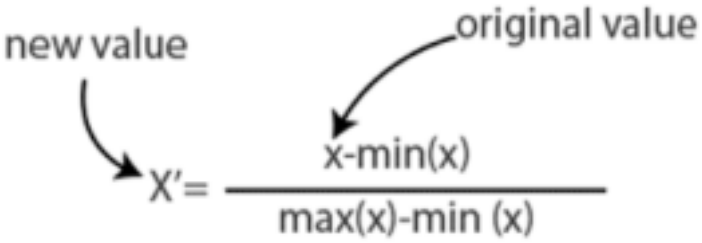
In every dataset we have different columns has different units
In every dataset we have different columns has values varies from -inf to inf
It is very important standardize the data, make sure all the column values under same range
To achieve this we have two methods
Normalization
standardization

Normalization:

min max scalar



Standardization

Z-score

$$Z = \frac{x - \mu}{\sigma}$$

In [2]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [3]:
```
file_path="C:\\Users\\omkar\\OneDrive\\Documents\\Data science\\Naresh IT\\"
visa_df=pd.read_csv(file_path)
visa_df
```

Out[3]:

	case_id	continent	education_of_employee	has_job_experience	requires_job_traini
0	EZYV01	Asia	High School	N	
1	EZYV02	Asia	Master's	Y	
2	EZYV03	Asia	Bachelor's	N	
3	EZYV04	Asia	Bachelor's	N	
4	EZYV05	Africa	Master's	Y	
...	...	...	...	...	...
25475	EZYV25476	Asia	Bachelor's	Y	
25476	EZYV25477	Asia	High School	Y	
25477	EZYV25478	Asia	Master's	Y	
25478	EZYV25479	Asia	Master's	Y	
25479	EZYV25480	Asia	Bachelor's	Y	

25480 rows × 12 columns

In [ ]: In [5]:
```
# step-5: Nr/Dr

######################## Read the data
#####################################
file_path="C:\\Users\\omkar\\OneDrive\\Documents\\Data science\\Naresh IT\\"
visa_df=pd.read_csv(file_path)
```

```
# step-1: calcaulate min value of p_Wage= min_wage
# step-2: calculate max value of p_wage = max_wage
# step-3: Dr= max_wage-min_wage
# step-4: Nr= p_wage-min_wage

min_wage=visa_df['prevailing_wage'].min()
max_wage=visa_df['prevailing_wage'].max()
dr=max_wage-min_wage
nr=visa_df['prevailing_wage']-min_wage
visa_df['prevailing_wage_norm']=nr/dr
```


In [7]:
```
visa_df[['prevailing_wage','prevailing_wage_norm']]
```

Out[7]:

	prevailing_wage	prevailing_wage_norm
0	592.2029	0.001849
1	83425.6500	0.261345
2	122996.8600	0.385312
3	83434.0300	0.261371
4	149907.3900	0.469616
...	...	...
25475	77092.5700	0.241505
25476	279174.7900	0.874579
25477	146298.8500	0.458311
25478	86154.7700	0.269895
25479	70876.9100	0.222033

25480 rows × 2 columns

In [9]:
```
visa_df['prevailing_wage_norm'].max(),visa_df['prevailing_wage_norm'].min()
```

Out[9]: (1.0, 0.0)

In [10]:
```
visa_df['prevailing_wage'].max(),visa_df['prevailing_wage'].min()
```

Out[10]: (319210.27, 2.1367)

In [14]:
```
max_id=visa_df['prevailing_wage_norm'].idxmax()
min_id=visa_df['prevailing_wage_norm'].idxmin()
max_id,min_id
```

Out[14]: (21077, 20575)

In [16]:
```
visa_df[['prevailing_wage','prevailing_wage_norm']].iloc[[max_id,min_id]]
```

Out[16]:

	prevailing_wage	prevailing_wage_norm
21077	319210.2700	1.0
20575	2.1367	0.0

MinMaxScalar

MinMaxScalar is a method from sklearn preprocessing
Read the packages
Save the package
Apply fit transform

In [27]:
```
################### Read the data
############################
file_path="C:\\Users\\omkar\\OneDrive\\Doc
uments\\Data science\\Naresh IT\\
visa_df=pd.read_csv(file_path)

# step-1:
```

In [29]:
```
from sklearn.preprocessing import
MinMaxScaler
#step-2:
mms=MinMaxScaler()
#step-3:
visa_df['prevailing_wage_norm1']=mms.fit_t

ransform(visa_df[['prevailing_wag

visa_df[['prevailing_wage_norm1','prevaili

ng_wage']]
```

Out[29]:

	prevailing_wage_norm1	prevailing_wage
0	0.001849	592.2029
1	0.261345	83425.6500
2	0.385312	122996.8600
3	0.261371	83434.0300
4	0.469616	149907.3900
...	...	...
25475	0.241505	77092.5700
25476	0.874579	279174.7900
25477	0.458311	146298.8500
25478	0.269895	86154.7700
25479	0.222033	70876.9100

25480 rows × 2 columns

Note:

Inside MinMaxScalar pass dataframe not series

In [21]:
```
    # array
    v1=np.array([[[1,
    2,3,4]]]) v1.ndim
```

Out[21]: 3

In [26]:
```
visa_df[['prevaili
ng_wage']]
```

Out[26]:

	prevailing_wage
0	592.2029
1	83425.6500
2	122996.8600
3	83434.0300
4	149907.3900
...	...
25475	77092.5700
25476	279174.7900
25477	146298.8500
25478	86154.7700
25479	70876.9100

25480 rows × 1 columns

Z-score

In [31]:
```
# step-1: calculate mean
# step-2: calculate std
# step-3: Nr= x-mean
# step-4: Nr/Std

mean_wage=visa_df['prevailing_w
age'].mean()
std_wage=visa_df['prevailing_wa
ge'].std()
nr=visa_df['prevailing_wage']-m
ean_wage
visa_df['prevailing_wage_zscore
']=nr/std_wage

visa_df[['prevailing_wage','pre
vailing_wage_zscore']]
```

In [32]:

Out[32]:

	prevailing_wage	prevailing_wage_zscore
0	592.2029	

-1.398510

1 83425.6500 0.169832

2 122996.8600 0.919060

3 83434.0300 0.169991

4 149907.3900 1.428576

... ...

25475 77092.5700 0.049923 **25476** 279174.7900

3.876083 **25477** 146298.8500 1.360253 **25478**

86154.7700 0.221504 **25479** 70876.9100 -0.067762

25480 rows × 2 columns

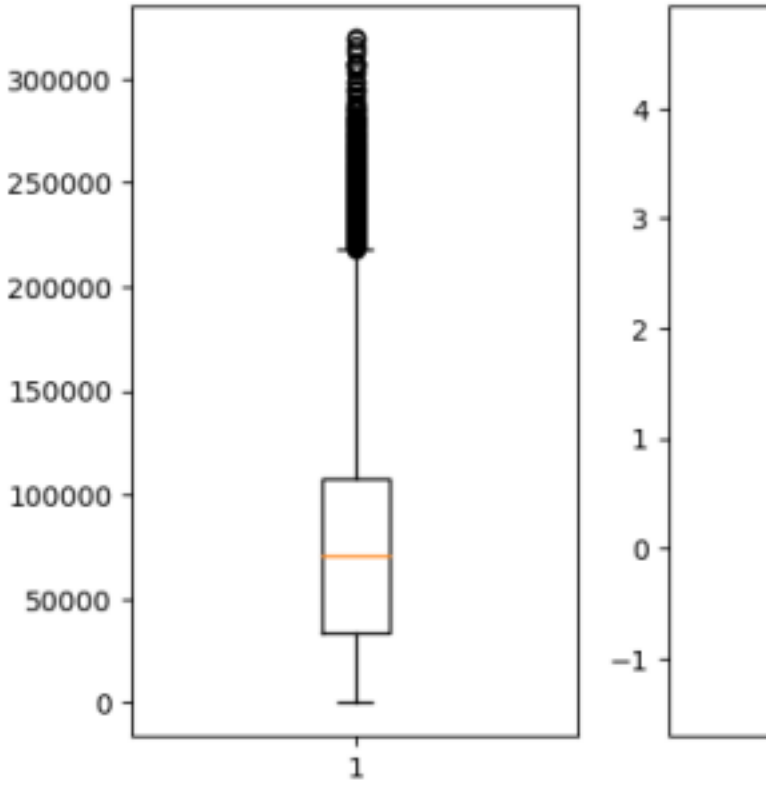
In [35]:

```
plt.subplot(1,2,1)
plt.boxplot(visa_df['prevailing_wage'])
```

Out[36]:
```
array([[-1.39853722],
       [ 0.1698353 ],
       [ 0.91907852],
       ...,
       [ 1.36027953],
       [ 0.22150859],
```

In [36]:

```
plt.subplot(1,2,2)
plt.boxplot(visa_df['prevailing_wage_zscore'])
plt.show()
```



StandardScalar

```
from sklearn.preprocessing import StandardScaler
ss=StandardScaler()
ss.fit_transform(visa_df[['prevailing_wage']])
```

```
       [-0.06776315]])
```


In [ ]:
```
Q1
Q2
Q3
IQR=Q3-Q1
UB=Q3+1.5*IQR
LB=Q1-1.5*IQR

con1=visa_df['prevailing_wage_z']>UB
con2=visa_df['prevailing_wage_z']<LB

con=con1|con2
visa_df['prevailing_wage_z'][con]
```