In [52]:

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split

import warnings;
warnings.simplefilter('ignore')
```

In [53]:

```
df = pd.read_csv('teams.csv')
df.head()
```

Out[53]:

	team	year	athletes	events	age	height	weight	prev_medals	medals
0	AFG	1964	8	8	22.0	161.0	64.2	0.0	0
1	AFG	1968	5	5	23.2	170.2	70.0	0.0	0
2	AFG	1972	8	8	29.0	168.3	63.8	0.0	0
3	AFG	1980	11	11	23.6	168.4	63.2	0.0	0
4	AFG	2004	5	5	18.6	170.8	64.8	0.0	0

In [54]:

```
train, test = train_test_split(df, test_size=0.3, random_state=1)
```

In [55]:

```
X=df.drop(['team','year','medals'],axis = 1)
y=df['medals']
```

In [56]:

```
X
```

Out[56]:

	athletes	events	age	height	weight	prev_medals
0	8	8	22.0	161.0	64.2	0.0
1	5	5	23.2	170.2	70.0	0.0
2	8	8	29.0	168.3	63.8	0.0
3	11	11	23.6	168.4	63.2	0.0
4	5	5	18.6	170.8	64.8	0.0
...	...	...	...	...	...	...
2009	26	19	25.0	179.0	71.1	0.0
2010	14	11	25.1	177.8	70.5	0.0
2011	16	15	26.1	171.9	63.7	3.0
2012	9	8	27.3	174.4	65.2	4.0
2013	31	13	27.5	167.8	62.2	0.0

2014 rows × 6 columns

In [57]:

```
y
```

Out[57]:

```
0       0
1       0
2       0
3       0
4       0
       ..
2009    0
2010    3
2011    4
2012    0
2013    0
Name: medals, Length: 2014, dtype: int64
```

In [58]:

```
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 [59]:

```
x_train.head()
```

Out[59]:

	athletes	events	age	height	weight	prev_medals
96	224	105	23.1	175.0	69.7	44.0
945	399	171	26.8	176.6	69.7	68.0
1204	140	71	24.1	174.3	67.8	2.0
1749	3	3	22.0	176.7	74.7	0.0
1161	49	35	26.9	172.7	64.1	3.0

In [60]:

```
x_test.head()
```

Out[60]:

	athletes	events	age	height	weight	prev_medals
773	7	7	24.9	172.1	65.0	0.0
1550	172	59	21.9	167.8	63.2	106.0
1771	6	5	26.7	166.3	58.2	0.0
1578	7	7	27.3	162.5	59.0	0.0
1916	18	16	24.3	177.7	73.4	1.0

In [61]:

```
y_train.head()
```

Out[61]:

```
96      51
945     72
1204     1
1749     0
1161     2
Name: medals, dtype: int64
```

In [62]:

```
y_test.head()
```

Out[62]:

```
773      0
1550    58
1771     0
1578     0
1916     0
Name: medals, dtype: int64
```

Linear Regression

In [63]:

```
from sklearn.model_selection import cross_val_score
from sklearn.linear_model import LinearRegression

lin_regressor=LinearRegression()
mse=cross_val_score(lin_regressor,X,y,scoring='neg_mean_squared_error',cv=5)
mean_mse=np.mean(mse)
print(mean_mse)
```

```
-145.42031462787676
```

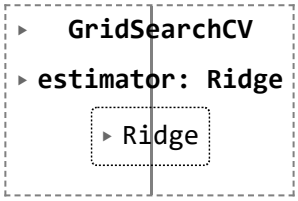
Ridge Regression

In [72]:

```
from sklearn.linear_model import Ridge

ridge=Ridge()
ridge_regressor=GridSearchCV(ridge,parameters,scoring='neg_mean_squared_error',cv=5)
ridge_regressor.fit(X,y)
```

Out[72]:



In [68]:

```
print(ridge_regressor.best_params_)
print(ridge_regressor.best_score_)
```

```
{'alpha': 1e-15}
-145.42031462787622
```

Lasso Regression

In [69]:

```
from sklearn.linear_model import Lasso
lasso=Lasso()
lasso_regressor=GridSearchCV(lasso,scoring='neg_mean_squared_error',cv=5)

lasso_regressor.fit(X,y)
print(lasso_regressor.best_params_)
print(lasso_regressor.best_score_)
```

```
{'alpha': 1e-15}
-145.4203146278768
```

In [ ]:

```

```