Data Visualization – Lab 6

Name: Ayush Sharma
Reg. No: 15BCE1335
Faculty: Dr. Priyadarshini J

For the given airport dataset do the following

a. **Pre-Processing: Assign column headers**

Code for airport dataset:

```
import pandas as pd
xl = pd.ExcelFile('Airport dataset.xlsx')
df1 = xl.parse('Sheet1', header=None)
dfx=df1[0].str.split(',',expand=True)
del dfx[14]
del dfx[13]
del dfx[12]
dfx.columns=['id','name','city','country','code','icao','latitude','longitude','altitude','offset','dst','timezone']
dfx
```

Output:

	id	name	city	country	code	icao	latitude	longitude	altitude	offset	dst	timezone
0	1	"Goroka Airport"	"Goroka"	"Papua New Guinea"	"GKA"	"AYGA"	-6.081689834590001	145.391998291	5282	10	"U"	"Pacific/Port_Moresby"
1	2	"Madang Airport"	"Madang"	"Papua New Guinea"	"MAG"	"AYMD"	-5.20707988739	145.789001465	20	10	"U"	"Pacific/Port_Moresby"
2	3	"Mount Hagen Kagamuga Airport"	"Mount Hagen"	"Papua New Guinea"	"HGU"	"AYMH"	-5.826789855957031	144.29600524902344	5388	10	"U"	"Pacific/Port_Moresby"
3	4	"Nadzab Airport"	"Nadzab"	"Papua New Guinea"	"LAE"	"AYNZ"	-6.569803	146.725977	239	10	"U"	"Pacific/Port_Moresby"
4	5	"Port Moresby Jacksons International Airport"	"Port Moresby"	"Papua New Guinea"	"POM"	"AYPY"	-9.443380355834961	147.22000122070312	146	10	"U"	"Pacific/Port_Moresby"

Code for airline dataset:

```
xl2 = pd.ExcelFile('Airline datase.xlsx')
df2 = xl2.parse('Sheet1')
dfx2 = df2['-1,"Unknown",\\N,"-","N/A",\\N,\\N,"Y"'].str.split(',',expand=True)
del dfx2[8]
dfx2.columns=['id','name','alias','iata','icao','callsign','country','active']
dfx2
```

Output:

	id	name	alias	iata	icao	callsign	country	active
0	1	"Private flight"	\N	"-"	"N/A"	""	""	"Y"
1	2	"135 Airways"	\N	""	"GNL"	"GENERAL"	"United States"	"N"
2	3	"1Time Airline"	\N	"1T"	"RNX"	"NEXTIME"	"South Africa"	"Y"
3	4	"2 Sqn No 1 Elementary Flying Training School"	\N	""	"WYT"	""	"United Kingdom"	"N"
4	5	"213 Flight Unit"	\N	""	"TFU"	""	"Russia"	"N"
5	6	"223 Flight Unit State Airline"	\N	""	"CHD"	"CHKALOVSK-AVIA"	"Russia"	"N"

Code for route dataset:

```
xl3 = pd.ExcelFile('Route dataset.xlsx')
df3 = xl3.parse('Sheet1', header=None)
dfx3 = df3[0].str.split(',',expand=True)
dfx3.columns=['airline','airline_id','source','source_id','dest','dest_id','codeshare','stops','equip
ment']
dfx3
```

Output:

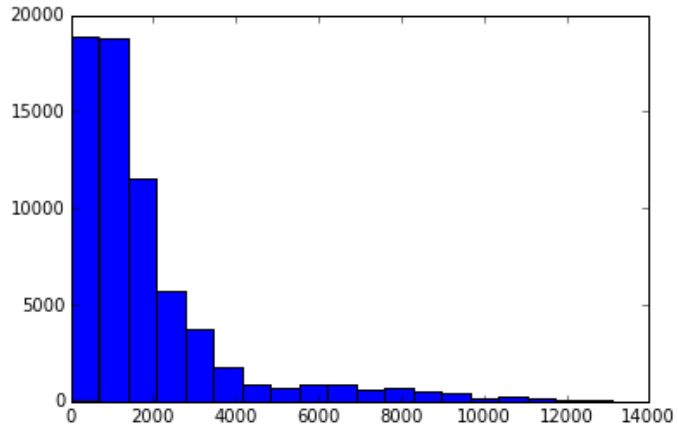
	airline	airline_id	source	source_id	dest	dest_id	codeshare	stops	equipment
0	2B	410	AER	2965	KZN	2990		0	CR2
1	2B	410	ASF	2966	KZN	2990		0	CR2
2	2B	410	ASF	2966	MRV	2962		0	CR2
3	2B	410	CEK	2968	KZN	2990		0	CR2
4	2B	410	CEK	2968	OVB	4078		0	CR2
5	2B	410	DME	4029	KZN	2990		0	CR2

