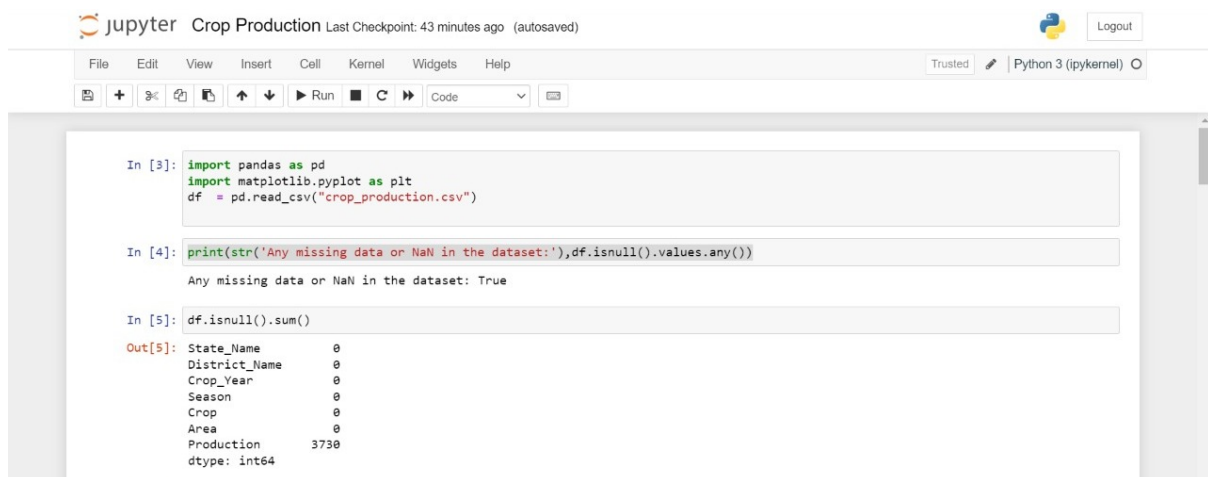


## SPRINT 2

### ESTIMATE THE CROP YIELD USING DATA ANALYTICS

Team ID :PNT2022TMID23348

#### Detecting and Missing Values



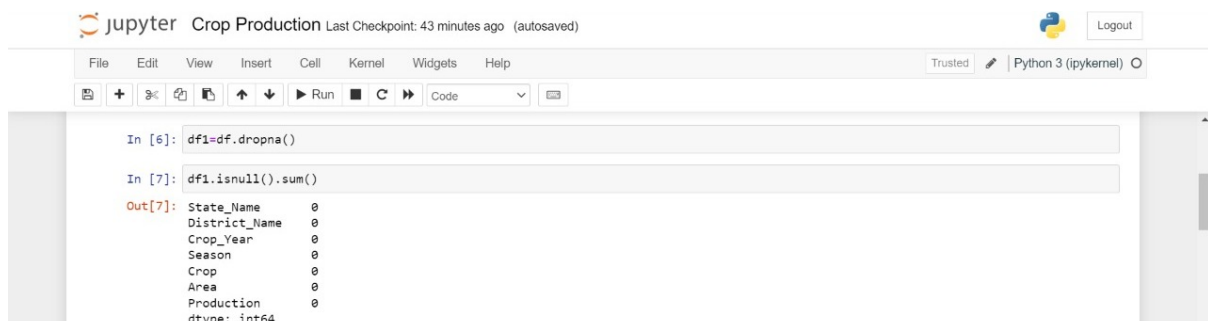
A Jupyter Notebook interface titled "Crop Production" with a last checkpoint of 43 minutes ago. The notebook shows three input cells and one output cell. The first cell imports pandas and matplotlib, and reads a CSV file named "crop\_production.csv". The second cell prints a message and checks for any missing data using `df.isnull().values.any()`. The third cell sums the missing values for each column using `df.isnull().sum()`. The output shows that only the "Production" column has missing values, with a count of 3730.

```
In [3]: import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv("crop_production.csv")

In [4]: print(str('Any missing data or NaN in the dataset:'),df.isnull().values.any())
Any missing data or NaN in the dataset: True

In [5]: df.isnull().sum()
Out[5]: State_Name      0
District_Name    0
Crop_Year        0
Season           0
Crop             0
Area             0
Production      3730
dtype: int64
```

#### Dropping Missing Values



A Jupyter Notebook interface titled "Crop Production" with a last checkpoint of 43 minutes ago. The notebook shows two input cells and one output cell. The first cell drops rows with missing values using `df1=df.dropna()`. The second cell sums the missing values for each column using `df1.isnull().sum()`. The output shows that all columns now have zero missing values.

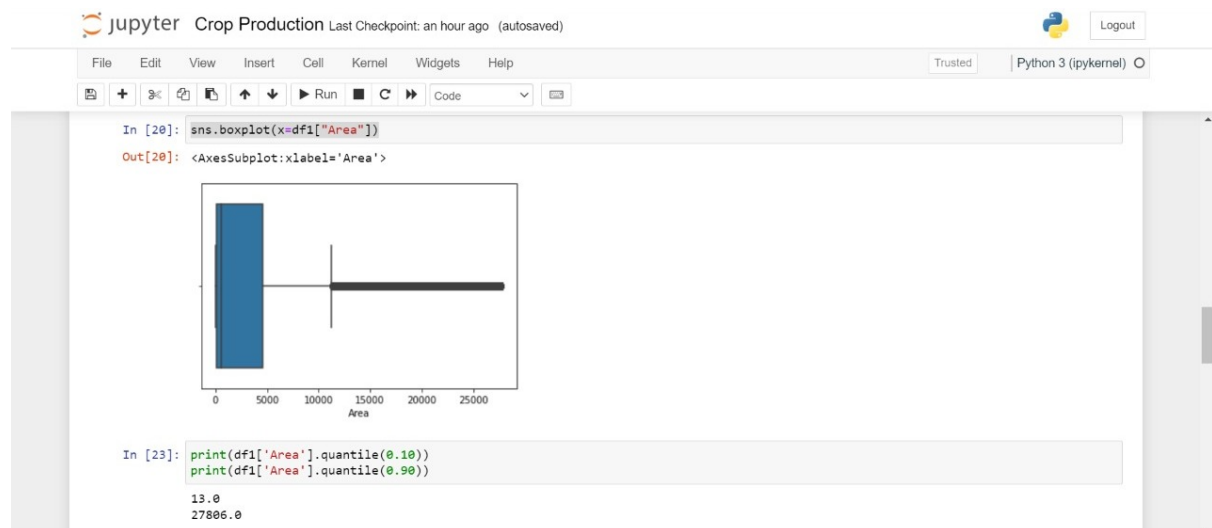
```
In [6]: df1=df.dropna()

In [7]: df1.isnull().sum()
Out[7]: State_Name      0
District_Name    0
Crop_Year        0
Season           0
Crop             0
Area             0
Production       0
dtype: int64
```

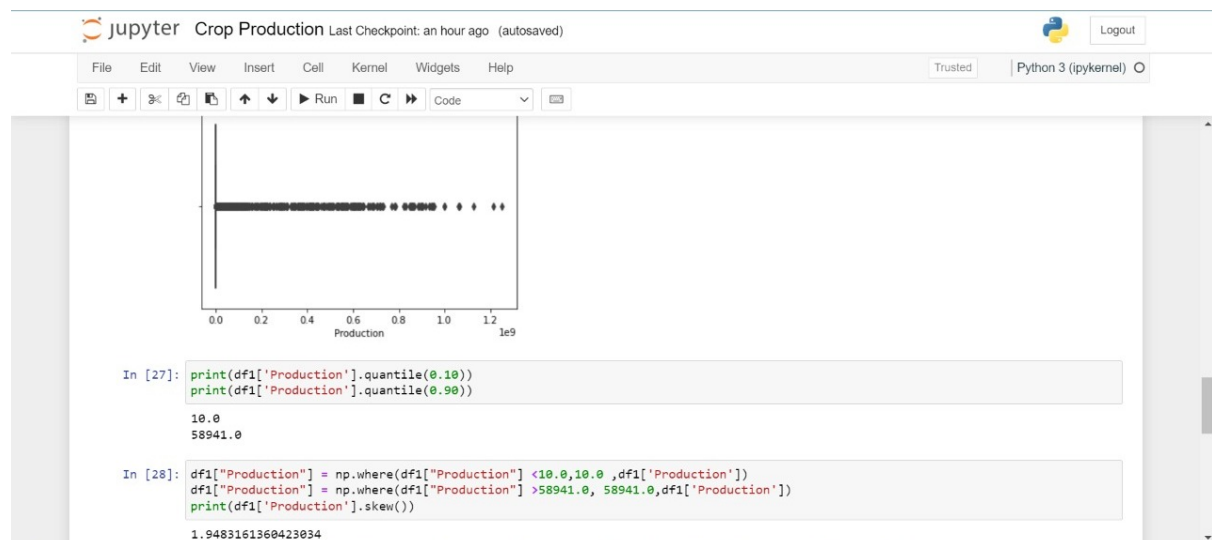
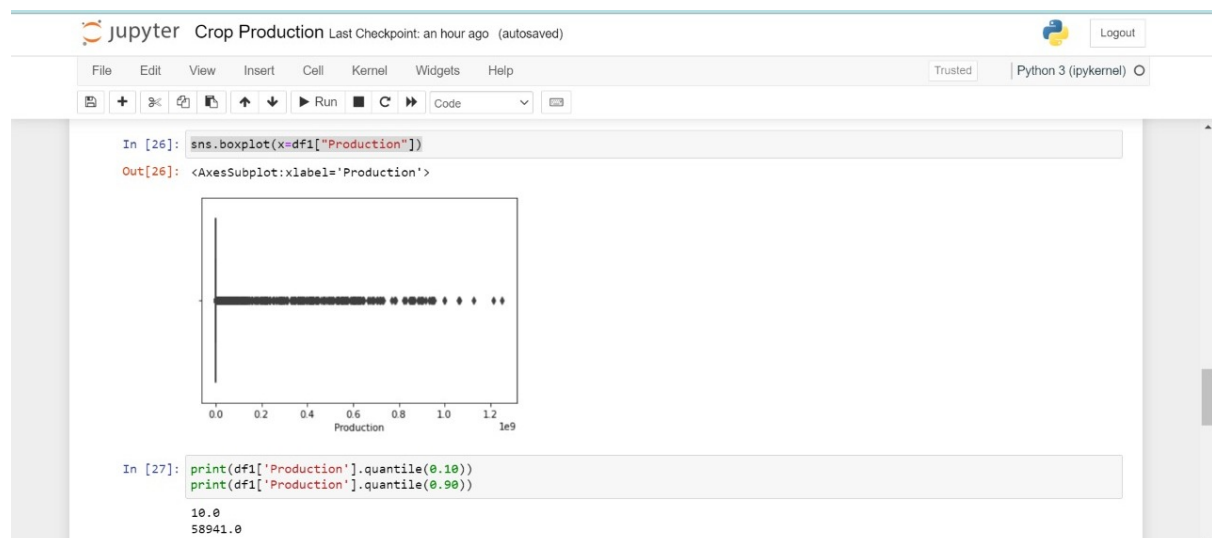
# Detecting and Removing Outliers

## For Area





## For Production



```
In [29]: sns.boxplot(x=df1["Production"])
```

```
Out[29]: <AxesSubplot:xlabel='Production'>
```

