

CENTRAL FINAL- LINEAR REGRESSION

March 20, 2024

```
[175]: import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
from sklearn.model_selection import GridSearchCV
```

```
[176]: data = pd.read_csv("C:\\Users\\HP 840\\Desktop\\Study Performance.csv")
data
```

```
[176]:
```

	gender	math_score	reading_score	writing_score
0	female	72	72	74
1	female	69	90	88
2	female	90	95	93
3	male	47	57	44
4	male	76	78	75
..
995	female	88	99	95
996	male	62	55	55
997	female	59	71	65
998	female	68	78	77
999	female	77	86	86

[1000 rows x 4 columns]

```
[177]: data['gender'] = data['gender'].astype('category')
data['gender'] = data['gender'].cat.codes
data
```

```
[177]:
```

	gender	math_score	reading_score	writing_score
0	0	72	72	74
1	0	69	90	88
2	0	90	95	93
3	1	47	57	44
4	1	76	78	75
..
995	0	88	99	95
996	1	62	55	55

997	0	59	71	65
998	0	68	78	77
999	0	77	86	86

[1000 rows x 4 columns]

```
[178]: x = data.drop(['math_score'], axis = 1)
x
```

```
[178]:
```

	gender	reading_score	writing_score
0	0	72	74
1	0	90	88
2	0	95	93
3	1	57	44
4	1	78	75
..
995	0	99	95
996	1	55	55
997	0	71	65
998	0	78	77
999	0	86	86

