Sri R Sankaranarayanan

**DSC640 – Data Visualization**

1. **PowerBI - Line and Step Chart**

**Data 1**

**A screenshot of a computer

Description automatically generated**

**Data 2**

**A picture containing graphical user interface

Description automatically generated**

**Data 3**

**Graphical user interface

Description automatically generated with medium confidence**

1. **Python – Line and Step Chart**

You need to submit 3 line charts and 3 step charts using Tableau or PowerBI, Python and R using the data below (or your own datasets). You can also submit using D3, though not required. You can choose which library to use in Python or R, documentation is provided to help you decide and as you start to play around in the libraries, you will decide which you prefer.

In [1]:

*# Import libraries*

**import** csv

*#import xlrd*

**import** pandas **as** pd

**import** matplotlib.pyplot **as** plt

**from** datetime **import** datetime **as** dt

In [3]:

*# Read world population data*

fileData1 **=** 'world-population.xlsm'

population **=** pd**.**read\_excel(fileData1, sheet\_no **=** 1)

*# Read AEP data*

fileData2 **=** 'AEP\_hourly.csv'

aep **=** pd**.**read\_csv(fileData2)

*# Read sample superstore data*

fileData3 **=** 'Sample - Superstore.xlsx'

superstore **=** pd**.**read\_excel(fileData3)

print(population**.**head())

print(aep**.**head())

print(superstore**.**head())

Year Population

0 1960 3028654024

1 1961 3068356747

2 1962 3121963107

3 1963 3187471383

4 1964 3253112403

Datetime AEP\_MW

0 2004-12-31 01:00:00 13478.0

1 2004-12-31 02:00:00 12865.0

2 2004-12-31 03:00:00 12577.0

3 2004-12-31 04:00:00 12517.0

4 2004-12-31 05:00:00 12670.0

Row ID Order ID Order Date Ship Date Ship Mode Customer ID \

0 1 CA-2016-152156 2016-11-08 2016-11-11 Second Class CG-12520

1 2 CA-2016-152156 2016-11-08 2016-11-11 Second Class CG-12520

2 3 CA-2016-138688 2016-06-12 2016-06-16 Second Class DV-13045

3 4 US-2015-108966 2015-10-11 2015-10-18 Standard Class SO-20335

4 5 US-2015-108966 2015-10-11 2015-10-18 Standard Class SO-20335

Customer Name Segment Country City ... \

0 Claire Gute Consumer United States Henderson ...

1 Claire Gute Consumer United States Henderson ...

2 Darrin Van Huff Corporate United States Los Angeles ...

3 Sean O'Donnell Consumer United States Fort Lauderdale ...

4 Sean O'Donnell Consumer United States Fort Lauderdale ...

Postal Code Region Product ID Category Sub-Category \

0 42420 South FUR-BO-10001798 Furniture Bookcases

1 42420 South FUR-CH-10000454 Furniture Chairs

2 90036 West OFF-LA-10000240 Office Supplies Labels

3 33311 South FUR-TA-10000577 Furniture Tables

4 33311 South OFF-ST-10000760 Office Supplies Storage

Product Name Sales Quantity \

0 Bush Somerset Collection Bookcase 261.9600 2

1 Hon Deluxe Fabric Upholstered Stacking Chairs,... 731.9400 3

2 Self-Adhesive Address Labels for Typewriters b... 14.6200 2

3 Bretford CR4500 Series Slim Rectangular Table 957.5775 5

4 Eldon Fold 'N Roll Cart System 22.3680 2

Discount Profit

0 0.00 41.9136

1 0.00 219.5820

2 0.00 6.8714

3 0.45 -383.0310

4 0.20 2.5164

[5 rows x 21 columns]

**World Population data**

**Line and Step graph**

In [69]:

X **=** population['Year']

Y **=** population['Population']

plt**.**plot(X, Y)

plt**.**show()

**Chart, line chart

Description automatically generated**

In [70]:

plt**.**step(X, Y)

plt**.**show()

**Chart, line chart

Description automatically generated**

**AEP Data**

**Line and Step graph**

For the AEP data, there are too many observations to put in a line plot. Hence I calculated the total of AEP\_MW for each year and plotted them

In [59]:

*# Convert pandas column to datetime*

aep['Datetime'] **=** pd**.**to\_datetime(aep['Datetime'])

*# Extract Year and store in a new column*

aep['Year'] **=** aep['Datetime']**.**dt**.**year

*# Calculate sum of MW for each year*

aep\_aggr **=** aep**.**groupby(['Year'])['AEP\_MW']**.**agg('sum')**.**reset\_index(name**=**'Total\_MW')

aep\_aggr**.**head()

Out[59]:

|  | **Year** | **Total\_MW** |
| --- | --- | --- |
| **0** | 2004 | 33479854.0 |
| **1** | 2005 | 138752914.0 |
| **2** | 2006 | 137826610.0 |
| **3** | 2007 | 145781458.0 |
| **4** | 2008 | 145224910.0 |

In [72]:

X **=** aep\_aggr['Year']

Y **=** aep\_aggr['Total\_MW']

plt**.**plot(X, Y)

plt**.**show()

**Line chart

Description automatically generated with medium confidence**

In [73]:

plt**.**step(X, Y)

plt**.**show()

**A picture containing shape

Description automatically generated**

**Sample Superstore data**

**Line and Step graph**

For this data set, I plan to plot the profit by order year and segment. This needs some additional data preparation as below.

In [5]:

*# Extract order year*

superstore['Order Year'] **=** superstore['Order Date']**.**dt**.**year

*# Calculate total profit for each segment each year*

superstore\_aggr **=** superstore**.**groupby(['Order Year'])['Profit']**.**agg('sum')**.**reset\_index(name**=**'Total Profit')

*# Check data*

superstore\_aggr**.**head()

Out[5]:

|  | **Order Year** | **Total Profit** |
| --- | --- | --- |
| **0** | 2014 | 49543.9741 |
| **1** | 2015 | 61618.6037 |
| **2** | 2016 | 81795.1743 |
| **3** | 2017 | 93439.2696 |

In [7]:

X **=** superstore\_aggr['Order Year']

Y **=** superstore\_aggr['Total Profit']

plt**.**plot(X, Y)

plt**.**show()

**Chart, line chart

Description automatically generated**

In [8]:

plt**.**step(X, Y)

plt**.**show()

**Shape

Description automatically generated**

In [ ]:

1. **R – Line and Step Chart**

*# Import required packages*

library('magrittr')

​

*# Import data to be used for visualization*

fileData1 **=** paste(getwd(), '/world-population.xlsm', sep **=** '')

population **=** xlsx::read.xlsx(fileData1, sheetIndex **=** 1, stringsAsFactors **=** FALSE)

​

fileData2 **=** paste(getwd(), '/AEP\_hourly.csv', sep **=** '')

aep **=** read.csv2(fileData2, sep**=**',', stringsAsFactors **=** FALSE) **%>%**

**as**.data.frame()

​

fileData3 **=** paste(getwd(), '/Sample - Superstore.xlsx', sep **=** '')

superstore **=** xlsx::read.xlsx(fileData3, sheetIndex **=** 1, stringsAsFactors **=** FALSE)

​

*# Examine data*

print(head(population))

print(head(aep))

print(head(superstore))

Year Population

1 1960 3028654024

2 1961 3068356747

3 1962 3121963107

4 1963 3187471383

5 1964 3253112403

6 1965 3320396924

Datetime AEP\_MW

1 2004-12-31 01:00:00 13478.0

2 2004-12-31 02:00:00 12865.0

3 2004-12-31 03:00:00 12577.0

4 2004-12-31 04:00:00 12517.0

5 2004-12-31 05:00:00 12670.0

6 2004-12-31 06:00:00 13038.0

Row.ID Order.ID Order.Date Ship.Date Ship.Mode Customer.ID

1 1 CA-2016-152156 2016-11-08 2016-11-11 Second Class CG-12520

2 2 CA-2016-152156 2016-11-08 2016-11-11 Second Class CG-12520

3 3 CA-2016-138688 2016-06-12 2016-06-16 Second Class DV-13045

4 4 US-2015-108966 2015-10-11 2015-10-18 Standard Class SO-20335

5 5 US-2015-108966 2015-10-11 2015-10-18 Standard Class SO-20335

6 6 CA-2014-115812 2014-06-09 2014-06-14 Standard Class BH-11710

Customer.Name Segment Country City State

1 Claire Gute Consumer United States Henderson Kentucky

2 Claire Gute Consumer United States Henderson Kentucky

3 Darrin Van Huff Corporate United States Los Angeles California

4 Sean O'Donnell Consumer United States Fort Lauderdale Florida

5 Sean O'Donnell Consumer United States Fort Lauderdale Florida

6 Brosina Hoffman Consumer United States Los Angeles California

Postal.Code Region Product.ID Category Sub.Category

1 42420 South FUR-BO-10001798 Furniture Bookcases

2 42420 South FUR-CH-10000454 Furniture Chairs

3 90036 West OFF-LA-10000240 Office Supplies Labels

4 33311 South FUR-TA-10000577 Furniture Tables

5 33311 South OFF-ST-10000760 Office Supplies Storage

6 90032 West FUR-FU-10001487 Furniture Furnishings

Product.Name Sales

1 Bush Somerset Collection Bookcase 261.9600

2 Hon Deluxe Fabric Upholstered Stacking Chairs, Rounded Back 731.9400

3 Self-Adhesive Address Labels for Typewriters by Universal 14.6200

4 Bretford CR4500 Series Slim Rectangular Table 957.5775

5 Eldon Fold 'N Roll Cart System 22.3680

6 Eldon Expressions Wood and Plastic Desk Accessories, Cherry Wood 48.8600

Quantity Discount Profit

1 2 0.00 41.9136

2 3 0.00 219.5820

3 2 0.00 6.8714

4 5 0.45 -383.0310

5 2 0.20 2.5164

6 7 0.00 14.1694

**World Population data**

**Line and Step graph**

In [2]:



options(repr.plot.width **=** 4, repr.plot.height **=** 3)

​

ggplot2::ggplot(data**=**population, ggplot2::aes(x**=**Year, y**=**Population)) **+**

ggplot2::geom\_line(linetype**=**'solid', color**=**'blue', size**=**1.2)

**Chart, line chart

Description automatically generated**

In [3]:



ggplot2::ggplot(data**=**population, ggplot2::aes(x**=**Year, y**=**Population)) **+**

ggplot2::geom\_step(linetype**=**'solid', color**=**'blue', size**=**1.2)

**Chart, line chart

Description automatically generated**

**AEP Data**

**Line and Step graph**

For the AEP data, there are too many observations to put in a line plot. Hence I calculated the total of AEP\_MW for each year and plotted them

In [4]:



*# Prepare dataset*

aep\_aggr **=** aep **%>%**

dplyr::mutate(Year **=** **as**.numeric(format(**as**.Date(Datetime), '%Y')),

AEP\_MW **=** **as**.numeric(AEP\_MW)) **%>%**

dplyr::group\_by(Year) **%>%**

dplyr::summarise(Total\_MW **=** sum(AEP\_MW))

In [5]:



options(repr.plot.width **=** 16, repr.plot.height **=** 7)

​

ggplot2::ggplot(data**=**aep\_aggr, ggplot2::aes(x**=**Year, y**=**Total\_MW)) **+**

ggplot2::geom\_line(linetype**=**'solid', color**=**'blue', size**=**1.2)

**Chart, line chart

Description automatically generated**

In [6]:



ggplot2::ggplot(data**=**aep\_aggr, ggplot2::aes(x**=**Year, y**=**Total\_MW)) **+**

ggplot2::geom\_step(linetype**=**'solid', color**=**'blue', size**=**1.2)

**Chart, line chart

Description automatically generated**

**Sample Superstore data**

**Line and Step graph**

For this data set, I plan to plot the profit by order year and segment. This needs some additional data preparation as below.

In [7]:



superstorePrep **=** superstore **%>%**

dplyr::mutate(OrderYear **=** **as**.numeric(format(**as**.Date(Order.Date), '%Y'))) **%>%**

dplyr::select(OrderYear, Segment, Profit) **%>%**

dplyr::group\_by(OrderYear, Segment) **%>%**

dplyr::summarise(TotalProfit **=** sum(Profit))

In [8]:



ggplot2::ggplot(data**=**superstorePrep, ggplot2::aes(x**=**OrderYear, y**=**TotalProfit, group**=**Segment)) **+**

ggplot2::geom\_line(linetype**=**'solid', size**=**1.2, ggplot2::aes(color**=**Segment)) **+**

ggplot2::scale\_color\_brewer(palette**=**'Dark2') **+**

ggplot2::theme\_minimal() **+**

ggplot2::theme(legend.position**=**'top',

legend.title**=**ggplot2::element\_text(size**=**20),

legend.text**=**ggplot2::element\_text(size**=**18),

axis.text**=**ggplot2::element\_text(size**=**15),

axis.title**=**ggplot2::element\_text(size**=**15)

)

**Chart, line chart

Description automatically generated**

In [9]:



ggplot2::ggplot(data**=**superstorePrep, ggplot2::aes(x**=**OrderYear, y**=**TotalProfit, group**=**Segment)) **+**

ggplot2::geom\_step(linetype**=**'solid', size**=**1.2, ggplot2::aes(color**=**Segment)) **+**

ggplot2::scale\_color\_brewer(palette**=**'Dark2') **+**

ggplot2::theme\_minimal() **+**

ggplot2::theme(legend.position**=**'top',

legend.title**=**ggplot2::element\_text(size**=**20),

legend.text**=**ggplot2::element\_text(size**=**18),

axis.text**=**ggplot2::element\_text(size**=**15),

axis.title**=**ggplot2::element\_text(size**=**15)

)

**Chart, box and whisker chart

Description automatically generated**

**End of code**

Chart, line chart

Description automatically generatedChart, line chart

Description automatically generatedChart, line chart

Description automatically generatedChart, line chart

Description automatically generatedChart, line chart

Description automatically generatedChart, box and whisker chart

Description automatically generated