



LISBON
SCHOOL OF
ECONOMICS &
MANAGEMENT

UNIVERSIDADE DE LISBOA

Carlos J. Costa

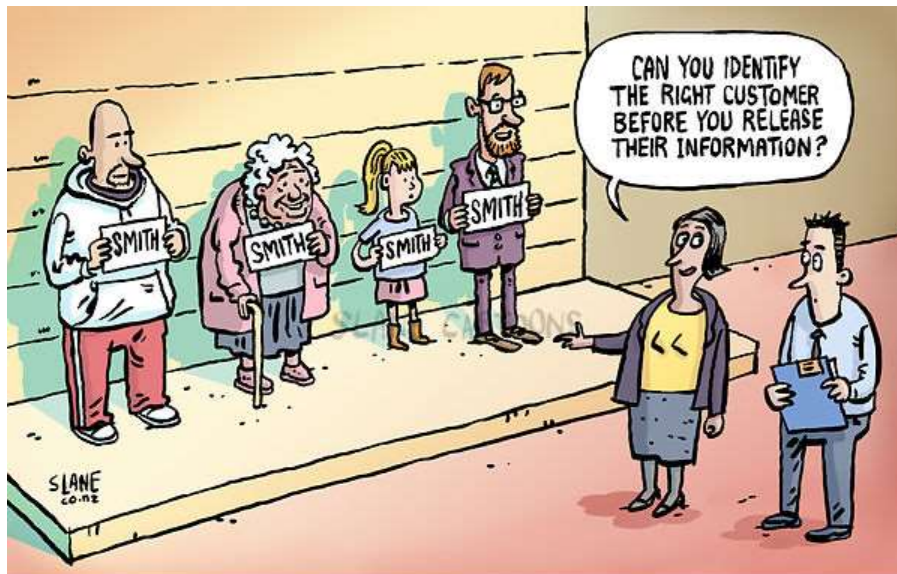
EXPLORATORY DATA ANALYSIS

(2021)

Summary

- Content identification
- Variables Type and Domain
- Missing values
- Variables Distribution
- Granularity
- Multi-Variate Analysis

Content identification



Double check recipients before you hit 'send.'

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
file='https://raw.githubusercontent.com/isegul/progtech/master/WorldBankPort.csv'
data = pd.read_csv(file, sep=";")
data.shape
```

(55, 6)

Variables Type and Domain

```
data.dtypes
```

```
Transports      float64  
other           float64  
ManAndConst     float64  
Electricity     float64  
ResidCom        float64  
year            int64  
dtype: object
```

Missing values

- It is possible analyse the numbers by plotting them through a bar chart.

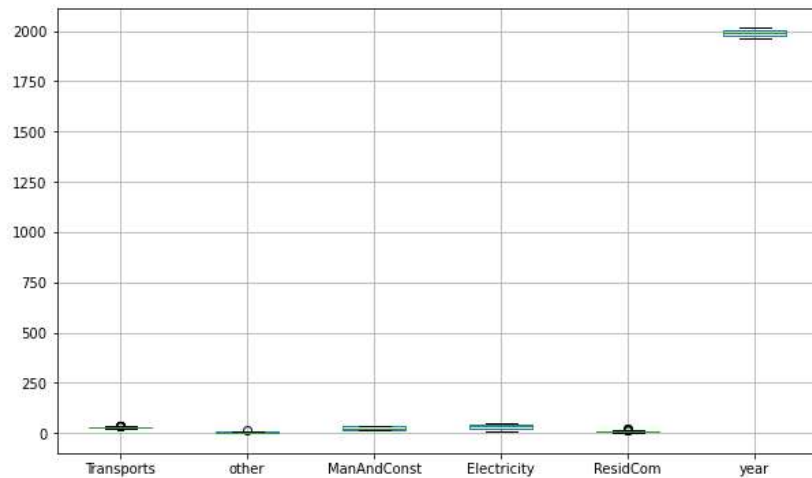


Variables Distribution