2. Make histogram for route length, bin the values into ranges and count how many routes fall into each range

Code:

```
import math
def haversine(lon1, lat1, lon2, lat2):
    lon1, lat1, lon2, lat2 = [float(lon1), float(lat1), float(lon2), float(lat2)]
    lon1, lat1, lon2, lat2 = map(math.radians, [lon1, lat1, lon2, lat2])
    dlon = lon2 - lon1
    dlat = lat2 - lat1
    a = math.sin(dlat/2)**2 + math.cos(lat1) * math.cos(lat2) * math.sin(dlon/2)**2
    c = 2 * math.asin(math.sqrt(a))
    km = 6367 * c
    return km
dfx3 = dfx3[dfx3["airline_id"] != "\\N"]
def calc_dist(row):
    dist = 0
    try:
        source = dfx[dfx["id"] == row["source_id"]].iloc[0]
        dest = dfx[dfx["id"] == row["dest_id"]].iloc[0]
        dist = haversine(dest["longitude"], dest["latitude"], source["longitude"], source["latitude"])
    except (ValueError, IndexError):
        pass
    return dist
route_lengths = dfx3.apply(calc_dist, axis=1)
import matplotlib.pyplot as plt
%matplotlib inline
plt.hist(route_lengths, bins=20)
```

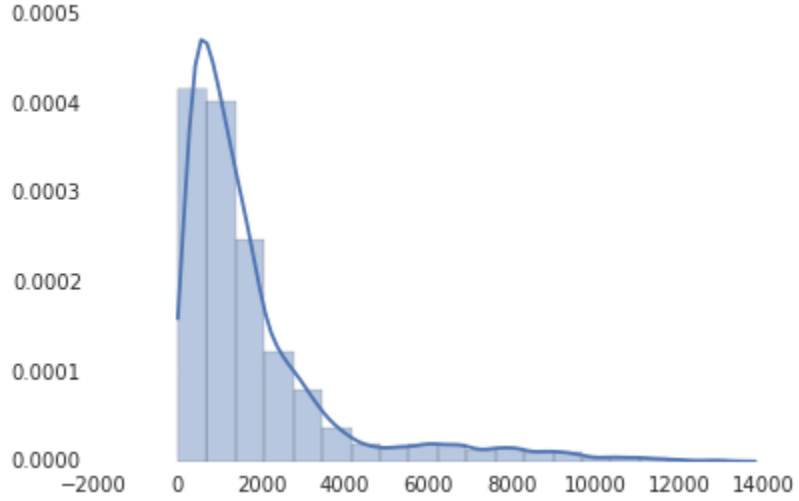
Output:



a. Use seaborn for route dataset (route Length)

```
import seaborn
seaborn.distplot(route_lengths, bins=20)
```

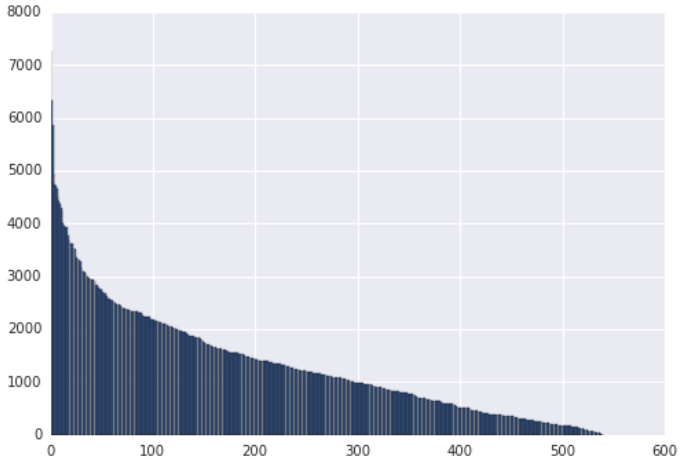
Output:



b. Bar chart - plot each airline against the average route length each airline flies

```
import numpy
import pandas
route_length_df = pandas.DataFrame({"length": route_lengths, "id": dfx3["airline_id"]})
airline_route_lengths = route_length_df.groupby("id").aggregate(numpy.mean)
airline_route_lengths = airline_route_lengths.sort_values("length", ascending=False)
plt.bar(range(airline_route_lengths.shape[0]), airline_route_lengths["length"])
```

Output:



c. Create a scatter plot comparing the airline ids to the name lengths

```
name_lengths = dfx2["name"].apply(lambda x: len(str(x)))
plt.scatter(dfx2["id"].astype(int), name_lengths)
```

Output: