

# Exploring a dataset

June 6, 2018

```
In [3]: import numpy as np
import pandas as pd
% matplotlib inline
```

```
In [4]: dataframe = pd.read_csv('weather.csv')
```

```
In [7]: #retorna uma tuple contendo o tamanho do Data Frame, n° de linhas e n° de colunas
dataframe.shape
```

```
Out[7]: (8784, 5)
```

```
In [9]: #informações das colunas
dataframe.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8784 entries, 0 to 8783
Data columns (total 5 columns):
MONTH      8784 non-null int64
DAY        8784 non-null int64
TIME       8784 non-null int64
TEMP       8784 non-null float64
PRESSURE   8784 non-null int64
dtypes: float64(1), int64(4)
memory usage: 343.2 KB
```

```
In [12]: #retorna as primeiras 3 linhas
dataframe.head(3)
```

```
Out[12]:
```

	MONTH	DAY	TIME	TEMP	PRESSURE
0	1	1	1	6.8	10207
1	1	1	2	5.8	10214
2	1	1	3	5.7	10220

```
In [14]: #retorna as últimas 3 linhas
dataframe.tail(3)
```

```
Out[14]:
```

	MONTH	DAY	TIME	TEMP	PRESSURE
8781	12	366	22	3.0	10253
8782	12	366	23	2.8	10246
8783	12	366	24	2.8	10239

```
In [16]: #para cada coluna do dataframe a função describe retorna as informações estatísticas
dataframe.describe()
```

```
Out[16]:
```

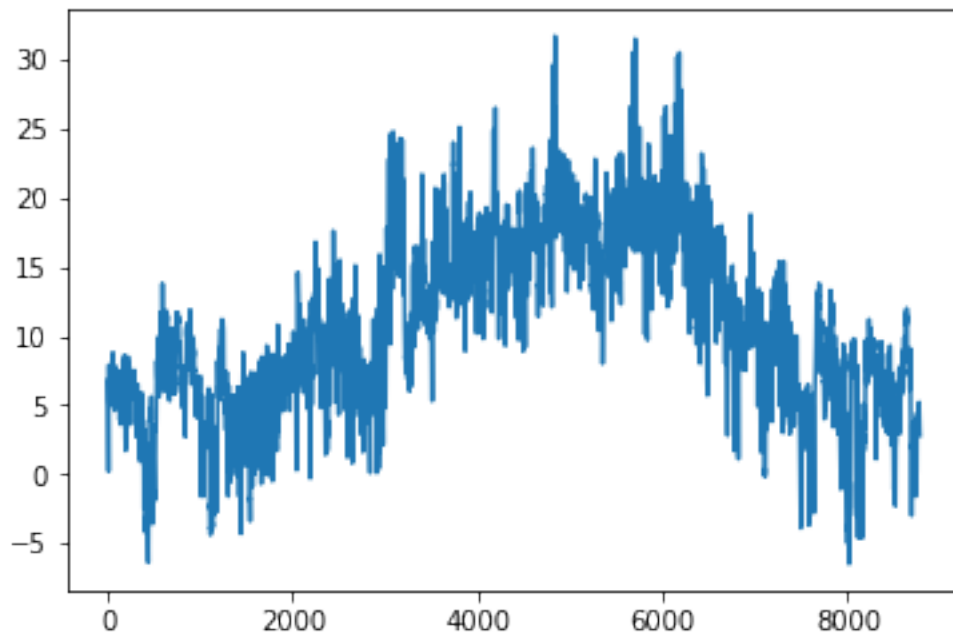
	MONTH	DAY	TIME	TEMP	PRESSURE
count	8784.000000	8784.000000	8784.000000	8784.000000	8784.000000
mean	6.513661	183.500000	12.500000	10.768613	10159.254781
std	3.451430	105.660719	6.922581	6.420003	102.895130
min	1.000000	1.000000	1.000000	-6.500000	9812.000000
25%	4.000000	92.000000	6.750000	6.100000	10102.000000
50%	7.000000	183.500000	12.500000	10.200000	10169.000000
75%	10.000000	275.000000	18.250000	15.900000	10227.000000
max	12.000000	366.000000	24.000000	31.700000	10453.000000

```
In [17]: #retorna a temperatura média da coluna 'TEMP' do dataframe
dataframe['TEMP'].mean()
```

```
Out[17]: 10.768613387978167
```

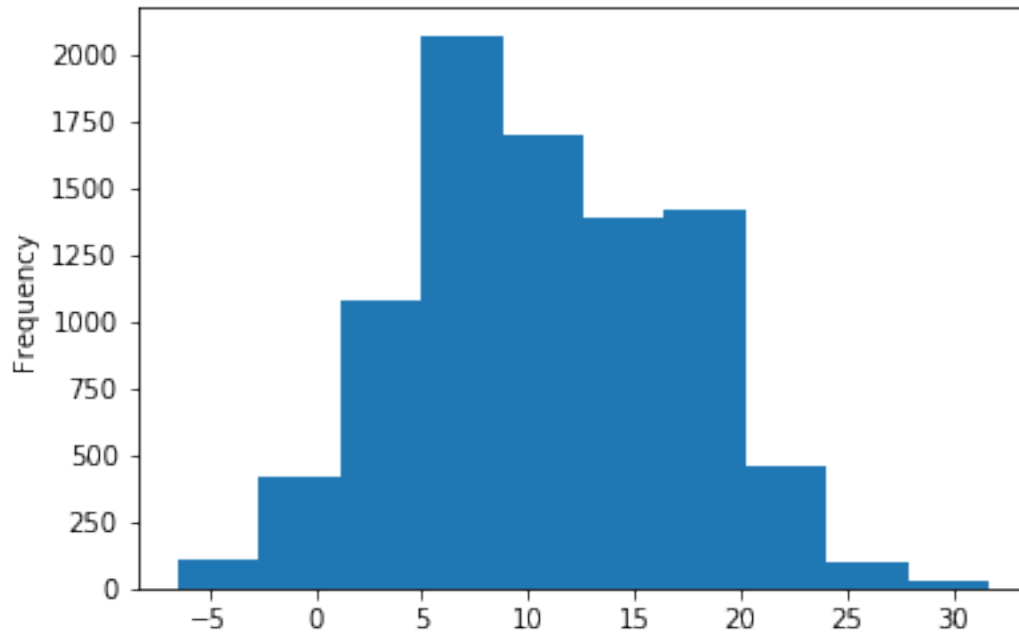
```
In [19]: dataframe['TEMP'].plot()
```

```
Out[19]: <matplotlib.axes._subplots.AxesSubplot at 0x9a1cfd0>
```



```
In [20]: dataframe['TEMP'].plot.hist()
```

```
Out[20]: <matplotlib.axes._subplots.AxesSubplot at 0x9c11c88>
```



```
In [22]: dataframe['TEMP'].plot.hist(bins=50)
```

```
Out[22]: <matplotlib.axes._subplots.AxesSubplot at 0xad2f198>
```

