Chapter 2

April 20, 2025

1 Hands-On Data Preprocessing in Python

Learn how to effectively prepare data for successful data analytics

`AUTHOR: Dr. Roy Jafari`

1.0.1 Chapter 2: Review of another core module: Matplotlib

[1]:

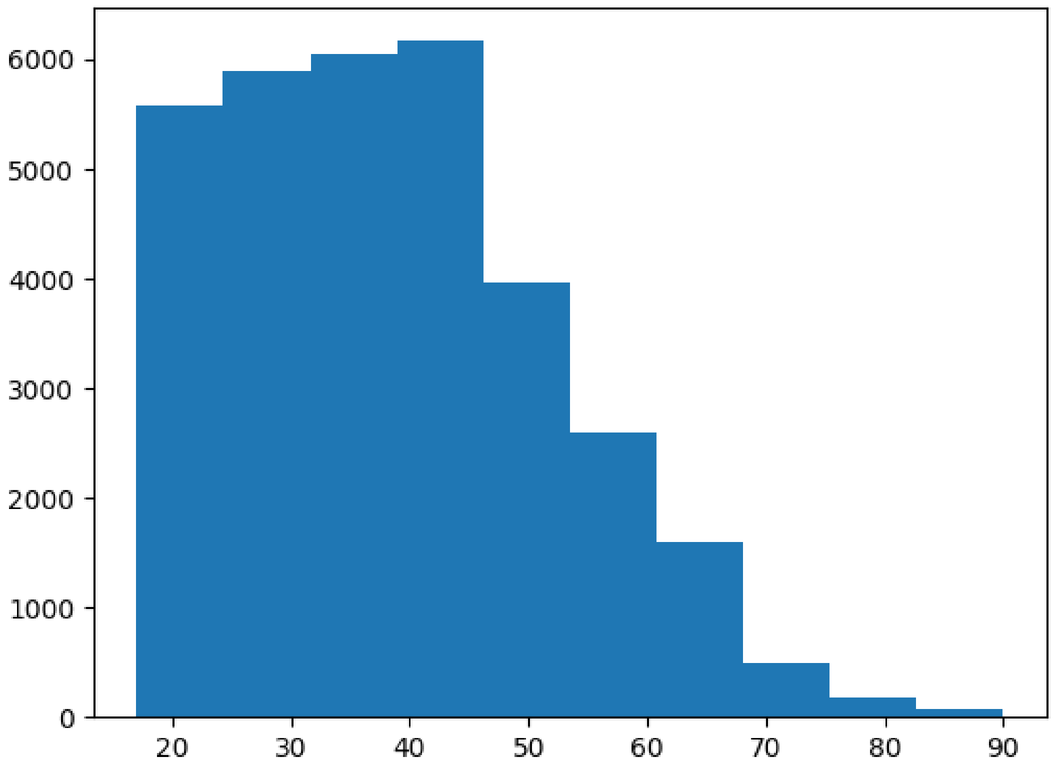
```
#from previous chapter

import pandas as pd
import numpy as np
adult_df = pd.read_csv('adult.csv')
```

[2]:

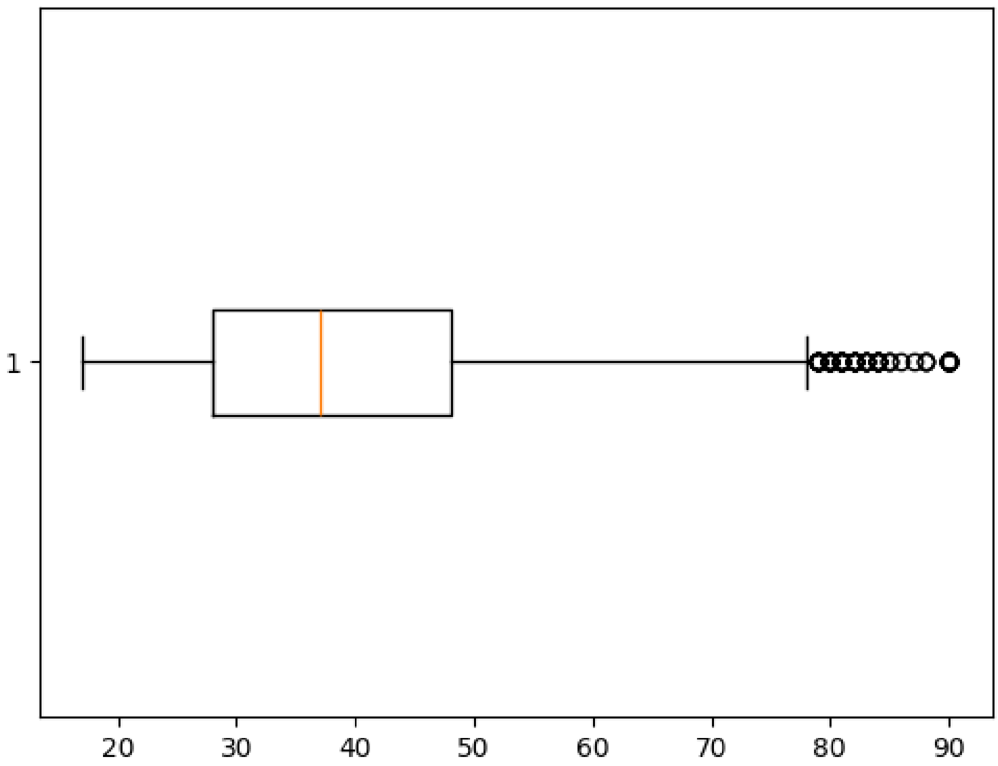
```
import matplotlib.pyplot as plt
```

[3]:

```
plt.hist(adult_df.age)
plt.show()
```



[4]:
```
plt.boxplot(adult_df.age, vert=False)
plt.show()
```

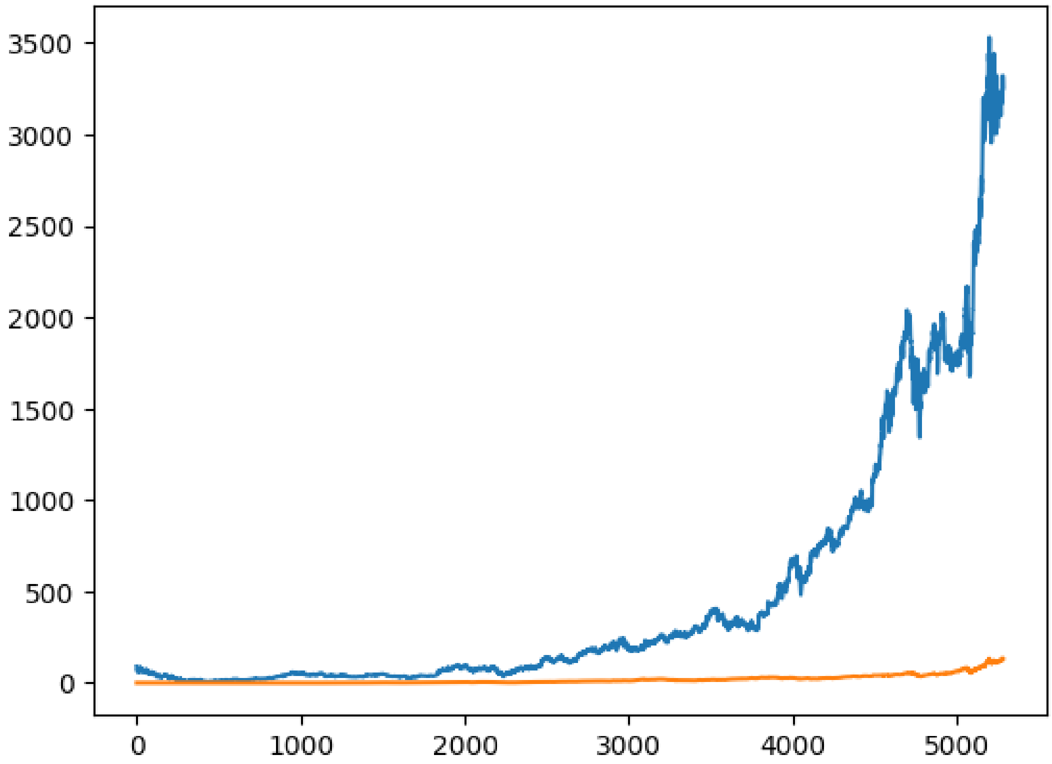


[5]:

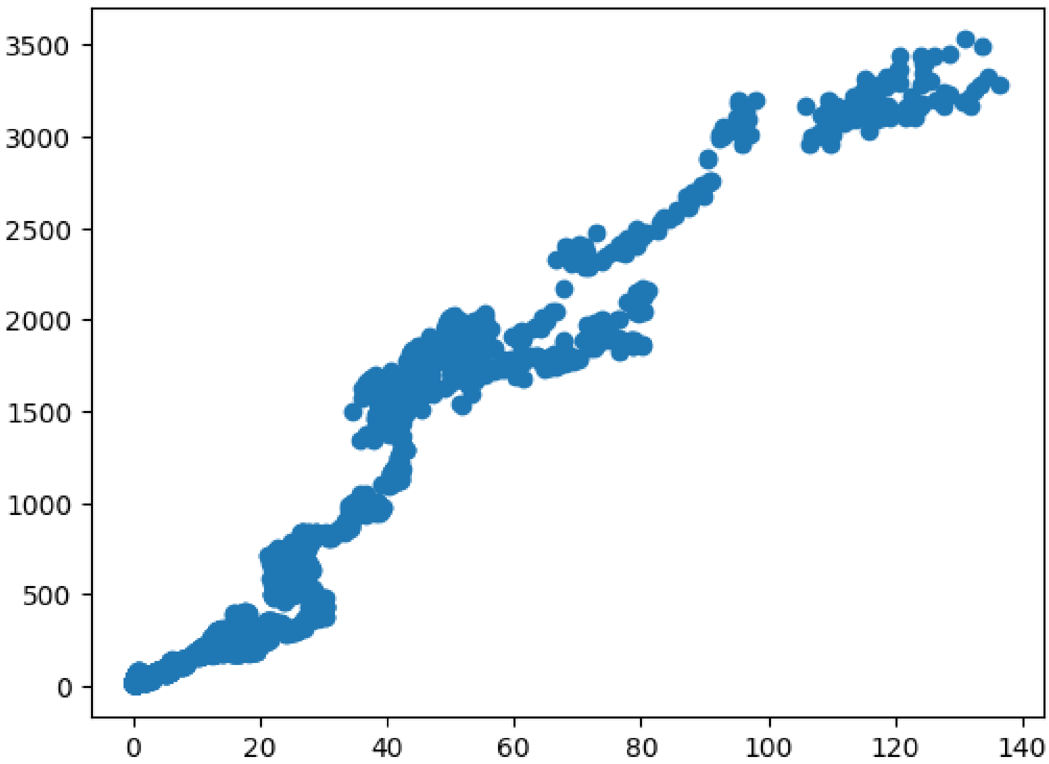
```
amz_df = pd.read_csv('Amazon Stock.csv')
apl_df = pd.read_csv('Apple Stock.csv')
```

[6]:

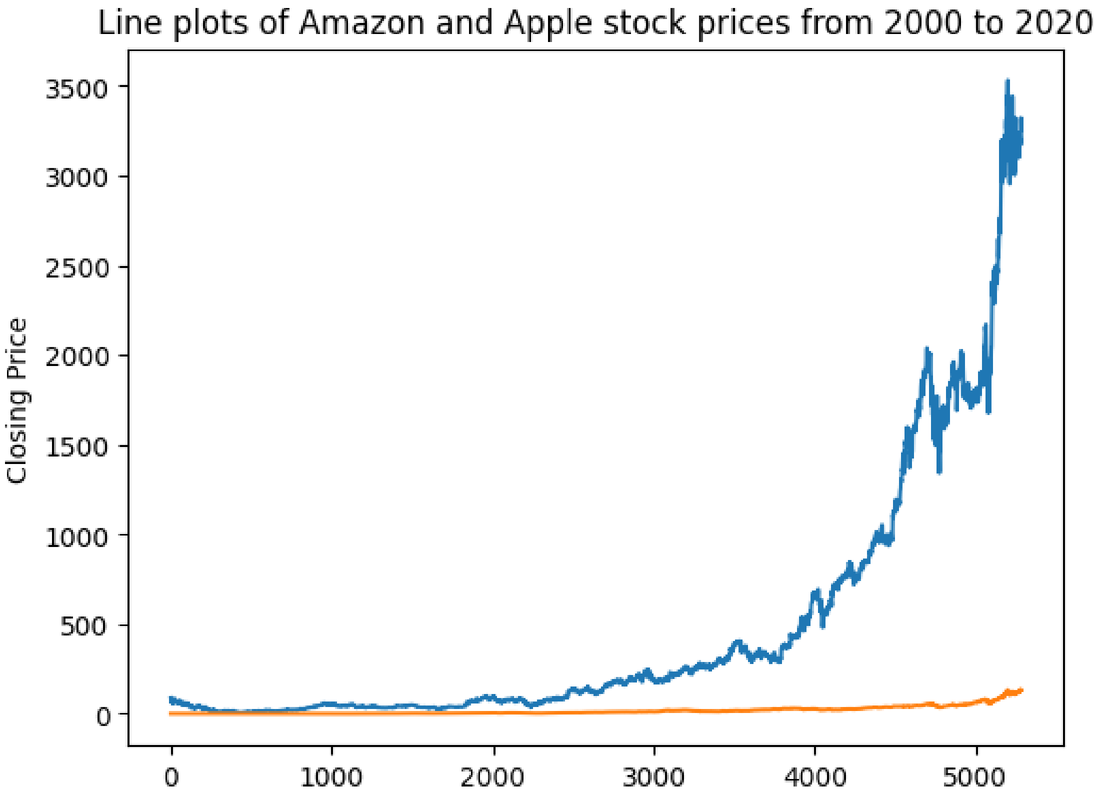
```
plt.plot(amz_df.Close)
plt.plot(apl_df.Close)
plt.show()
```



[7]:
```
plt.scatter(apl_df.Close,amz_df.Close)
plt.show()
```

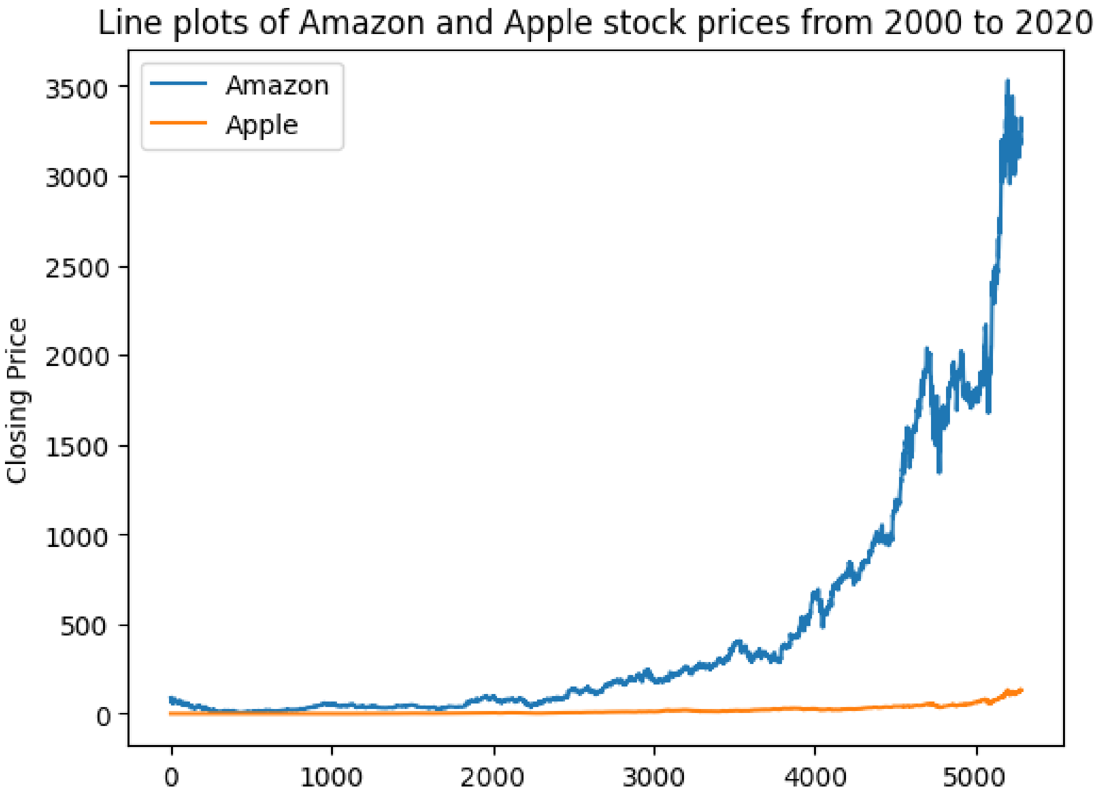


[8]:
```
plt.plot(amz_df.Close)
plt.plot(apl_df.Close)
plt.title('Line plots of Amazon and Apple stock prices from 2000 to 2020')
plt.ylabel('Closing Price')
plt.show()
```

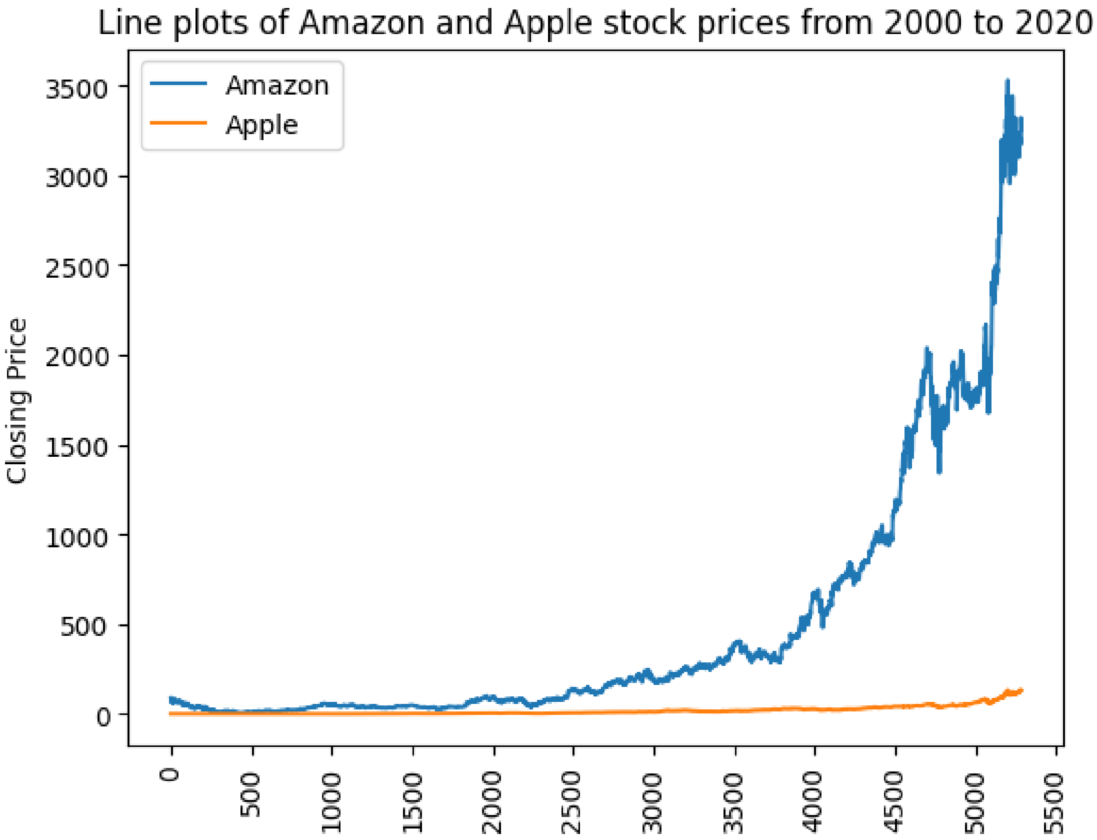


[9]:

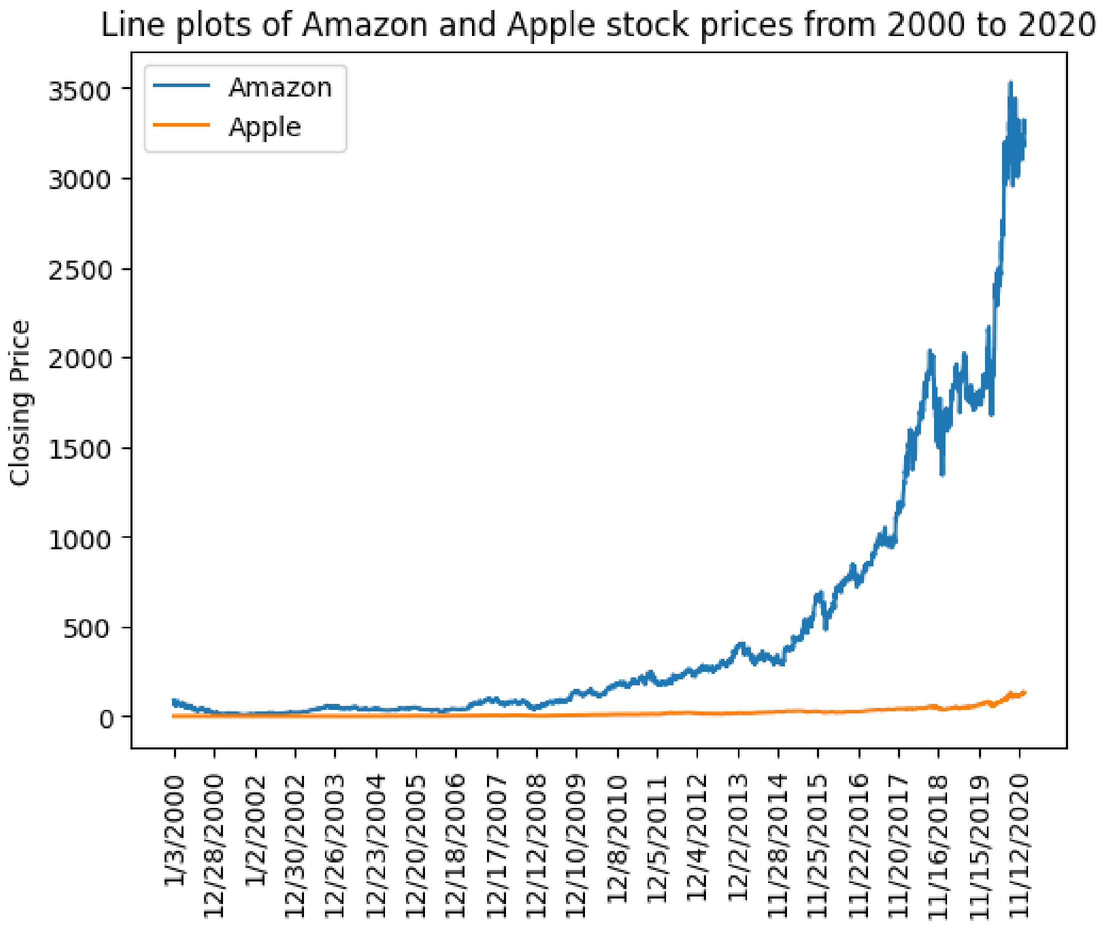
```
plt.plot(amz_df.Close, label='Amazon')
plt.plot(apl_df.Close, label='Apple')
plt.title('Line plots of Amazon and Apple stock prices from 2000 to 2020')
plt.ylabel('Closing Price')
plt.legend()
plt.show()
```



[10]:
```
plt.plot(amz_df.Close, label='Amazon')
plt.plot(apl_df.Close, label='Apple')
plt.title('Line plots of Amazon and Apple stock prices from 2000 to 2020')
plt.ylabel('Closing Price')
plt.xticks([0,500,1000,1500,2000,2500,3000,3500,4000,4500,5000,5500],
        rotation=90)
plt.legend()
plt.show()
```

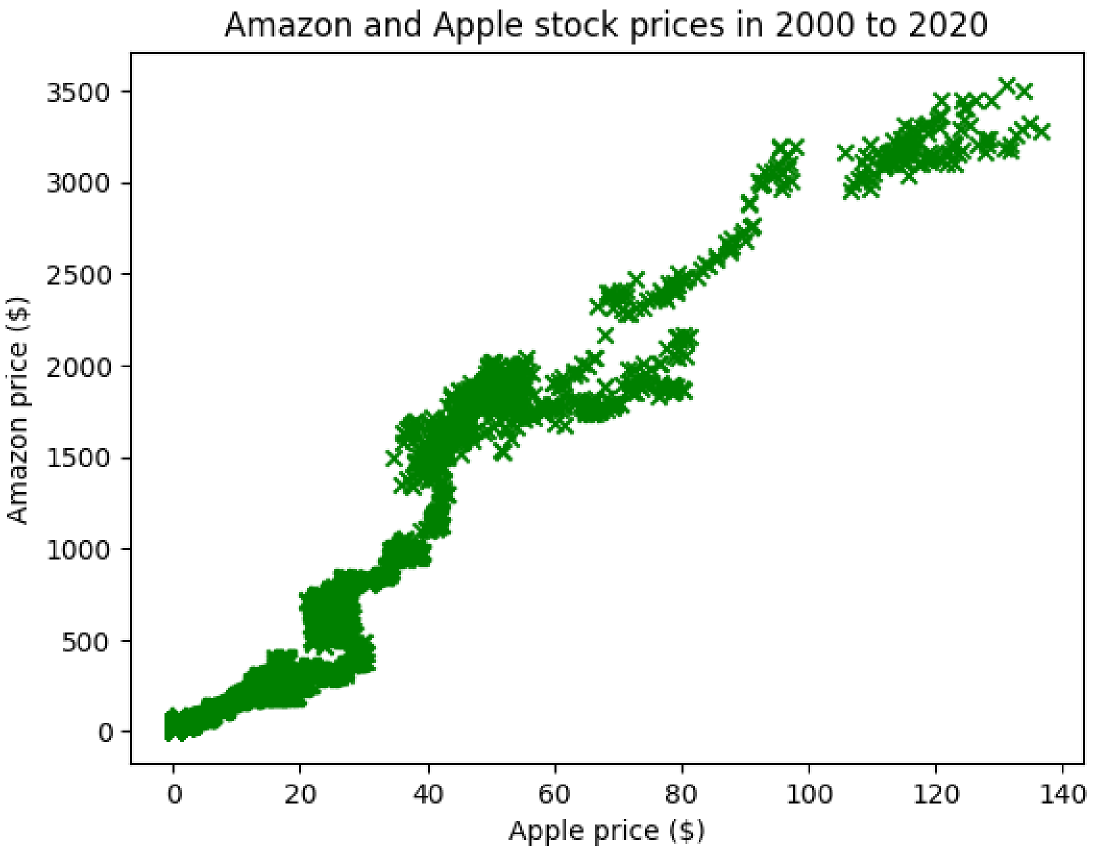


```
[11]: plt.plot(amz_df.Close, label='Amazon')
      plt.plot(apl_df.Close, label='Apple')
      plt.title('Line plots of Amazon and Apple stock prices from 2000 to 2020')
      plt.ylabel('Closing Price')
      plt.legend()
      plt.xticks(np.arange(0,len(amz_df),250),amz_df.Date[0:len(amz_df):250],
              rotation=90)
      plt.show()
```

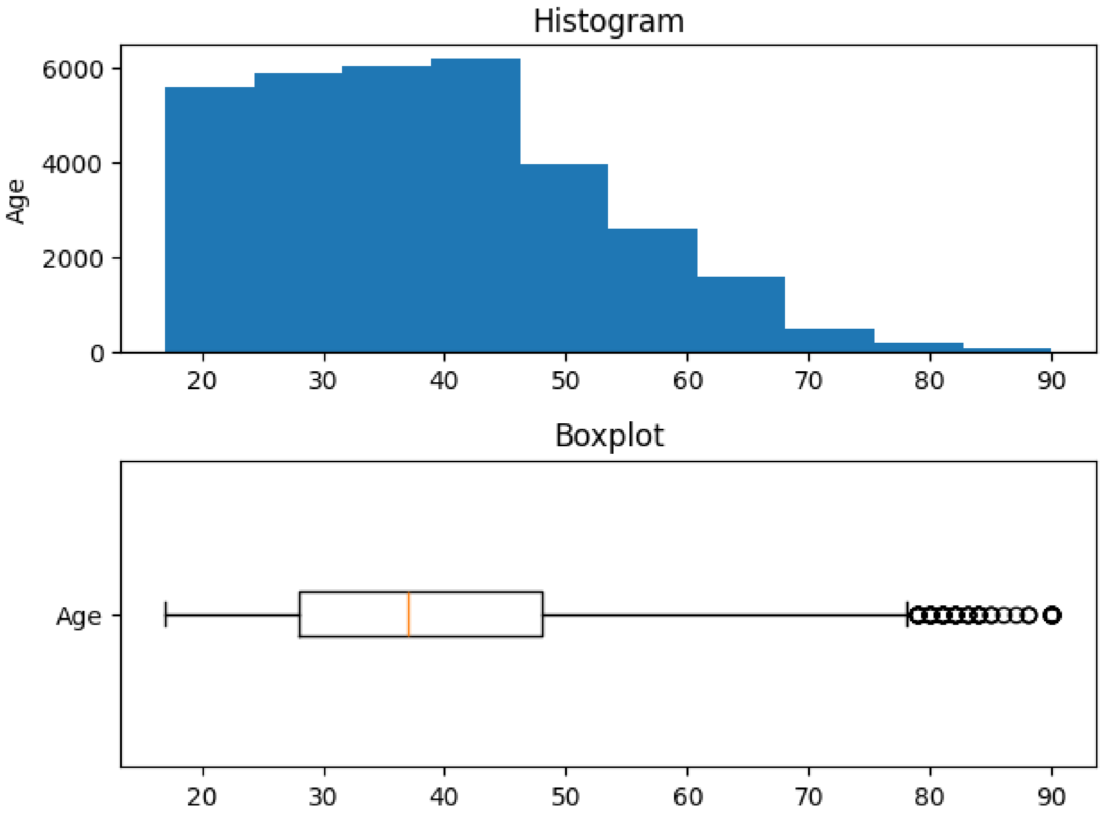


[12]:

```
plt.scatter(apl_df.Close,amz_df.Close, marker = 'x', color='green')
plt.title('Amazon and Apple stock prices in 2000 to 2020')
plt.xlabel('Apple price ($)')
plt.ylabel('Amazon price ($)')
plt.show()
```

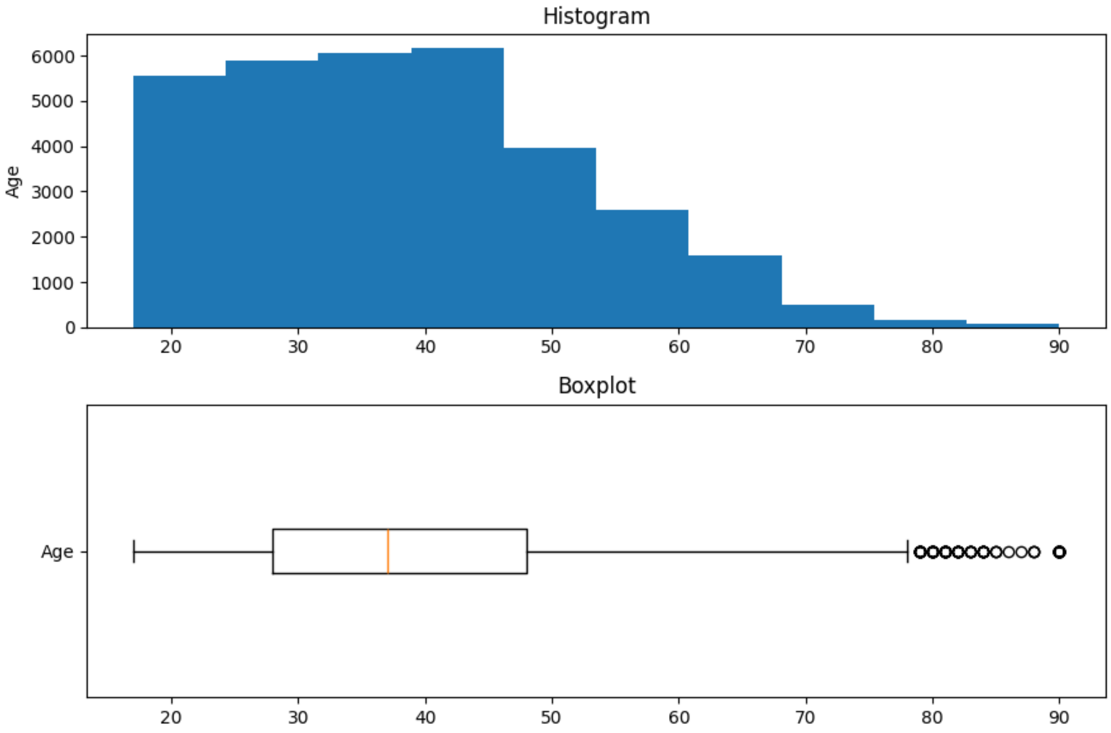


[14]:
```
plt.subplot(2,1,1)
plt.hist(adult_df.age)
plt.title('Histogram')
plt.ylabel('Age')

plt.subplot(2,1,2)
plt.boxplot(adult_df.age, vert=False)
plt.title('Boxplot')
plt.yticks([1],['Age'])

plt.tight_layout()
plt.show()
```

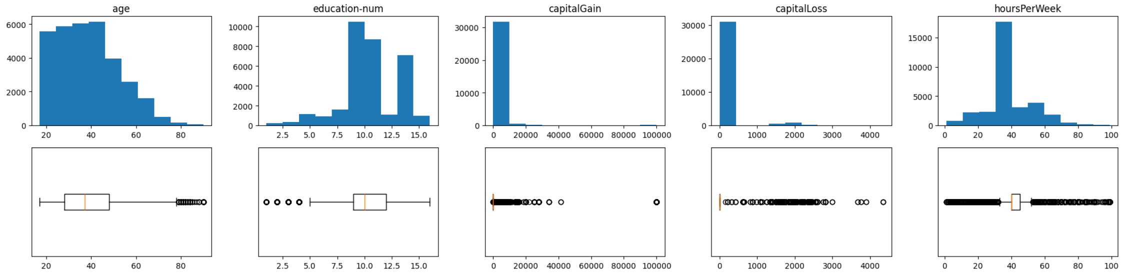


[15]:

```
plt.figure(figsize=(9,6))

plt.subplot(2,1,1)
plt.hist(adult_df.age)
plt.title('Histogram')
plt.ylabel('Age')

plt.subplot(2,1,2)
plt.boxplot(adult_df.age, vert=False)
plt.title('Boxplot')
plt.yticks([1],['Age'])

plt.tight_layout()
plt.show()
```



[18]:
```
Numerical_colums = ['age', 'education-num', 'capitalGain', 'capitalLoss',␣
 ↪'hoursPerWeek']

plt.figure(figsize=(20,5))

for i,col in enumerate(Numerical_colums):
    plt.subplot(2,5,i+1)
    plt.hist(adult_df[col])
    plt.title(col)

for i,col in enumerate(Numerical_colums):
    plt.subplot(2,5,i+6)
    plt.boxplot(adult_df[col],vert=False)
    plt.yticks([])


plt.tight_layout()
plt.savefig('ColumnsVsiaulization.png', dpi=900)
```


age
0
2000
4000
6000
20
40
60
80
education-num
0
2000
4000
6000
8000
10000
2.5
5.0
7.5
10.0
12.5
15.0
capitalGain
0
10000
20000
30000
0
20000
40000
60000
80000
100000
capitalLoss
0
10000
20000
30000
0
1000
2000
3000
4000
hoursPerWeek
0
5000
10000
15000
0
20
40
60
80
100