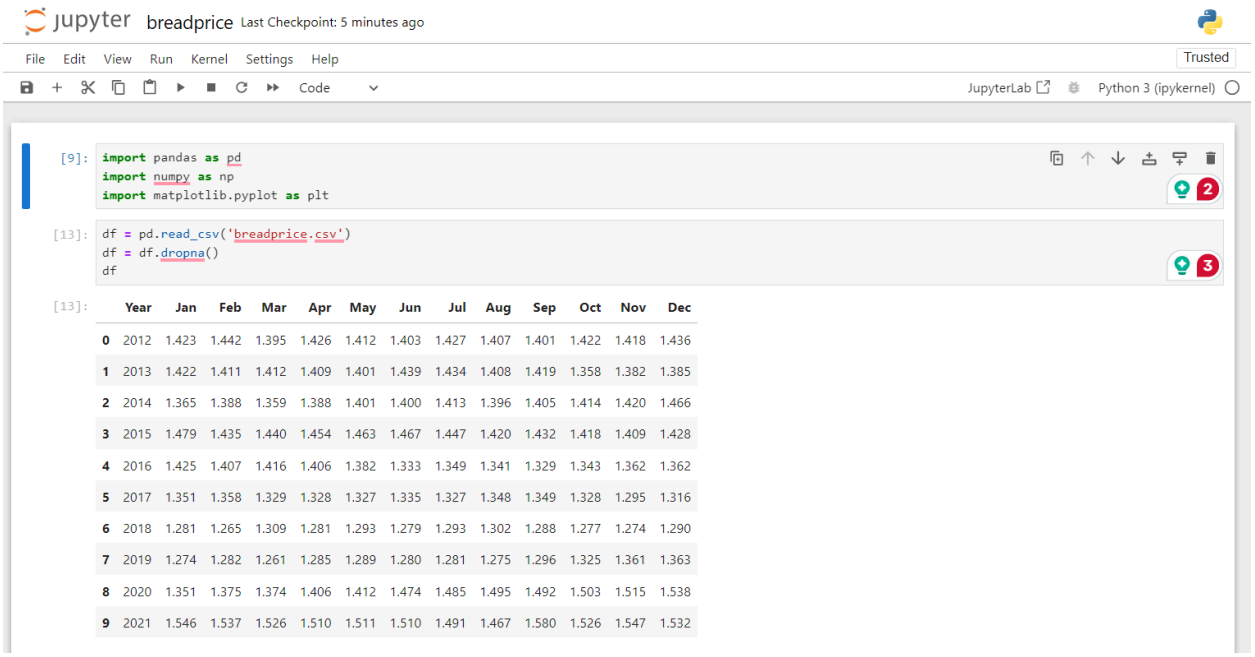
PostLab

1.



	Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
0	2012	1.423	1.442	1.395	1.426	1.412	1.403	1.427	1.407	1.401	1.422	1.418	1.436
1	2013	1.422	1.411	1.412	1.409	1.401	1.439	1.434	1.408	1.419	1.358	1.382	1.385
2	2014	1.365	1.388	1.359	1.388	1.401	1.400	1.413	1.396	1.405	1.414	1.420	1.466
3	2015	1.479	1.435	1.440	1.454	1.463	1.467	1.447	1.420	1.432	1.418	1.409	1.428
4	2016	1.425	1.407	1.416	1.406	1.382	1.333	1.349	1.341	1.329	1.343	1.362	1.362
5	2017	1.351	1.358	1.329	1.328	1.327	1.335	1.327	1.348	1.349	1.328	1.295	1.316
6	2018	1.281	1.265	1.309	1.281	1.293	1.279	1.293	1.302	1.288	1.277	1.274	1.290
7	2019	1.274	1.282	1.261	1.285	1.289	1.280	1.281	1.275	1.296	1.325	1.361	1.363
8	2020	1.351	1.375	1.374	1.406	1.412	1.474	1.485	1.495	1.492	1.503	1.515	1.538
9	2021	1.546	1.537	1.526	1.510	1.511	1.510	1.491	1.467	1.580	1.526	1.547	1.532

Fig 1: loading dataset: breadprice.csv

In figure 1 we use jupyter notebook for us to perform the postlab exercise, for this part we input command such as: import pandas as pd and import numpy as np for us to load the given dataset we have. Then we use this syntax to load and clean our data :

df = pd.read_csv('breadprice.csv')

df = df.dropna()

df

```
[15]: # Calculate the average price for each year
df['Average Price'] = df.mean(axis=1)  # Calculate mean along rows
# Extract year and average price columns
year_price_df = df[['Year', 'Average Price']]
df
```

[15]:

	Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Average Price
0	2012	1.423	1.442	1.395	1.426	1.412	1.403	1.427	1.407	1.401	1.422	1.418	1.436	156.077846
1	2013	1.422	1.411	1.412	1.409	1.401	1.439	1.434	1.408	1.419	1.358	1.382	1.385	156.144615
2	2014	1.365	1.388	1.359	1.388	1.401	1.400	1.413	1.396	1.405	1.414	1.420	1.466	156.216538
3	2015	1.479	1.435	1.440	1.454	1.463	1.467	1.447	1.420	1.432	1.418	1.409	1.428	156.330154
4	2016	1.425	1.407	1.416	1.406	1.382	1.333	1.349	1.341	1.329	1.343	1.362	1.362	156.342692
5	2017	1.351	1.358	1.329	1.328	1.327	1.335	1.327	1.348	1.349	1.328	1.295	1.316	156.383923
6	2018	1.281	1.265	1.309	1.281	1.293	1.279	1.293	1.302	1.288	1.277	1.274	1.290	156.417846
7	2019	1.274	1.282	1.261	1.285	1.289	1.280	1.281	1.275	1.296	1.325	1.361	1.363	156.505538
8	2020	1.351	1.375	1.374	1.406	1.412	1.474	1.485	1.495	1.492	1.503	1.515	1.538	156.724615
9	2021	1.546	1.537	1.526	1.510	1.511	1.510	1.491	1.467	1.580	1.526	1.547	1.532	156.867923

Fig 2: calculating average price for each year

In figure 2, it shows the syntax allows to calculate mean along rows: df['Average Price'] = df.mean(axis=1) for us to determine the calculations of average price for each year.

```
[12]: # Plotting
plt.figure(figsize=(10, 6))
plt.plot(year_price_df['Year'], year_price_df['Average Price'], marker='o')
plt.title('Average Price of Bread Each Year')
plt.xlabel('Year')
plt.ylabel('Average Price')
plt.grid(True)
plt.xticks(year_price_df['Year'])
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

Fig 3: code for plotting

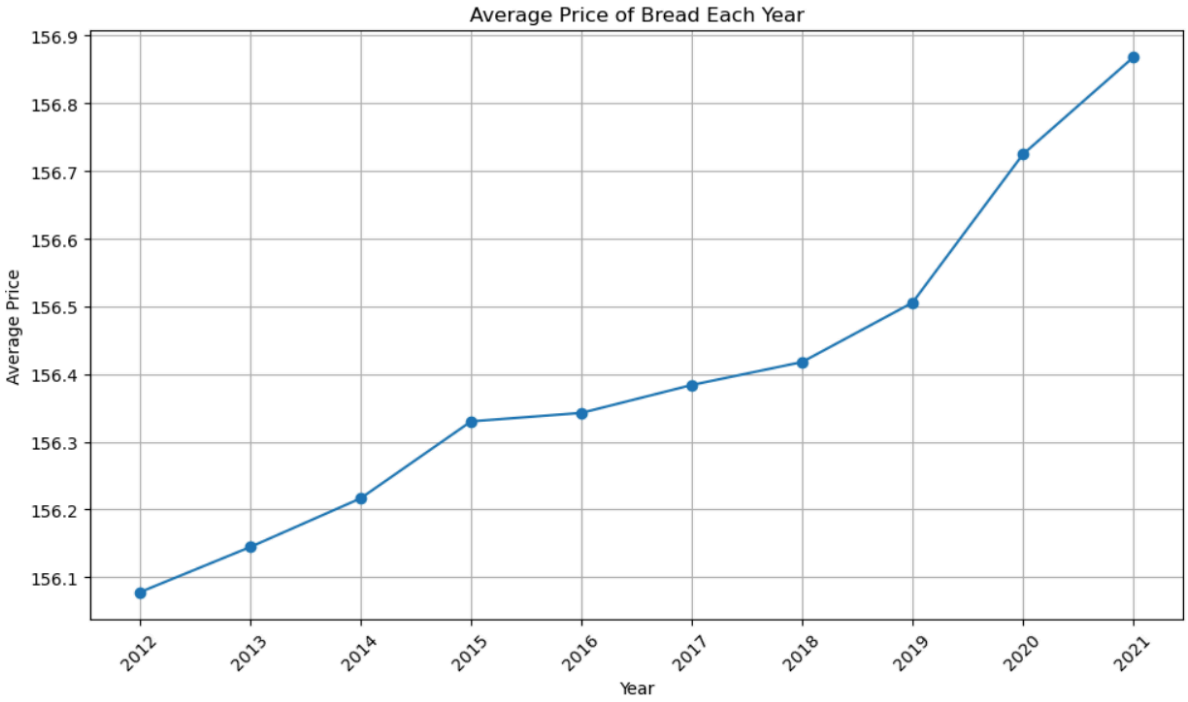


Fig 3.1: Line plot of the average price for each year

For figure 3 and 3.1, it shows the code and the line plot that allows us to shows the line plot of the average price for each year.