MAJOR PROJECT 1

By Akshat Burman

IIT Bhubaneswar

2nd year

Mechanical Engineering

Single variable regressor model:

```
import pandas as pd
df = pd.read_csv('/content/annual.csv')
df
```

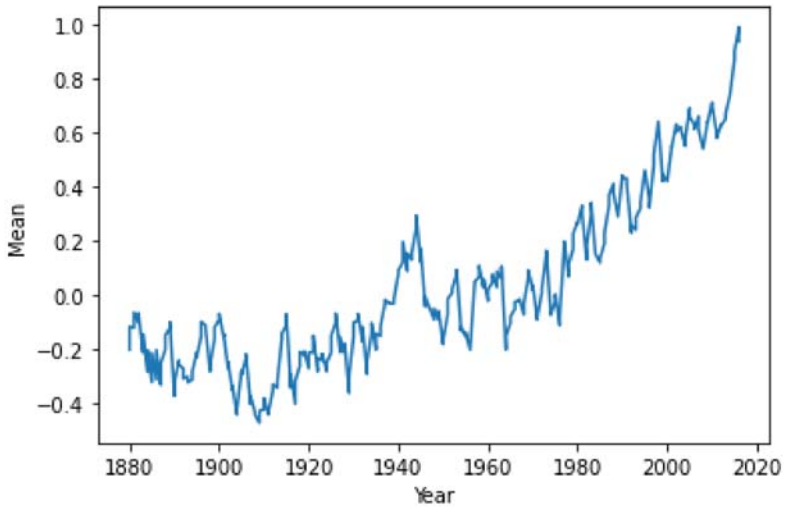
	Source	Year	Mean
0	GCAG	2016	0.9363
1	GISTEMP	2016	0.9900
2	GCAG	2015	0.8998
3	GISTEMP	2015	0.8700
4	GCAG	2014	0.7408
...	...	...	...
269	GISTEMP	1882	-0.1000
270	GCAG	1881	-0.0628
271	GISTEMP	1881	-0.1200
272	GCAG	1880	-0.1148
273	GISTEMP	1880	-0.2000

274 rows × 3 columns

```
#Data Visualisation
import matplotlib.pyplot as plt
import numpy as np
x= df['Year'].to_numpy()
y= df['Mean'].to_numpy()
plt.xlabel('Year')
plt.ylabel('Mean')

plt.plot(x,y)
```

```
[<matplotlib.lines.Line2D at 0x7f5fa8727390>]
```



```
x = df.iloc[:,1:2].values
x
```

```
array([[2016],
       [2016],
       [2015],
       [2015],
       [2014],
       [2014],
       [2013],
       [2013],
       [2012],
       [2012],
       [2011],
       [2011],
       [2010],
       [2010],
       [2009],
       [2009],
       [2008],
       [2008],
       [2007],
       [2007],
       [2006],
       [2006],
       [2005],
       [2005],
       [2004],
       [2004],
       [2003],
       [2003],
       [2002],
       [2002],
       [2001],
       [2001],
       [2000],
       [2000],
       [1999],
       [1999],
       [1998],
       [1998],
       [1997],
       [1997],
       [1996],
       [1996],
       [1995],
       [1995],
       [1994],
       [1994],
       [1993],
       [1993],
       [1992],
       [1992],
       [1991],
       [1991],
       [1990],
       [1990],
       [1989],
       [1989],
       [1988],
```

```
            [1988],
```

```
y = df.iloc[:,2].values
y
```

```
array([ 0.9363,  0.99  ,  0.8998,  0.87  ,  0.7408,  0.74  ,  0.6679,
        0.65  ,  0.624 ,  0.63  ,  0.5788,  0.6   ,  0.7014,  0.71  ,
        0.6367,  0.64  ,  0.5419,  0.54  ,  0.61  ,  0.66  ,  0.6125,
        0.63  ,  0.6585,  0.69  ,  0.5783,  0.55  ,  0.6134,  0.62  ,
        0.6023,  0.63  ,  0.5473,  0.55  ,  0.4262,  0.42  ,  0.4438,
        0.42  ,  0.6344,  0.64  ,  0.5187,  0.48  ,  0.3228,  0.35  ,
        0.4577,  0.46  ,  0.3409,  0.32  ,  0.2853,  0.24  ,  0.2571,
        0.23  ,  0.4055,  0.43  ,  0.4328,  0.44  ,  0.297 ,  0.29  ,
        0.3757,  0.41  ,  0.3696,  0.33  ,  0.2296,  0.19  ,  0.1342,
        0.12  ,  0.149 ,  0.15  ,  0.3411,  0.3   ,  0.1815,  0.13  ,
        0.2999,  0.33  ,  0.2637,  0.27  ,  0.2273,  0.17  ,  0.1123,
        0.07  ,  0.1978,  0.18  , -0.0792, -0.11  ,  0.0034, -0.02  ,
       -0.0719, -0.07  ,  0.1641,  0.15  ,  0.0264,  0.01  , -0.0783,
       -0.09  ,  0.0372,  0.02  ,  0.0929,  0.07  , -0.0296, -0.07  ,
       -0.0131, -0.02  , -0.0227, -0.05  , -0.078 , -0.1   , -0.1495,
       -0.2   ,  0.1068,  0.06  ,  0.0888,  0.03  ,  0.0775,  0.05  ,
        0.0204, -0.02  ,  0.0596,  0.03  ,  0.1095,  0.07  ,  0.0488,
        0.04  , -0.199 , -0.2   , -0.1354, -0.15  , -0.1165, -0.13  ,
        0.0952,  0.08  ,  0.0248,  0.01  , -0.0132, -0.07  , -0.1616,
       -0.18  , -0.0568, -0.09  , -0.0487, -0.09  , -0.0477, -0.05  ,
       -0.004 , -0.04  ,  0.171 ,  0.12  ,  0.2928,  0.25  ,  0.157 ,
        0.13  ,  0.1538,  0.09  ,  0.196 ,  0.12  ,  0.0947,  0.08  ,
       -0.0139, -0.03  , -0.0288, -0.03  , -0.0157, -0.03  , -0.1134,
       -0.15  , -0.1392, -0.2   , -0.1015, -0.14  , -0.2439, -0.29  ,
       -0.1168, -0.17  , -0.0686, -0.09  , -0.1003, -0.15  , -0.2985,
       -0.36  , -0.1774, -0.21  , -0.1546, -0.21  , -0.0667, -0.1   ,
       -0.1481, -0.21  , -0.2486, -0.28  , -0.2156, -0.24  , -0.2304,
       -0.28  , -0.1485, -0.21  , -0.2105, -0.27  , -0.2055, -0.22  ,
       -0.2084, -0.26  , -0.3146, -0.4   , -0.293 , -0.34  , -0.0693,
       -0.11  , -0.1395, -0.16  , -0.3162, -0.34  , -0.3288, -0.35  ,
       -0.4332, -0.44  , -0.3789, -0.42  , -0.4261, -0.47  , -0.4396,
       -0.43  , -0.3706, -0.4   , -0.2174, -0.23  , -0.2931, -0.28  ,
       -0.4194, -0.44  , -0.3369, -0.35  , -0.2463, -0.27  , -0.1417,
       -0.15  , -0.0679, -0.09  , -0.1173, -0.16  , -0.2546, -0.28  ,
       -0.1224, -0.11  , -0.0974, -0.15  , -0.229 , -0.21  , -0.2808,
       -0.31  , -0.3212, -0.3   , -0.3062, -0.27  , -0.2532, -0.24  ,
       -0.322 , -0.37  , -0.0982, -0.12  , -0.1471, -0.2   , -0.2489,
       -0.33  , -0.2003, -0.31  , -0.2125, -0.32  , -0.2009, -0.28  ,
       -0.1424, -0.21  , -0.0648, -0.1   , -0.0628, -0.12  , -0.1148,
       -0.2   ])
```

```
#Applying regressor
from sklearn.linear_model import LinearRegression
model = LinearRegression()
```

```
#Model fitting
model.fit(x,y)
```

```
LinearRegression()
```

```
y_pred = model.predict(x)
y_pred #Predicted
```

```
y_pred #Predicted
```

```
array([ 0.51171944,  0.51171944,  0.50473221,  0.50473221,  0.49774498,
        0.49774498,  0.49075776,  0.49075776,  0.48377053,  0.48377053,
        0.4767833 ,  0.4767833 ,  0.46979607,  0.46979607,  0.46280884,
        0.46280884,  0.45582162,  0.45582162,  0.44883439,  0.44883439,
        0.44184716,  0.44184716,  0.43485993,  0.43485993,  0.42787271,
        0.42787271,  0.42088548,  0.42088548,  0.41389825,  0.41389825,
        0.40691102,  0.40691102,  0.3999238 ,  0.3999238 ,  0.39293657,
        0.39293657,  0.38594934,  0.38594934,  0.37896211,  0.37896211,
        0.37197488,  0.37197488,  0.36498766,  0.36498766,  0.35800043,
        0.35800043,  0.3510132 ,  0.3510132 ,  0.34402597,  0.34402597,
        0.33703875,  0.33703875,  0.33005152,  0.33005152,  0.32306429,
        0.32306429,  0.31607706,  0.31607706,  0.30908984,  0.30908984,
        0.30210261,  0.30210261,  0.29511538,  0.29511538,  0.28812815,
        0.28812815,  0.28114092,  0.28114092,  0.2741537 ,  0.2741537 ,
        0.26716647,  0.26716647,  0.26017924,  0.26017924,  0.25319201,
        0.25319201,  0.24620479,  0.24620479,  0.23921756,  0.23921756,
        0.23223033,  0.23223033,  0.2252431 ,  0.2252431 ,  0.21825588,
        0.21825588,  0.21126865,  0.21126865,  0.20428142,  0.20428142,
        0.19729419,  0.19729419,  0.19030697,  0.19030697,  0.18331974,
        0.18331974,  0.17633251,  0.17633251,  0.16934528,  0.16934528,
        0.16235805,  0.16235805,  0.15537083,  0.15537083,  0.1483836 ,
        0.1483836 ,  0.14139637,  0.14139637,  0.13440914,  0.13440914,
        0.12742192,  0.12742192,  0.12043469,  0.12043469,  0.11344746,
        0.11344746,  0.10646023,  0.10646023,  0.09947301,  0.09947301,
        0.09248578,  0.09248578,  0.08549855,  0.08549855,  0.07851132,
        0.07851132,  0.07152409,  0.07152409,  0.06453687,  0.06453687,
        0.05754964,  0.05754964,  0.05056241,  0.05056241,  0.04357518,
        0.04357518,  0.03658796,  0.03658796,  0.02960073,  0.02960073,
        0.0226135 ,  0.0226135 ,  0.01562627,  0.01562627,  0.00863905,
        0.00863905,  0.00165182,  0.00165182, -0.00533541, -0.00533541,
       -0.01232264, -0.01232264, -0.01930987, -0.01930987, -0.02629709,
       -0.02629709, -0.03328432, -0.03328432, -0.04027155, -0.04027155,
       -0.04725878, -0.04725878, -0.054246  , -0.054246  , -0.06123323,
       -0.06123323, -0.06822046, -0.06822046, -0.07520769, -0.07520769,
       -0.08219491, -0.08219491, -0.08918214, -0.08918214, -0.09616937,
       -0.09616937, -0.1031566 , -0.1031566 , -0.11014383, -0.11014383,
       -0.11713105, -0.11713105, -0.12411828, -0.12411828, -0.13110551,
       -0.13110551, -0.13809274, -0.13809274, -0.14507996, -0.14507996,
       -0.15206719, -0.15206719, -0.15905442, -0.15905442, -0.16604165,
       -0.16604165, -0.17302887, -0.17302887, -0.1800161 , -0.1800161 ,
       -0.18700333, -0.18700333, -0.19399056, -0.19399056, -0.20097778,
       -0.20097778, -0.20796501, -0.20796501, -0.21495224, -0.21495224,
       -0.22193947, -0.22193947, -0.2289267 , -0.2289267 , -0.23591392,
       -0.23591392, -0.24290115, -0.24290115, -0.24988838, -0.24988838,
       -0.25687561, -0.25687561, -0.26386283, -0.26386283, -0.27085006,
       -0.27085006, -0.27783729, -0.27783729, -0.28482452, -0.28482452,
       -0.29181174, -0.29181174, -0.29879897, -0.29879897, -0.3057862 ,
       -0.3057862 , -0.31277343, -0.31277343, -0.31976066, -0.31976066,
       -0.32674788, -0.32674788, -0.33373511, -0.33373511, -0.34072234,
       -0.34072234, -0.34770957, -0.34770957, -0.35469679, -0.35469679,
       -0.36168402, -0.36168402, -0.36867125, -0.36867125, -0.37565848,
       -0.37565848, -0.3826457 , -0.3826457 , -0.38963293, -0.38963293,
       -0.39662016, -0.39662016, -0.40360739, -0.40360739, -0.41059462,
       -0.41059462, -0.41758184, -0.41758184, -0.42456907, -0.42456907,
       -0.4315563 , -0.4315563 , -0.43854353, -0.43854353])
```

```
y
```

```
array([ 0.9363,  0.99  ,  0.8998,  0.87  ,  0.7408,  0.74  ,  0.6679,
        0.65  ,  0.624 ,  0.63  ,  0.5788,  0.6   ,  0.7014,  0.71  ,
        0.6367,  0.64  ,  0.5419,  0.54  ,  0.61  ,  0.66  ,  0.6125,
        0.63  ,  0.6585,  0.69  ,  0.5783,  0.55  ,  0.6134,  0.62  ,
        0.6023,  0.63  ,  0.5473,  0.55  ,  0.4262,  0.42  ,  0.4438,
        0.42  ,  0.6344,  0.64  ,  0.5187,  0.48  ,  0.3228,  0.35  ,
        0.4577,  0.46  ,  0.3409,  0.32  ,  0.2853,  0.24  ,  0.2571,
        0.23  ,  0.4055,  0.43  ,  0.4328,  0.44  ,  0.297 ,  0.29  ,
        0.3757,  0.41  ,  0.3696,  0.33  ,  0.2296,  0.19  ,  0.1342,
        0.12  ,  0.149 ,  0.15  ,  0.3411,  0.3   ,  0.1815,  0.13  ,
        0.2999,  0.33  ,  0.2637,  0.27  ,  0.2273,  0.17  ,  0.1123,
        0.07  ,  0.1978,  0.18  , -0.0792, -0.11  ,  0.0034, -0.02  ,
       -0.0719, -0.07  ,  0.1641,  0.15  ,  0.0264,  0.01  , -0.0783,
       -0.09  ,  0.0372,  0.02  ,  0.0929,  0.07  , -0.0296, -0.07  ,
       -0.0131, -0.02  , -0.0227, -0.05  , -0.078 , -0.1   , -0.1495,
       -0.2   ,  0.1068,  0.06  ,  0.0888,  0.03  ,  0.0775,  0.05  ,
        0.0204, -0.02  ,  0.0596,  0.03  ,  0.1095,  0.07  ,  0.0488,
        0.04  , -0.199 , -0.2   , -0.1354, -0.15  , -0.1165, -0.13  ,
        0.0952,  0.08  ,  0.0248,  0.01  , -0.0132, -0.07  , -0.1616,
       -0.18  , -0.0568, -0.09  , -0.0487, -0.09  , -0.0477, -0.05  ,
       -0.004 , -0.04  ,  0.171 ,  0.12  ,  0.2928,  0.25  ,  0.157 ,
        0.13  ,  0.1538,  0.09  ,  0.196 ,  0.12  ,  0.0947,  0.08  ,
       -0.0139, -0.03  , -0.0288, -0.03  , -0.0157, -0.03  , -0.1134,
       -0.15  , -0.1392, -0.2   , -0.1015, -0.14  , -0.2439, -0.29  ,
       -0.1168, -0.17  , -0.0686, -0.09  , -0.1003, -0.15  , -0.2985,
       -0.36  , -0.1774, -0.21  , -0.1546, -0.21  , -0.0667, -0.1   ,
       -0.1481, -0.21  , -0.2486, -0.28  , -0.2156, -0.24  , -0.2304,
       -0.28  , -0.1485, -0.21  , -0.2105, -0.27  , -0.2055, -0.22  ,
       -0.2084, -0.26  , -0.3146, -0.4   , -0.293 , -0.34  , -0.0693,
       -0.11  , -0.1395, -0.16  , -0.3162, -0.34  , -0.3288, -0.35  ,
       -0.4332, -0.44  , -0.3789, -0.42  , -0.4261, -0.47  , -0.4396,
       -0.43  , -0.3706, -0.4   , -0.2174, -0.23  , -0.2931, -0.28  ,
       -0.4194, -0.44  , -0.3369, -0.35  , -0.2463, -0.27  , -0.1417,
       -0.15  , -0.0679, -0.09  , -0.1173, -0.16  , -0.2546, -0.28  ,
       -0.1224, -0.11  , -0.0974, -0.15  , -0.229 , -0.21  , -0.2808,
       -0.31  , -0.3212, -0.3   , -0.3062, -0.27  , -0.2532, -0.24  ,
       -0.322 , -0.37  , -0.0982, -0.12  , -0.1471, -0.2   , -0.2489,
       -0.33  , -0.2003, -0.31  , -0.2125, -0.32  , -0.2009, -0.28  ,
       -0.1424, -0.21  , -0.0648, -0.1   , -0.0628, -0.12  , -0.1148,
       -0.2   ])
```

```
model.predict([[2050]])
```

```
array([0.74928518])
```

```
m = model.coef_ # slope
m
```

```
array([0.00698723])
```

```
C = model.intercept_ # Y -intercept/constant
C
```

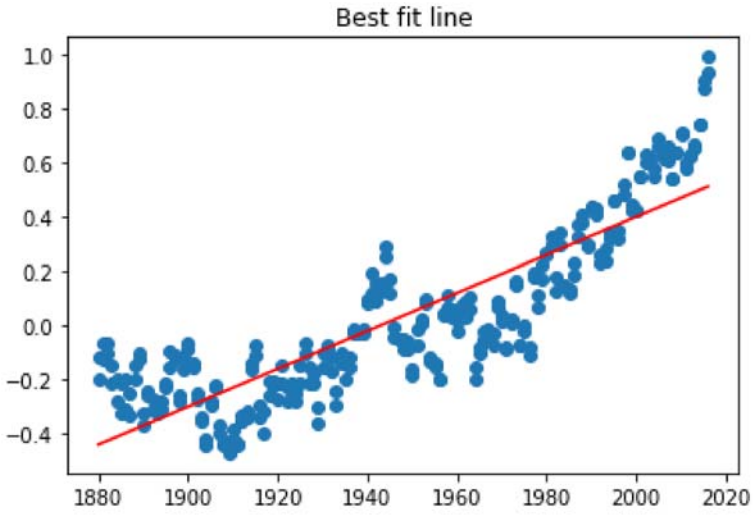
```
-13.574531559542251
```

```
m * 2050 + C
```

```
array([0.74928518])
```

```
plt.scatter(x,y)
plt.plot(x,y_pred,c = 'red')
plt.title('Best fit line')
```

```
Text(0.5, 1.0, 'Best fit line')
```



Colab paid products - Cancel contracts here

0s completed at 11:39

Multivariable regressor model :

```
import matplotlib.pyplot as plt
import numpy as np


import pandas as pd
df= pd.read_csv('/content/measures_v2.csv')
df
```

	u_q	coolant	stator_winding	u_d	stator_tooth	motor_speed
0	-0.450682	18.805172	19.086670	-0.350055	18.293219	0.002866
1	-0.325737	18.818571	19.092390	-0.305803	18.294807	0.000257
2	-0.440864	18.828770	19.089380	-0.372503	18.294094	0.002355
3	-0.327026	18.835567	19.083031	-0.316199	18.292542	0.006105
4	-0.471150	18.857033	19.082525	-0.332272	18.291428	0.003133
...	...	...	...	...	...	...
55953	27.639210	19.375916	103.863991	-128.161346	77.511452	3779.966309
55954	27.669556	19.434158	103.948357	-128.147598	77.517960	3779.971924
55955	27.710018	19.484137	103.954788	-128.152359	77.514587	3779.973389
55956	27.713411	19.495962	103.947815	-128.173584	77.707138	3779.974365
55957	27.759415	19.492443	103.937996	-128.155594	77.826485	3779.967285

55958 rows × 13 columns

```
df.head(10)
```

	u_q	coolant	stator_winding	u_d	stator_tooth	motor_speed	
0	-0.450682	18.805172	19.086670	-0.350055	18.293219	0.002866	0.00
1	-0.325737	18.818571	19.092390	-0.305803	18.294807	0.000257	0.00

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 55958 entries, 0 to 55957
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   u_q             55958 non-null  float64
 1   coolant         55958 non-null  float64
 2   stator_winding  55958 non-null  float64
 3   u_d             55958 non-null  float64
 4   stator_tooth    55958 non-null  float64
 5   motor_speed     55958 non-null  float64
 6   i_d             55958 non-null  float64
 7   i_q             55958 non-null  float64
 8   pm              55958 non-null  float64
 9   stator_yoke     55958 non-null  float64
 10  ambient         55958 non-null  float64
 11  torque          55957 non-null  float64
 12  profile_id      55957 non-null  float64
dtypes: float64(13)
memory usage: 5.6 MB
```

```
df.shape
```

```
(55958, 13)
```

```
df.size
```

```
727454
```

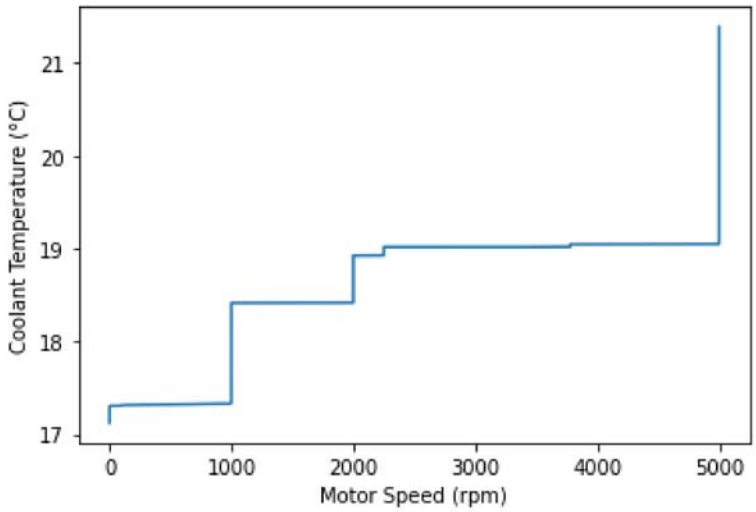
```
df.isnull().sum()
```

```
u_q               0
coolant           0
stator_winding    0
u_d               0
stator_tooth      0
motor_speed       0
i_d               0
i_q               0
pm                0
stator_yoke       0
ambient           0
torque            1
profile_id        1
dtype: int64
```

```
x= df['motor_speed'].to_numpy()
```

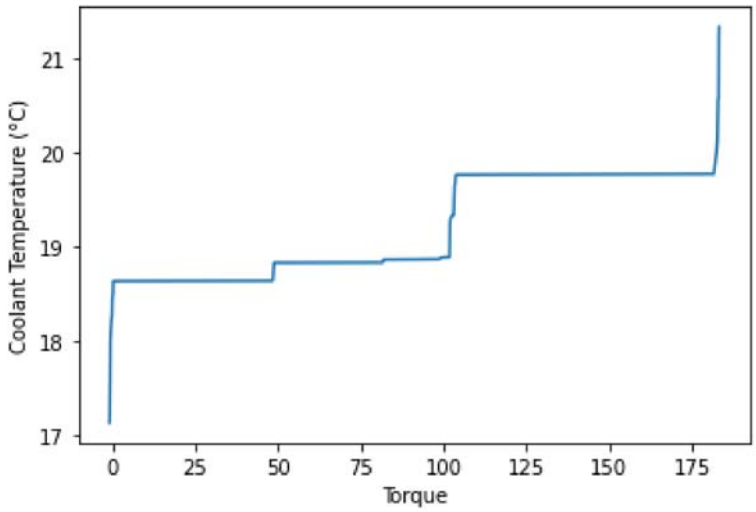
```
x= np.sort(x)
y= df['coolant'].to_numpy()
y= np.sort(y)
plt.plot(x,y)                          #Data visualisation
plt.xlabel('Motor Speed (rpm)')
plt.ylabel('Coolant Temperature (°C)')
```

```
Text(0, 0.5, 'Coolant Temperature (°C)')
```




```
x1= df['torque'].to_numpy()
x1= np.sort(x1)
y= df['coolant'].to_numpy()
y= np.sort(y)
plt.plot(x1,y)                     #Data visualisation
plt.xlabel('Torque')
plt.ylabel('Coolant Temperature (°C)')
```

```
Text(0, 0.5, 'Coolant Temperature (°C)')
```




```
#dropping rows to avoid error:- ValueError: Input contains NaN, infinity or a value too la


df.drop(df.index[300:55958], inplace=True)
df.head(10)
```

	u_q	coolant	stator_winding	u_d	stator_tooth	motor_speed	
0	-0.450682	18.805172	19.086670	-0.350055	18.293219	0.002866	0.00
1	-0.325737	18.818571	19.092390	-0.305803	18.294807	0.000257	0.00
2	-0.440864	18.828770	19.089380	-0.372503	18.294094	0.002355	0.00
3	-0.327026	18.835567	19.083031	-0.316199	18.292542	0.006105	0.00
4	-0.471150	18.857033	19.082525	-0.332272	18.291428	0.003133	-0.06
5	-0.538973	18.901548	19.077108	0.009147	18.290628	0.009636	-0.61
6	-0.653148	18.941711	19.074583	0.238890	18.292524	0.001337	-1.00
7	-0.758392	18.960861	19.082499	0.395099	18.294041	0.001422	-1.28
8	-0.727128	18.973545	19.085533	0.546623	18.291964	0.000577	-1.49
9	-0.874307	18.987812	19.076025	0.578944	18.287233	-0.001248	-1.63

```
df.shape
```

```
(300, 13)
```

```
x
```

```
array([-6.29892992e-03, -5.88588184e-03, -5.36493398e-03, ...,
        4.99996973e+03,  4.99997021e+03,  4.99997119e+03])
```

```
df = df.drop('profile_id', axis = 1)
from sklearn.model_selection import train_test_split

x = df.drop('coolant', axis = 1)
y = df.coolant

x_train, x_test, y_train, y_test = train_test_split(x,y, random_state = 0)


print(x.shape)
print(x_train.shape)
print(x_test.shape)
```

```
(300, 11)
(225, 11)
(75, 11)
```

```
print(y.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(300,)
(225,)
(75,)
```

```
#Normalization
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
x_train = scaler.fit_transform(x_train)
x_test = scaler.fit_transform(x_test)


#Running a regressor
from sklearn.linear_model import LinearRegression
model = LinearRegression()


#Model fitting
model.fit(x_train,y_train)
```

```
LinearRegression()
```

```
#Predicting output
y_pred = model.predict(x_test)
y_pred #Predicted
```

```
array([18.96252203, 18.95184183, 18.85999362, 18.99909162, 19.04984957,
       18.88716077, 19.01811353, 18.96303456, 18.90105461, 18.87546483,
       19.15014213, 18.84822248, 18.92981896, 19.05167007, 18.86370596,
       18.88812522, 19.09606112, 18.86572014, 19.02960661, 18.91629322,
       18.95852789, 18.87224081, 18.88499549, 18.88668314, 19.00160587,
       18.87457657, 18.96732726, 19.14581348, 19.06129606, 19.15509171,
       18.87101521, 18.86572782, 18.86536484, 18.85459909, 18.99860116,
       18.9382887 , 18.92170736, 18.91878194, 19.15493497, 18.86042074,
       19.01093227, 18.87317291, 18.87671409, 18.8699556 , 18.90488689,
       18.89509275, 19.06309323, 18.88102519, 18.86486819, 18.88915674,
       18.87894745, 18.98538478, 18.93088818, 18.87730552, 18.89196909,
       19.05487784, 18.93582045, 18.93547274, 19.13717678, 18.93765569,
       18.87489522, 18.91050465, 18.88400199, 19.15763827, 19.13868822,
       19.01304402, 19.05049204, 18.88462844, 18.85821581, 18.88702338,
       18.92859137, 19.02721756, 19.08121793, 18.9633951 , 18.8951246 ])
```

```
y_test #Actual values
```

```
208    19.156012
188    19.091515
12     18.990061
221    18.896414
239    18.905981
         ...
156    18.844925
228    19.179750
273    19.197226
27     18.974653
144    18.909874
Name: coolant, Length: 75, dtype: float64
```

```
print(x_train[10]) #Showing normalised values
```

```
[0.95694597 0.18584679 0.015018   0.08991301 0.99999795 0.00330179
 0.95428125 0.16649225 0.09831079 0.         0.99523854]
```

```
model.predict([x_train[10]]) #Predicting particular values
```

```
array([18.90351377])
```

Colab paid products - Cancel contracts here

✓ 0s completed at 21:09