```
data.describe()
```

	Transports	other	ManAndConst	Electricity	ResidCom	year
count	55.000000	55.000000	55.000000	55.000000	55.000000	55.000000
mean	29.897565	5.128321	24.689692	31.509938	8.768899	1987.000000
std	3.280454	2.955396	8.482740	12.383553	3.573663	16.02082
min	24.400754	2.059014	12.193743	7.747489	5.276683	1960.000000
25%	27.820253	2.882284	17.487552	20.912832	6.423182	1973.500000
50%	29.160935	4.212300	25.297297	34.630631	7.554638	1987.000000
75%	30.680251	6.334884	32.625009	42.250330	9.593983	2000.500000
max	38.948475	13.157895	36.354430	46.481779	19.901316	2014.000000

```
data.boxplot(figsize=(10,6))  
plt.show()
```



Statistics

Plot the ditribution

Granularity

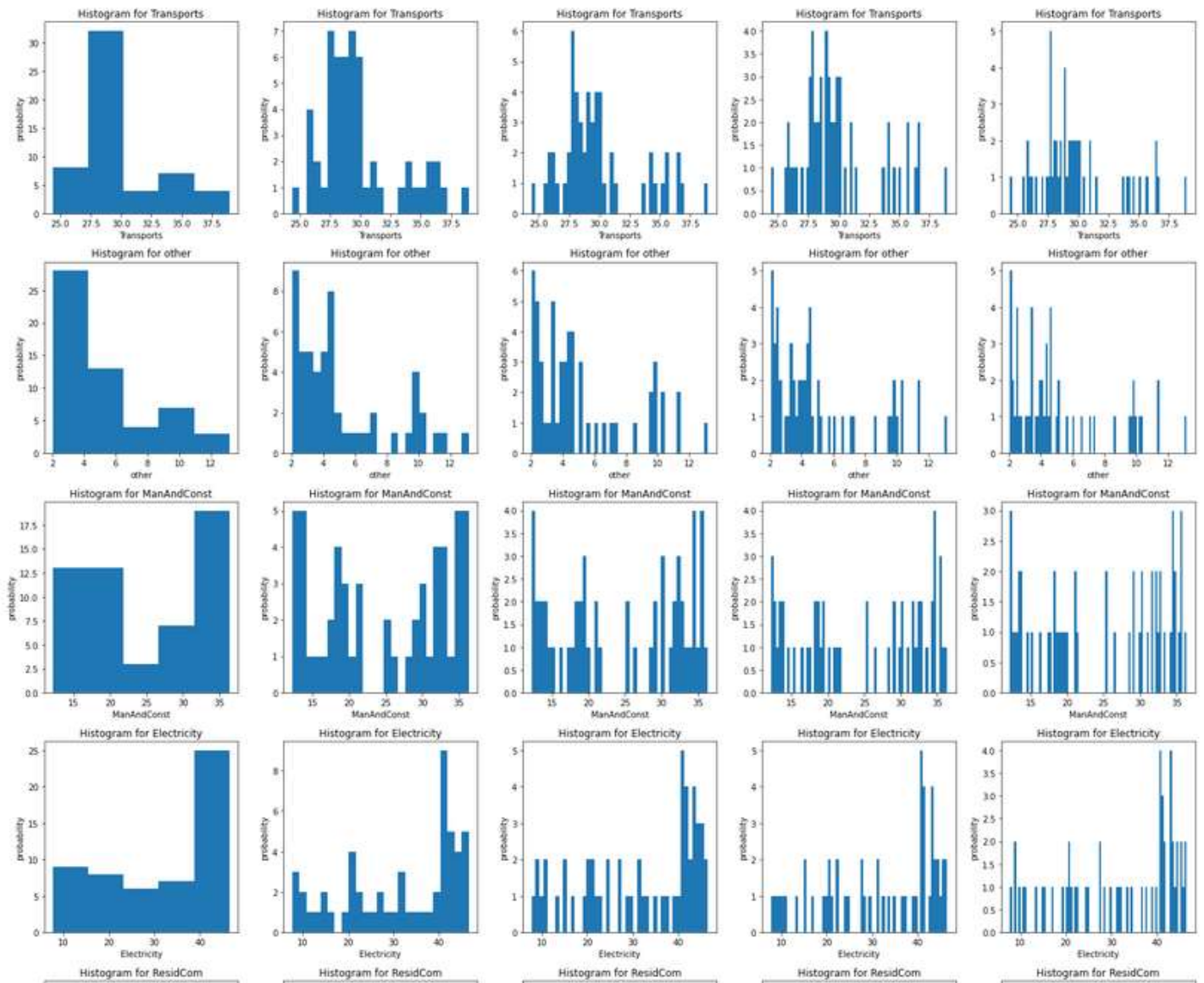
```
columns = data.select_dtypes(include='number').columns
rows = len(columns)
cols = 5
plt.figure()
fig, axs = plt.subplots(rows, cols, figsize=(cols*4, rows*4), squeeze=False)
bins = range(5, 100, 20)
for i in range(len(columns)):
    for j in range(len(bins)):
        axs[i, j].set_title('Histogram for %s'%columns[i])
        axs[i, j].set_xlabel(columns[i])
        axs[i, j].set_ylabel("probability")
        axs[i, j].hist(data[columns[i]].dropna().values, bins[j])
fig.tight_layout()
plt.show()
```



```

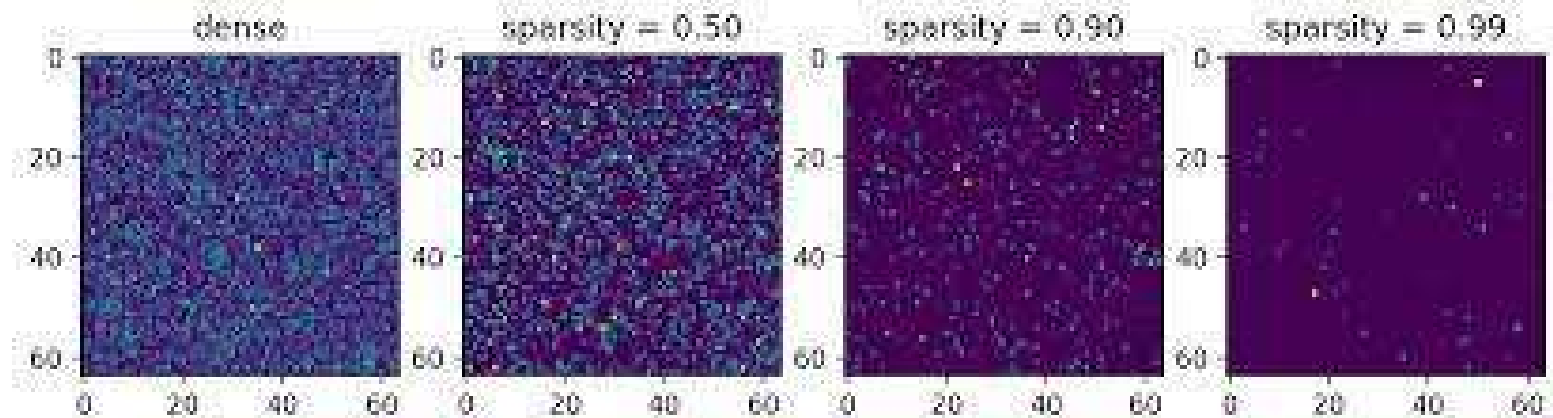
columns = data.select_
rows = len(columns)
cols = 5
plt.figure()
fig, axs = plt.subplots
bins = range(5, 100, 20)
for i in range(len(columns))
    for j in range(len(columns))
        axs[i, j].set_title
        axs[i, j].set_xlabel
        axs[i, j].set_ylabel
        axs[i, j].hist
fig.tight_layout()
plt.show()

```



Multi-Variate Analysis

- *Sparsity*
- *Correlation analysis*



Correlation analysis

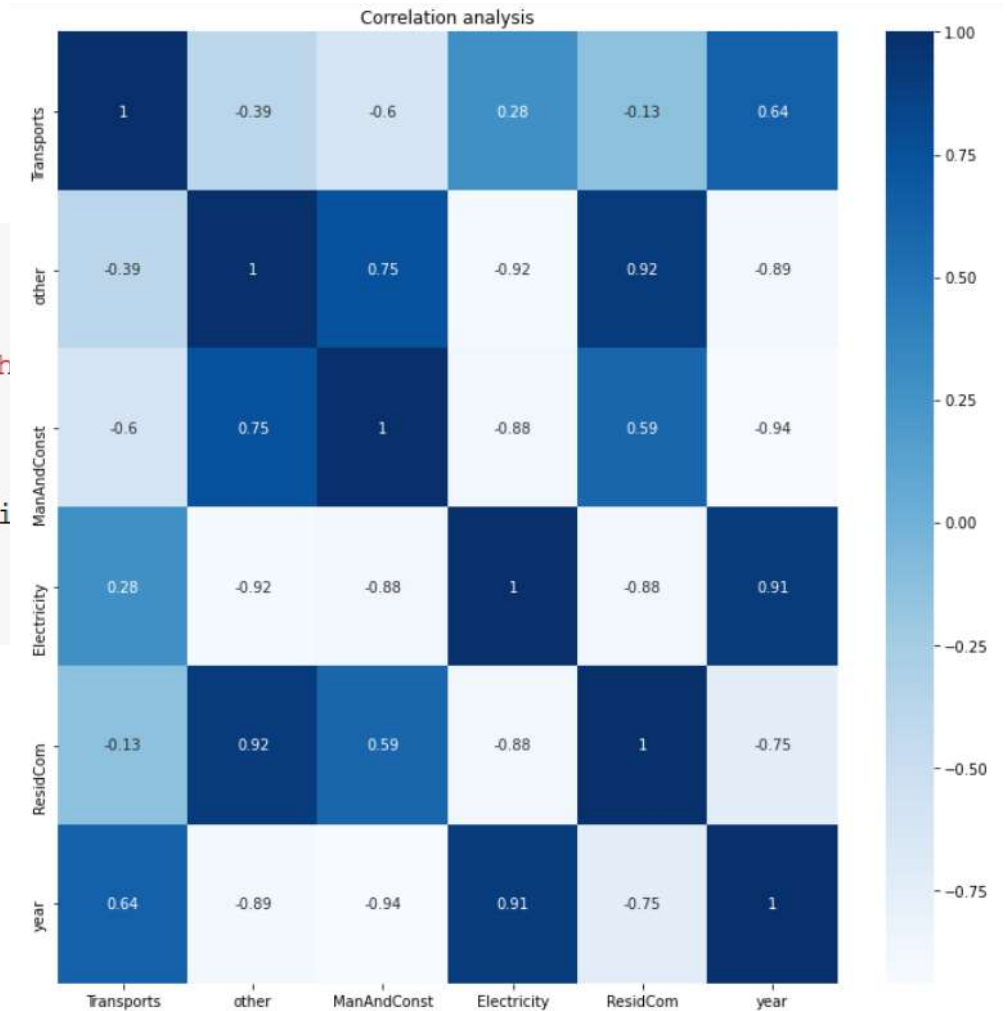
- Correlation analysis

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
file='https://raw.githubusercontent.com/isegul/progtech/master/WorldBankPort.csv'
data = pd.read_csv(file, sep=";")
fig = plt.figure(figsize=[12, 12])
corr_mtx = data.corr()
sns.heatmap(corr_mtx, xticklabels=corr_mtx.columns, yticklabels=corr_mtx.columns, annot=True, cmap='Blues')
plt.title('Correlation analysis')
plt.show()
```

Correlation analysis

- Correlation analysis

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
file='https://raw.githubusercontent.com/isegul/progtech
data = pd.read_csv(file,sep=";")
fig = plt.figure(figsize=[12, 12])
corr_mtx = data.corr()
sns.heatmap(corr_mtx, xticklabels=corr_mtx.columns, yti
plt.title('Correlation analysis')
plt.show()
```



Conclusion

- Content identification
- Variables Type and Domain
- Missing values
- Variables Distribution
- Granularity
- Multi-Variate Analysis