

```
In [1]: from __future__ import print_function

import pandas as pd
pd.__version__
```

Out[1]: '0.20.3'

```
In [16]: #read economeic date
ecnm_dataframe = pd.read_csv("ml_data_all.csv", sep=",")
ecnm_dataframe.describe()
```

Out[16]:

	y	m	d	tw_Open	tw_High	tw_Low	tw_Clk
<b>count</b>	5186.000000	5186.000000	5186.000000	5185.000000	5185.000000	5185.000000	5185.000000
<b>mean</b>	2007.523718	6.667759	15.811994	7523.724336	7566.244759	7468.191440	7513.696000
<b>std</b>	6.079854	3.397161	8.707811	1656.577972	1654.319321	1655.296814	1654.134000
<b>min</b>	1997.000000	1.000000	1.000000	3475.870117	3511.379883	3411.679932	3446.260000
<b>25%</b>	2002.000000	4.000000	8.000000	6194.899902	6231.879883	6151.509766	6190.830000
<b>50%</b>	2008.000000	7.000000	16.000000	7712.029785	7752.089844	7657.879883	7698.240000
<b>75%</b>	2013.000000	10.000000	23.000000	8702.959961	8753.049805	8651.540039	8702.330000
<b>max</b>	2018.000000	12.000000	31.000000	11243.589844	11270.179688	11201.519531	11253.110000

8 rows × 78 columns



```
In [17]: ecnm_dataframe = pd.read_csv("ml_data_1.csv", sep=",")
ecnm_dataframe.describe()
```

Out[17]:

	y	m	d	tw_Open	tw_High	tw_Low	tw_Clk
<b>count</b>	5186.000000	5186.000000	5186.000000	5185.000000	5185.000000	5185.000000	5185.000000
<b>mean</b>	2007.523718	6.667759	15.811994	7523.724336	7566.244759	7468.191440	7513.696000
<b>std</b>	6.079854	3.397161	8.707811	1656.577972	1654.319321	1655.296814	1654.134000
<b>min</b>	1997.000000	1.000000	1.000000	3475.870117	3511.379883	3411.679932	3446.260000
<b>25%</b>	2002.000000	4.000000	8.000000	6194.899902	6231.879883	6151.509766	6190.830000
<b>50%</b>	2008.000000	7.000000	16.000000	7712.029785	7752.089844	7657.879883	7698.240000
<b>75%</b>	2013.000000	10.000000	23.000000	8702.959961	8753.049805	8651.540039	8702.330000
<b>max</b>	2018.000000	12.000000	31.000000	11243.589844	11270.179688	11201.519531	11253.110000



```
In [18]: ecnm_dataframe = pd.read_csv("ml_data_2.csv", sep=",")
ecnm_dataframe.describe()
```

Out[18]:

	jp_Adj_Close	chnsse_Open	chnsse_Adj_Close	chnhs_Open	chnhs_Adj_Close	ko_Open
<b>count</b>	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000
<b>mean</b>	13956.092990	2287.098108	2288.878604	18178.526521	18169.386320	1401.925150
<b>std</b>	3928.537029	930.138147	931.448029	5789.014166	5782.338433	623.541041
<b>min</b>	1144.916000	1007.901000	1011.499000	4672.070000	2938.100000	283.410000
<b>25%</b>	10373.847500	1519.945250	1520.580500	13210.107500	13215.982500	786.040000
<b>50%</b>	14037.285000	2113.359500	2114.669000	18913.745000	18905.184500	1450.005000
<b>75%</b>	16921.307500	2947.748000	2949.483750	22726.300000	22704.117750	1968.830000
<b>max</b>	24124.150000	6057.428000	6092.057000	33335.480000	33154.121000	3220.000000

```
In [19]: ecnm_dataframe = pd.read_csv("ml_data_3.csv", sep=",")
ecnm_dataframe.describe()
```

Out[19]:

	ind_Adj_Close	astsp_Open	astsp_Adj_Close	ast_Open	ast_Adj_Close	usdow_Open	us
<b>count</b>	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	
<b>mean</b>	14004.646563	4355.366554	4355.964751	4370.722194	4371.246172	12639.993527	
<b>std</b>	9512.654983	1115.102891	1115.387994	1125.794460	1126.097450	4087.608399	
<b>min</b>	2600.120000	2260.900000	2260.900000	2299.200000	2299.200000	6547.010000	
<b>25%</b>	4488.792500	3298.575000	3299.625000	3261.062500	3261.775000	10061.377500	
<b>50%</b>	14042.860000	4457.000000	4455.900000	4458.100000	4459.100000	11064.247500	
<b>75%</b>	19900.193500	5317.375000	5318.400000	5333.475000	5332.725000	14617.312500	
<b>max</b>	37606.578000	6794.500000	6828.700000	6811.100000	6853.600000	26584.279000	

```
In [29]: ecnm_dataframe = pd.read_csv("ml_data_4.csv", sep=",")
ecnm_dataframe.describe()
```

Out[29]:

	usvix_Adj_Close	ussp_Open	ussp_Adj_Close	eurestx_Open	eurestx_Adj_Close	blx_Op
<b>count</b>	5186.000000	5186.000000	5186.000000	5213.000000	5213.000000	5186.0000
<b>mean</b>	20.344425	1435.203605	1435.458626	3242.090240	3242.084505	3000.2245
<b>std</b>	8.486162	463.966119	464.022736	708.064511	708.063922	676.7434
<b>min</b>	9.140000	679.280000	676.530000	1809.980000	1809.980000	1450.3700
<b>25%</b>	14.010000	1121.015000	1120.796250	2721.240000	2721.240000	2485.0550
<b>50%</b>	18.745000	1299.535000	1299.105000	3118.910000	3118.910000	2962.0950
<b>75%</b>	24.187500	1567.962500	1569.045000	3618.640000	3618.640000	3501.6900
<b>max</b>	80.860000	2867.230000	2872.870000	5464.430000	5464.430000	4737.3200

```
In [28]: ecnm_dataframe = pd.read_csv("ml_data_5.csv", sep=",")
ecnm_dataframe.describe()
```

Out[28]:

	grm_Adj_Close	cnd_Open	cnd_Adj_Close	mxo_Open	mxo_Adj_Close	agt_Open	ag
<b>count</b>	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5213.000000	
<b>mean</b>	6829.775136	11075.066998	11072.683976	24895.056571	24903.838158	4808.019016	
<b>std</b>	2614.734512	3013.843449	3013.078529	16103.750579	16105.519370	7250.446800	
<b>min</b>	2202.960000	5343.800000	5398.100000	2859.290000	2856.100000	199.640000	
<b>25%</b>	4915.950000	8091.575000	8119.925000	6864.862500	6860.917500	674.460000	
<b>50%</b>	6210.040000	11682.050000	11672.925000	26657.200500	26681.554500	1929.170000	
<b>75%</b>	8007.991250	13582.400000	13578.675000	40801.646250	40825.009000	3499.080000	
<b>max</b>	13559.600000	16560.900000	16567.400000	51590.480000	51713.379000	35141.719000	3

```
In [22]: ecnm_dataframe = pd.read_csv("ml_data_6.csv", sep=",")
ecnm_dataframe.describe()
```

Out[22]:

	bx_Adj_Close	lsrl_Open	lsrl_Adj_Close	tw_VALUE	jp_VALUE	chn_VALUE	chnhk_V
<b>count</b>	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5186.0
<b>mean</b>	40032.655226	843.672323	843.371445	31.895417	108.180731	7.349908	7.7
<b>std</b>	22230.677196	366.288780	365.945394	1.714437	14.335341	0.866049	0.0
<b>min</b>	4761.000000	249.190000	249.190000	27.870000	75.720000	6.040200	7.7
<b>25%</b>	15396.500000	466.387500	466.380000	30.350000	101.332500	6.503275	7.7
<b>50%</b>	45508.000000	925.780000	926.470000	32.106500	109.890000	7.257050	7.7
<b>75%</b>	58617.500000	1166.212500	1164.772500	33.080000	118.647500	8.277000	7.7
<b>max</b>	87653.000000	1503.080000	1502.080000	35.210000	147.140000	8.322000	7.8

```
In [23]: ecnm_dataframe = pd.read_csv("ml_data_7.csv", sep=",")
ecnm_dataframe.describe()
```

Out[23]:

	ko_VALUE	mlx_VALUE	mxo_VALUE	sd_VALUE	nafrc_VALUE	sgp_VALUE	sz_VALUE
<b>count</b>	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000
<b>mean</b>	1137.454379	3.625013	12.296046	7.820118	8.633056	1.508719	1.199086
<b>std</b>	126.276427	0.369092	3.025422	1.110295	2.654874	0.191115	0.259707
<b>min</b>	889.000000	2.489000	7.717200	5.834600	4.525000	1.200700	0.729600
<b>25%</b>	1065.160000	3.287575	10.168500	6.886350	6.727525	1.347525	0.970800
<b>50%</b>	1133.000000	3.780000	11.301000	7.765050	7.716500	1.484300	1.156600
<b>75%</b>	1194.787500	3.800000	13.239950	8.475075	10.356875	1.695875	1.390500
<b>max</b>	1960.000000	4.730000	21.891000	11.027000	16.884500	1.854000	1.825000

```
In [24]: ecnm_dataframe = pd.read_csv("ml_data_8.csv", sep=",")
ecnm_dataframe.describe()
```

Out[24]:

	uk_VALUE	RDSCUNT_RATE	RATE_YEAR	RATE	BOND_RATE	PRP_M	PF
<b>count</b>	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5186.000000	5186
<b>mean</b>	1.614409	2.438512	2.283199	4.344070	2.554080	22966.351909	9645
<b>std</b>	0.184281	1.250208	1.617852	1.996797	1.684152	7692.667969	4177
<b>min</b>	1.211800	1.250000	0.770000	2.563000	0.660000	13573.000000	5068
<b>25%</b>	1.501225	1.500000	1.280000	2.857000	1.380000	16262.000000	5857
<b>50%</b>	1.601500	1.875000	1.400000	3.478000	1.850000	20112.000000	8159
<b>75%</b>	1.689125	3.125000	2.540000	7.225000	2.980000	29377.000000	12892
<b>max</b>	2.110400	5.250000	6.530000	7.955000	6.700000	39993.000000	20084

```
In [27]: ecnm_dataframe = pd.read_csv("ml_data_9.csv", sep=",")
ecnm_dataframe.describe()
```

Out[27]:

	PUR_A	ASSETS	LIABILITIES
<b>count</b>	5186.000000	5186.000000	5186.000000
<b>mean</b>	103656.390667	120254.397030	16598.006363
<b>std</b>	48323.653236	55601.559747	7887.055066
<b>min</b>	26101.000000	31400.000000	4027.000000
<b>25%</b>	63549.000000	70726.000000	7311.000000
<b>50%</b>	96666.000000	117205.000000	18931.000000
<b>75%</b>	136094.000000	156243.000000	22202.000000
<b>max</b>	185452.000000	210266.000000	32963.000000

In [\*]:

In [ ]: