

DAY 9:

Bottle Dataset

In [1]:

```
#to import libraries  
import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns
```

In [2]:

```
df=pd.read_csv(r"E:\Dataset\9_bottle.csv")[0:500]  
df
```

```
C:\ProgramData\Anaconda3\lib\site-packages\IPython\core\interactiveshell.p  
y:3165: DtypeWarning: Columns (47,73) have mixed types.Specify dtype optio  
n on import or set low_memory=False.  
    has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
```

Out[2]:

Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat
0	1	1	054.0 056.0 19-4903CR-HY-060-0930-05400560-0000A-3	0	10.50	33.440	NaN	25.649	NaN
1	1	2	054.0 056.0 19-4903CR-HY-060-0930-05400560-0008A-3	8	10.46	33.440	NaN	25.656	NaN
2	1	3	054.0 056.0 19-4903CR-HY-060-0930-05400560-0010A-7	10	10.46	33.437	NaN	25.654	NaN
3	1	4	054.0 056.0 19-4903CR-HY-060-0930-05400560-0019A-3	19	10.45	33.420	NaN	25.643	NaN
4	1	5	054.0 056.0 19-4903CR-HY-060-0930-05400560-0020A-7	20	10.45	33.421	NaN	25.643	NaN
...
495	16	496	063.3 058.0 19-4903CR-HY-065-1030-06330580-0700A-7	700	4.90	34.269	NaN	27.114	NaN
496	16	497	063.3 058.0 19-4903CR-HY-065-1030-06330580-0792A-3	792	4.50	34.310	NaN	27.191	NaN
497	16	498	063.3 058.0 19-4903CR-HY-065-1030-06330580-0800A-7	800	4.48	34.311	NaN	27.194	NaN
498	16	499	063.3 058.0 19-4903CR-HY-065-1030-06330580-0900A-7	900	4.21	34.319	NaN	27.230	NaN

In [31]:

```
Cst_Cnt  Btl_Cnt  Sta_ID  Depth_ID  Depthm  T_degC  Salnty  O2ml_L  STheta  O2Sat
df.info()
```

			19-4903CR-							
499	16	500	063.3 058.0	HY-065-1030-06330580-1000A-7	1000	3.95	34.329	NaN	27.265	NaN

500 rows × 74 columns

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 500 entries, 0 to 499
```

```
Data columns (total 74 columns):
```

#	Column	Non-Null Count	Dtype
0	Cst_Cnt	500 non-null	int64
1	Btl_Cnt	500 non-null	int64
2	Sta_ID	500 non-null	object
3	Depth_ID	500 non-null	object
4	Depthm	500 non-null	int64
5	T_degC	499 non-null	float64
6	Salnty	494 non-null	float64
7	O2ml_L	0 non-null	float64
8	STheta	493 non-null	float64
9	O2Sat	0 non-null	float64
10	Oxy_μmol/Kg	0 non-null	float64
11	BtlNum	0 non-null	float64
12	RecInd	500 non-null	int64
13	T_prec	499 non-null	float64
14	T_qual	4 non-null	float64
15	S_prec	494 non-null	float64
16	S_qual	10 non-null	float64
17	P_qual	500 non-null	float64
18	O_qual	500 non-null	float64
19	SThtaq	14 non-null	float64
20	O2Satq	500 non-null	float64
21	ChlorA	0 non-null	float64
22	Chlqua	500 non-null	float64
23	Phaeop	0 non-null	float64
24	Phaqua	500 non-null	float64
25	PO4uM	0 non-null	float64
26	PO4q	500 non-null	float64
27	SiO3uM	0 non-null	float64
28	SiO3qu	500 non-null	float64
29	NO2uM	0 non-null	float64
30	NO2q	500 non-null	float64
31	NO3uM	0 non-null	float64
32	NO3q	500 non-null	float64
33	NH3uM	0 non-null	float64
34	NH3q	500 non-null	float64
35	C14As1	0 non-null	float64
36	C14A1p	0 non-null	float64
37	C14A1q	500 non-null	float64
38	C14As2	0 non-null	float64
39	C14A2p	0 non-null	float64
40	C14A2q	500 non-null	float64
41	DarkAs	0 non-null	float64
42	DarkAp	0 non-null	float64
43	DarkAq	500 non-null	float64
44	MeanAs	0 non-null	float64
45	MeanAp	0 non-null	float64
46	MeanAq	500 non-null	float64
47	IncTim	0 non-null	object
48	LightP	0 non-null	float64
49	R_Depth	500 non-null	float64
50	R_TEMP	499 non-null	float64
51	R_POTEMP	495 non-null	float64
52	R_SALINITY	494 non-null	float64
53	R_SIGMA	486 non-null	float64
54	R_SVA	486 non-null	float64
55	R_DYNHT	500 non-null	float64

```

56 R_O2                0 non-null    float64
57 R_O2Sat             0 non-null    float64
58 R_SIO3              0 non-null    float64
59 R_PO4               0 non-null    float64
60 R_NO3               0 non-null    float64
61 R_NO2               0 non-null    float64
62 R_NH4               0 non-null    float64
63 R_CHLA              0 non-null    float64
64 R_PHAE0             0 non-null    float64
65 R_PRES              500 non-null  int64
66 R_SAMP              0 non-null    float64
67 DIC1                0 non-null    float64
68 DIC2                0 non-null    float64
69 TA1                 0 non-null    float64
70 TA2                 0 non-null    float64
71 pH2                 0 non-null    float64
72 pH1                 0 non-null    float64
73 DIC Quality Comment 0 non-null    object

```

dtypes: float64(65), int64(5), object(4)

memory usage: 289.2+ KB

In [4]:

```
df.columns
```

Out[4]:

```

Index(['Cst_Cnt', 'Btl_Cnt', 'Sta_ID', 'Depth_ID', 'Depthm', 'T_degC',
      'Salnty', 'O2ml_L', 'STheta', 'O2Sat', 'Oxy_μmol/Kg', 'BtlNum',
      'RecInd', 'T_prec', 'T_qual', 'S_prec', 'S_qual', 'P_qual', 'O_qual',
      'SThta', 'O2Satq', 'ChlorA', 'Chlqua', 'Phaeop', 'Phaqua', 'P04u',
      'P04q', 'SiO3uM', 'SiO3qu', 'NO2uM', 'NO2q', 'NO3uM', 'NO3q', 'NH3u',
      'NH3q', 'C14As1', 'C14A1p', 'C14A1q', 'C14As2', 'C14A2p', 'C14A2q',
      'DarkAs', 'DarkAp', 'DarkAq', 'MeanAs', 'MeanAp', 'MeanAq', 'IncTi',
      'LightP', 'R_Depth', 'R_TEMP', 'R_POTEMP', 'R_SALINITY', 'R_SIGMA',
      'R_SVA', 'R_DYNHT', 'R_O2', 'R_O2Sat', 'R_SIO3', 'R_PO4', 'R_NO3',
      'R_NO2', 'R_NH4', 'R_CHLA', 'R_PHAE0', 'R_PRES', 'R_SAMP', 'DIC1',
      'DIC2', 'TA1', 'TA2', 'pH2', 'pH1', 'DIC Quality Comment'],
      dtype='object')

```

Linear Regression

In [5]:

```

x=df[['Cst_Cnt', 'Btl_Cnt', 'Depthm', 'RecInd', 'P_qual', 'O_qual', 'O2Satq']]
y=df['R_PRES']

```

In [6]:

```
# to split my dataset into test and train data
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3)
```

In [7]:

```
from sklearn.linear_model import LinearRegression

lr=LinearRegression()
lr.fit(x_train,y_train)
```

Out[7]:

LinearRegression()

In [8]:

```
print(lr.score(x_test,y_test))
```

0.9999989086841241

In [9]:

```
lr.score(x_train,y_train)
```

Out[9]:

0.9999989178354033

Ridge Regression

In [10]:

```
from sklearn.linear_model import Ridge,Lasso
```

In [11]:

```
rr=Ridge(alpha=10)
rr.fit(x_train,y_train)
rr.score(x_test,y_test)
```

Out[11]:

0.9999987917054648

Lasso Regression

In [12]:

```
la=Lasso(alpha=10)
la.fit(x_train,y_train)
```

Out[12]:

Lasso(alpha=10)

In [13]:

```
la.score(x_test,y_test)
```

Out[13]:

0.9999986153960675

In []: