximum of 54.

3. Incentive appears to have a very big difference between the minimum and maximum which shows that there are outliers.

4.idle_time and idle_mean equally have a silently big difference the minimum and maximum hence presence of outliers

Generally the data contains outliers in some columns such as incentive, idle_time, idle_mean,

5. **Are there any correlations between the different variables in the dataframe?**

Using graphs such as pair plots, scatter plots bar plot and heat maps, the correlation between the different variables was observed

This shall be explained after the in depth data wrangling to fix the missing values and miss representations as they could affect the analysis

Data Wrangling

Identifying and removing duplicates

Running the duplication check on my dataframe, its shown that there are no duplicates

Fixing structural errors such as miss spellings

There is a miss spelled word and an extra white space under department which shows that there are three departments rather than two which is wrong on reviewing the data. There are actually only two departments.

The code below shows how i replace the misspelled word sweing and removing white space

```
# first before making changes i shall create a copy of the data frame so that i have the original for ret
pdt_clean = pdt.copy()

# replacing miss-spelt strings and removing white spaces in strings under department column
pdt_clean['department'] = pdt_clean['department'].str.replace('finishing ', 'finishing')
pdt_clean['department'] = pdt_clean['department'].str.replace('sweing', 'sewing')

#checking the output
pdt_clean['department'].unique()
```

```
array(['sewing', 'finishing'], dtype=object)
```

Fixing the quarter values to have only 4 quarters rather than 5 quarters.

An investigation of the quarter column shows that the Quarter5 label only appears in the month January for which the dates are above 28. To fix this i.e have only 4 quarters, i shall consider all Quarter5 labels to be Quarter4 since the dates are closer to the ones that fall on Quarter4.

```
#replacing quarter5 with quarter4
pdt_clean['quarter'] = pdt_clean.quarter.str.replace('Quarter5', 'Quarter4')


#checking that there are now 4 quarters only
pdt_clean.quarter.unique()
```

```
array(['Quarter1', 'Quarter2', 'Quarter3', 'Quarter4'], dtype=object)
```

Converting data types

The date column data type is ‘string’ rather than ‘datetime’ therefore this need to be change to the right data type

```
#converting date data type to datetime rather than string
pdt_clean['date'] = pd.to_datetime(pdt_clean['date'])

#checking the new output
pdt_clean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1197 entries, 0 to 1196
Data columns (total 15 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   date                   1197 non-null   datetime64[ns]
 1   quarter                1197 non-null   object
 2   department             1197 non-null   object
 3   day                    1197 non-null   object
 4   team                   1197 non-null   int64
 5   targeted_productivity  1197 non-null   float64
 6   smv                    1197 non-null   float64
 7   wip                    691 non-null    float64
 8   over_time              1197 non-null   int64
 9   incentive              1197 non-null   int64
 10  idle_time              1197 non-null   float64
 11  idle_men               1197 non-null   int64
 12  no_of_style_change     1197 non-null   int64
 13  no_of_workers          1197 non-null   float64
 14  actual_productivity    1197 non-null   float64
dtypes: datetime64[ns](1), float64(6), int64(5), object(3)
memory usage: 140.4+ KB
```

Handling missing values

The function to check for missing values returned that there are actually missing values in column ‘wip’ (work in progress). These missing values have to be imputed for better analysis. To do so, i shall replace all the missing values in this column with the mean value

```
#replacing missing values with the mean value
pdt_clean['wip'].fillna(pdt_clean['wip'].mean(), inplace=True)

#checking the output
pdt_clean['wip'].isnull().sum()
```

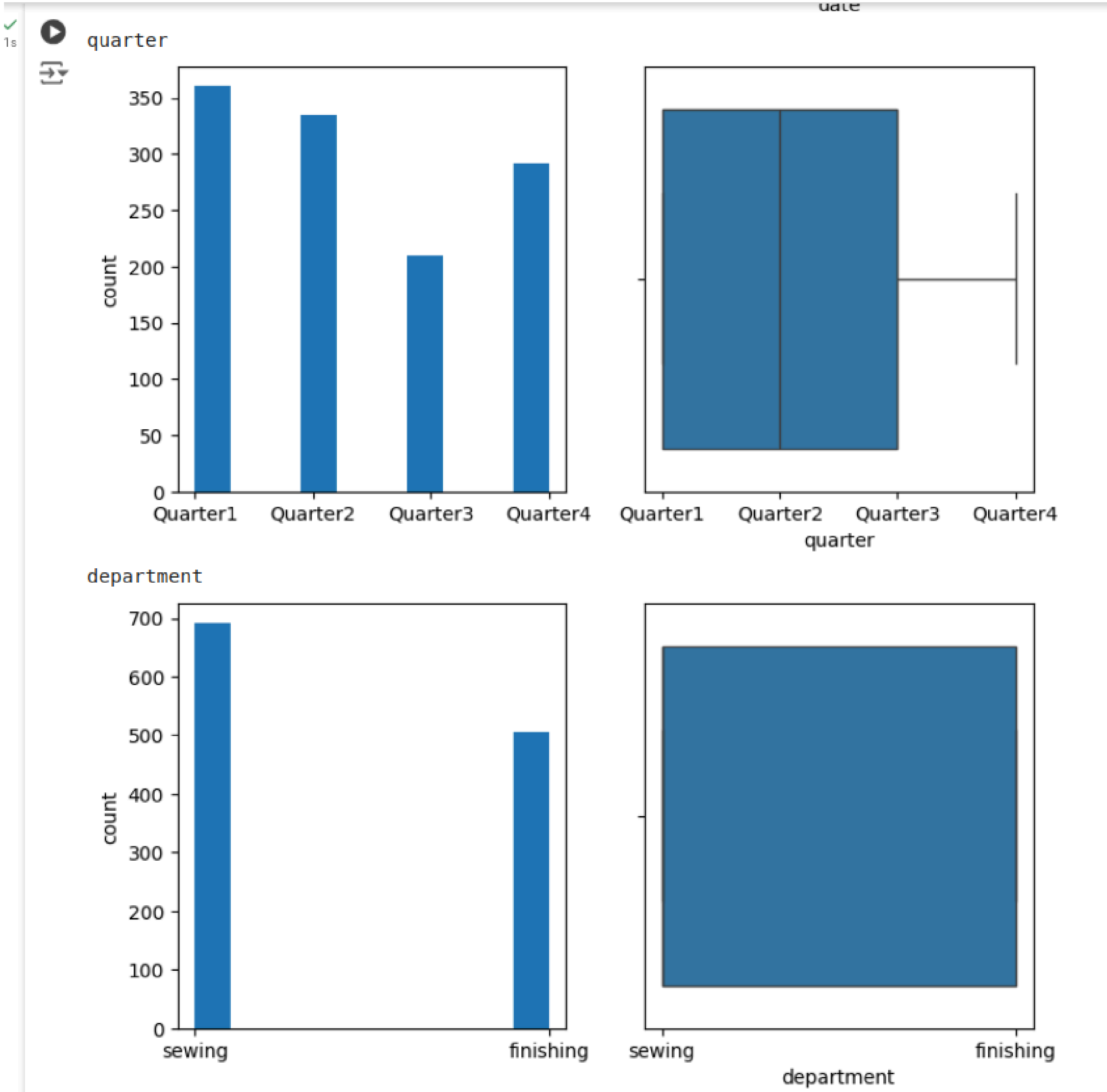
```
0
```

Exploratory data analysis

After performing the data wrangling as shown above, the data is well prepared for the further exploratory data analysis through the use of graphs

Investigation of categorical columns

Count plots for the categorical variables such as quarter, department, day.

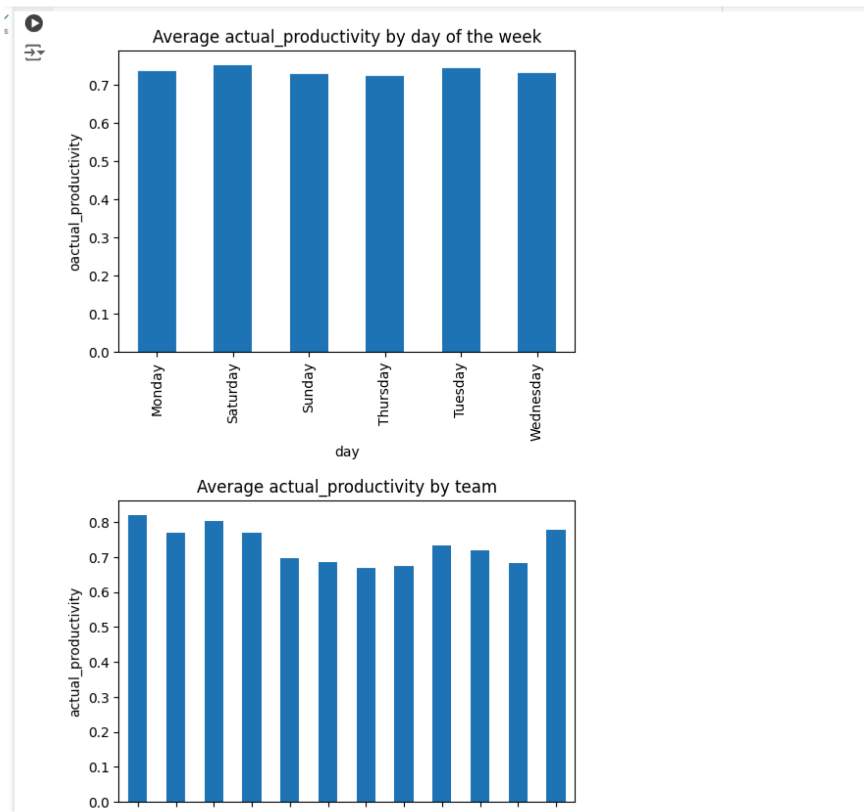


From the above observations, it is evident that;

- there are more records in the sewing department compared to the finishing department.
- under day column, wednesday has the most records followed by thursday.
- under quarter, quarter 1 had the most records compared to other quarters.

- under teams, 2 and 12 had the most records.

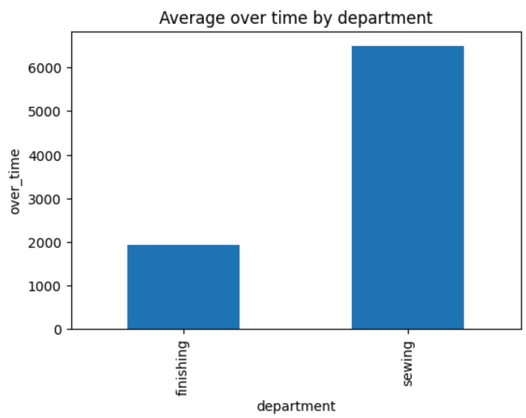
Bar plots showing the correlation between actual productivity and the categorical variables such as quarter, day, department, team

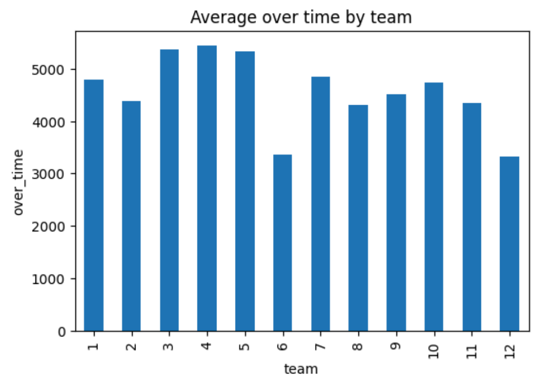


From the sample figure above, The mean actual productivity was recorded highest in:

- under quarters, quarter 1
- under department, the "finishing" department
- under day, on Saturdays
- under team, team 1

Bar plots showing the representation of department and teams against overtime



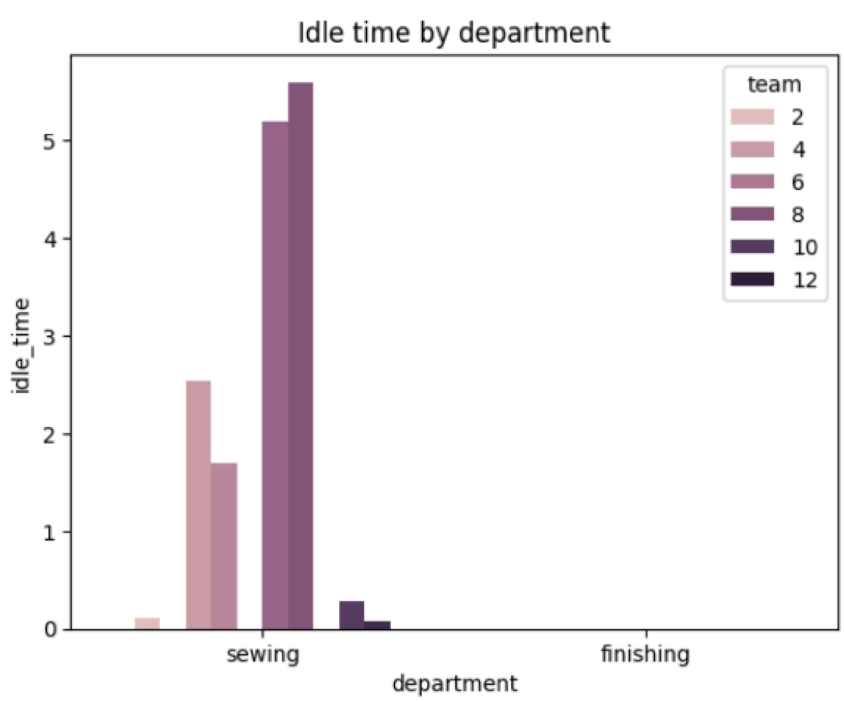


The above figure shows that

- The sewing department recorded significantly more overtime compared to the finishing department.
- The teams had a fairly even distribution of overtime recorded.

Investigating idle time by department

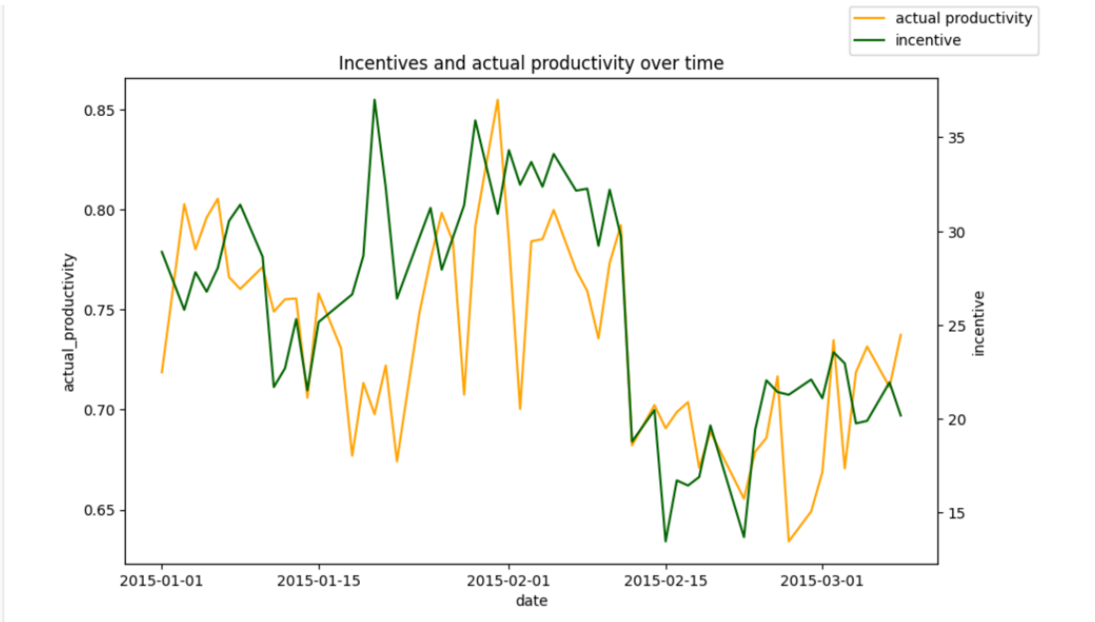
```
[19] sns.barplot(x='department', y='idle_time',hue ='team' ,data=pdt_clean, errorbar= None)
     plt.title("Idle time by department")
     plt.show()
```



- from the bar plot above, investigation show that only the sewing department had recorded idle_time and team 8 had the highest idle_time

Investigating continuous variables correlations

Investigating a time serie relation between incentives and actual productivity

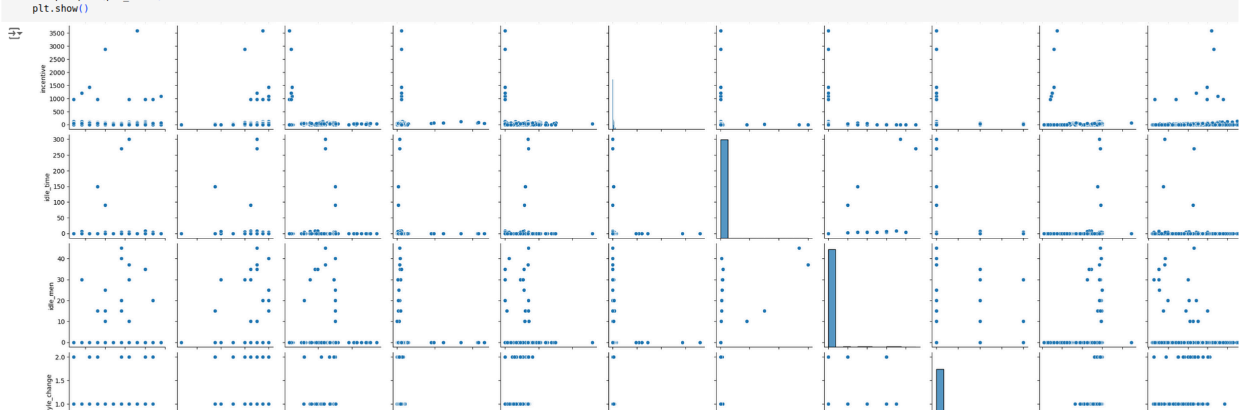


- The above figure shows that the actual productivity goes hand in hand with incentives, that is to say, lower incentives means lower actual productivity.

Scatter plot showing how all variables are correlated with each other

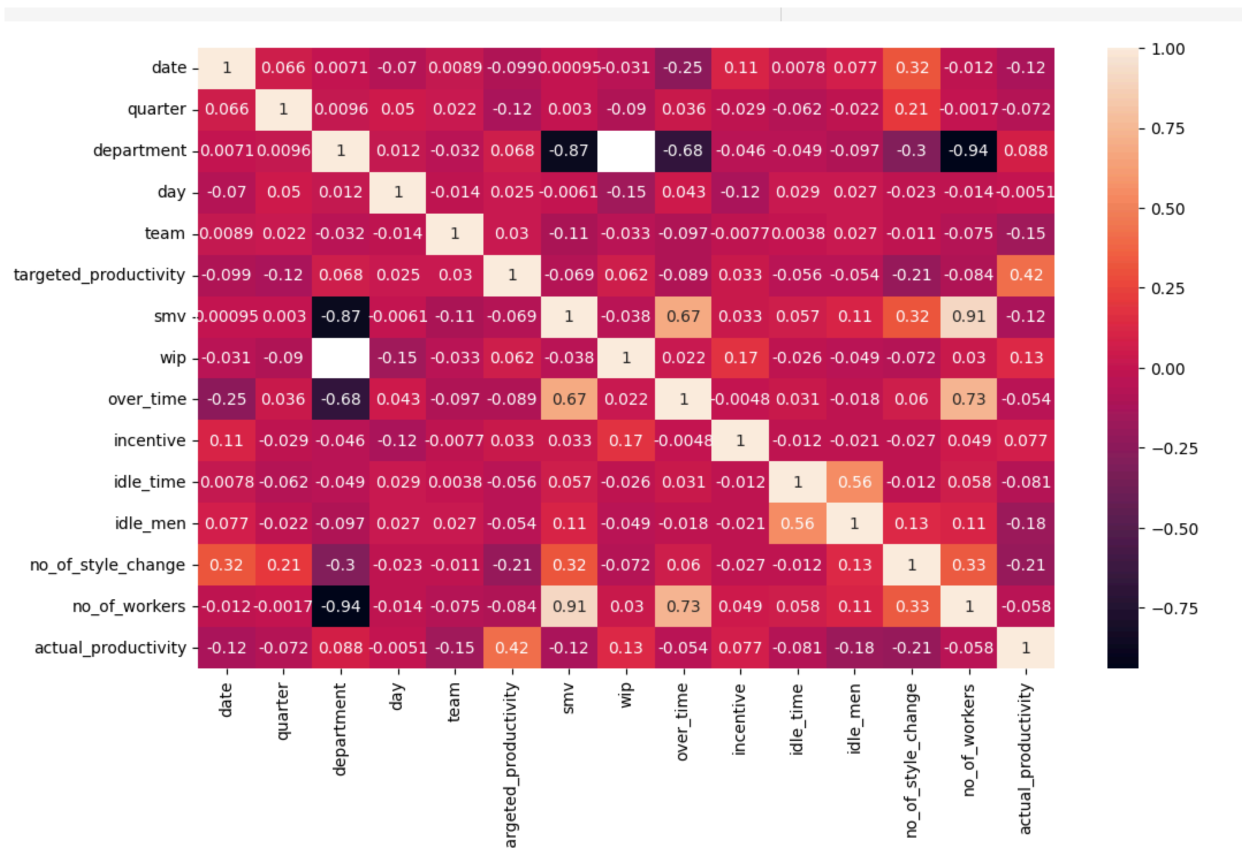
```
[19] # pair plot showing correlation between the variables

     import matplotlib.pyplot as plt
     sns.pairplot(pdt_clean)
     plt.show()
```



- From the above scatter plot representation, it's not very clear how most of the variables are correlated with one another. But we can take a closer and more elaborated view at a heat map to show the correlations more clearly.

Heat map showing correlations between variables



From the above representation it is evident that

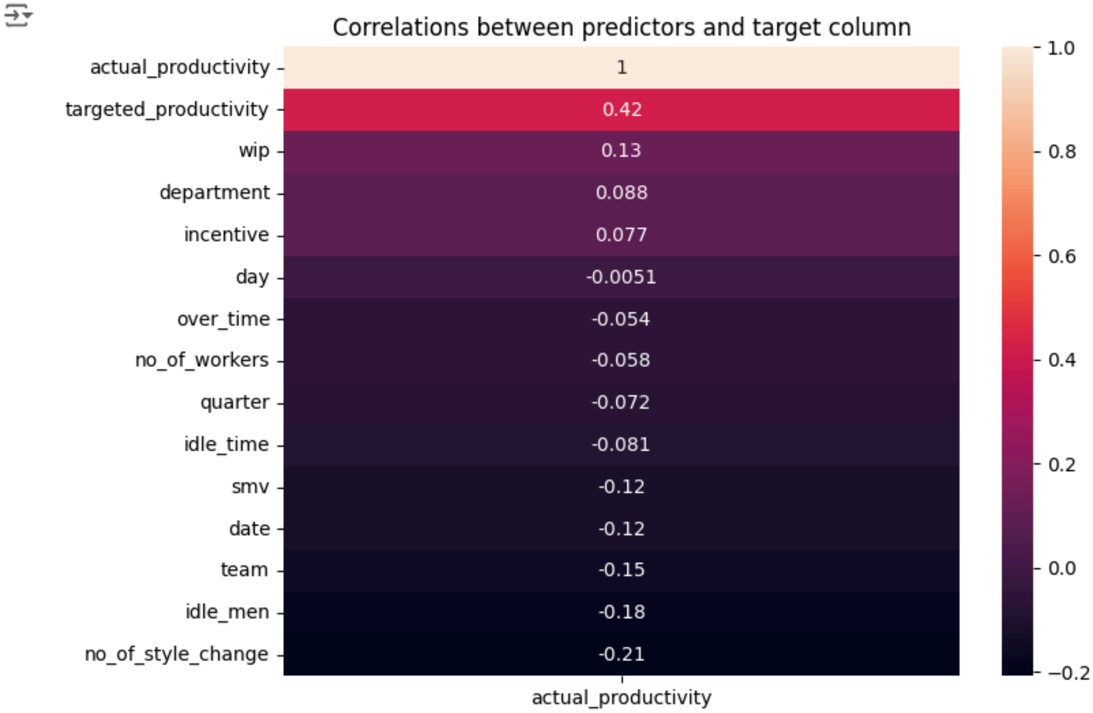
- There is a significantly high positive correlation between no_of_workers and smv (standard minute value) of 0.91, no_of_workers and overtime of 0.73 and fairly positive correlation of 0.67 between the over_time and smv
- There is a weak correlation between no_of_style_change and smv which is 0.32, no_of_workers and no_of_style which is 0.33.
- There’s a strong negative correlation between team and smv of -0.11

Investigating correlation between all the variables and actual productivity

```
[15] #checking the correlation between all the variables and the target variable i.e actual productivity
     corr = pdt_clean_num.corr()
     target_col=corr[['actual_productivity']].sort_values(by='actual_productivity', ascending=False)

     plt.figure(figsize=(8,6))

     sns.heatmap(target_col,annot=True)
     plt.title("Correlations between predictors and target column")
     plt.show()
```



- From the above heat map its shown that target productivity has a fairly positive correlation with the target (actual productivity)

Conclusion

The dataset's purpose is to find out the productivity of workers employed at a garment factory through the use of different variables such as department, team, day overtime among others. From the above analysis it is evident that the data was not well represented due to data entry errors such as misspelling, extra white spaces, missing values in some columns, however, an in depth data wrangling gave room to fix some of the problems in the dataset. The correlation between the different variables is not very clear judging from the scatter plot but it is not easy to draw conclusions just based on just the graphs because weak or no correlations do not always imply irrelevance of variables. Heat maps show that that there is a strong correlation between some variables such as number of workers and smv (standard minute value) , no_of_workers and overtime, one has to go deeper and calculate correlation coefficients, represent the data in other formats, draw lines of best fit etc to draw conclusions. A detailed review of the Exploratory data analysis is shown in the notebook link shared.