[1000 rows x 3 columns]

```
[179]: y = data['math_score']
y
```

```
[179]:
```

0	72
1	69
2	90
3	47
4	76
..	
995	88
996	62
997	59
998	68
999	77

Name: math_score, Length: 1000, dtype: int64

```
[180]: x_train,x_test,y_train, y_test = train_test_split(x,y, test_size = 0.25,
↳ random_state = 42)
```

```
[181]: model = LinearRegression()
```

```
[182]: model.fit(x_train, y_train)
model
```

```
[182]: LinearRegression()
```

```
[183]: y_pred = model.predict(x_test)
       y_pred
```

```
[183]: array([75.67954669, 61.79464656, 76.6831291 , 78.72994281, 83.92981513,
            81.05313159, 70.12812537, 53.30352807, 76.6831291 , 50.98837574,
            53.33905311, 29.3590166 , 71.69661823, 54.13339791, 87.46994778,
            71.58591928, 50.87767679, 49.937108 , 47.35699628, 55.35034178,
            74.49812786, 44.83989817, 62.48632886, 38.8658778 , 83.09994529,
            80.44465965, 78.25965841, 45.89116592, 35.85100671, 51.00782788,
            63.24102478, 70.54268798, 63.35172373, 75.96395821, 77.51299894,
            46.16754099, 79.08952825, 68.18397417, 66.71814381, 5.84122804,
            80.44465965, 68.08131166, 69.93421605, 57.06505863, 83.54274111,
            61.93283409, 73.77895697, 22.32569202, 88.16163008, 71.19884525,
            76.18535612, 65.99897292, 82.49147336, 49.07171312, 64.31174466,
            65.27980204, 85.17424758, 57.68156701, 81.60662635, 81.98566393,
            47.53070885, 76.7857916 , 69.40091803, 55.73741581, 93.38899099,
            69.53910557, 61.15868604, 57.32198157, 60.71589023, 81.95817535,
            68.2194992 , 80.04954918, 59.97726721, 48.2143547 , 62.88143934,
            80.8042451 , 84.19815376, 61.44309756, 56.92687109, 52.09462066,
            65.06644058, 42.44153547, 79.08952825, 64.56063115, 74.13854241,
            56.6779846 , 68.54355961, 63.84146026, 77.7263604 , 26.45484447,
            78.95134071, 85.74719448, 62.16226845, 76.90452701, 26.77890488,
            70.01742642, 65.17713953, 59.58215673, 65.6118989 , 60.68840164,
            78.48105632, 58.76032334, 59.36879527, 61.269385 , 72.17493908,
            66.50478235, 47.14363482, 78.95134071, 70.94995876, 47.21880874,
            36.98474021, 74.82218827, 57.81975455, 53.80130105, 77.03467809,
            60.2261537 , 55.13698032, 50.57306851, 72.81089959, 82.98924634,
            62.154232 , 55.46104074, 86.86147585, 68.90314505, 60.24560583,
            68.08131166, 69.29021908, 67.10521784, 46.67335042, 74.24924137,
            72.41578912, 75.57688419, 46.67335042, 87.04734872, 62.8459143 ,
            34.55085247, 80.87941902, 82.10439933, 77.04271454, 41.18906657,
            46.63782539, 60.19062866, 63.73879776, 51.87322275, 78.86813034,
            49.04422454, 39.67555088, 75.21729874, 56.35392419, 72.63718703,
            70.72856085, 77.50496249, 65.28783849, 62.13477987, 84.77913711,
            65.05840413, 45.78046697, 62.12674342, 77.04271454, 44.83989817,
            65.86078539, 65.42602602, 71.33703279, 59.00920983, 89.65494902,
            70.34952328, 53.5879396 , 89.4892729 , 62.29241954, 69.64980452,
            84.28940058, 72.92159855, 85.27691009, 75.07107476, 57.59835664,
            25.3405631 , 56.96239612, 72.05620367, 85.2494215 , 69.37342945,
            58.033116 , 62.34814133, 88.65136661, 65.06644058, 88.29981762,
            60.47504019, 62.12674342, 63.98768425, 69.29825553, 89.15717604,
            56.24322524, 58.29003894, 69.65784097, 68.29467312, 59.03669841,
            48.2143547 , 59.00920983, 69.01384401, 66.35855837, 64.06285817,
            64.4499322 , 73.99231843, 77.97524689, 70.87478484, 64.31978111,
            24.38054218, 90.44929383, 70.00938997, 69.79602851, 46.14005241,
```

```
66.75366884, 54.88005738, 66.14519691, 64.69881869, 58.5034004 ,
41.22459161, 67.61102727, 60.4475516 , 79.21967934, 85.97662884,
72.70020064, 49.18241208, 69.98190139, 86.72328831, 33.48013259,
84.39206308, 56.45658669, 59.00117338, 74.9603758 , 63.48991127,
73.64076943, 67.00255533, 48.93352559, 58.64962439, 44.84793462,
54.13339791, 47.6414078 , 66.60744486, 56.81617214, 83.95730372,
53.89254787, 46.92223691, 85.7197059 , 48.10365575, 87.19357271,
60.46700374, 66.25589586, 83.95730372, 41.33529056, 89.65494902])
```

```
[184]: mae = mean_absolute_error(y_test,y_pred )
mae
```

```
[184]: 4.793632083170076
```

```
[185]: mse = mean_squared_error(y_test,y_pred )
mse
```

```
[185]: 37.01110485993376
```

```
[186]: r2_sc = r2_score(y_test,y_pred )
r2_sc
```

```
[186]: 0.8495628451174323
```

```
[187]: model = LinearRegression()
model
```

```
[187]: LinearRegression()
```

```
[188]: param_grid = {
    'positive':[True, False],
    'copy_X':[True, False],
    'fit_intercept':[True, False]
}
```

```
[189]: grid_search = GridSearchCV(model, param_grid, cv=5, n_jobs=-1)
grid_search.fit(x_train, y_train)
```

```
[189]: GridSearchCV(cv=5, estimator=LinearRegression(), n_jobs=-1,
               param_grid={'copy_X': [True, False],
                           'fit_intercept': [True, False],
                           'positive': [True, False]})
```

```
[190]: best_params = grid_search.best_params_
print("Best Parameters :", best_params)
```

```
Best Parameters : {'copy_X': True, 'fit_intercept': True, 'positive': True}
```

```
[191]: best_model = LinearRegression(**best_params)
best_model
```

```
[191]: LinearRegression(positive=True)
```

```
[192]: best_model.fit(x_train, y_train)
```

```
[192]: LinearRegression(positive=True)
```

```
[193]: best_model
```

```
[193]: LinearRegression(positive=True)
```

```
[194]: y_pred = best_model.predict(x_test)
y_pred
```

```
[194]: array([75.67954669, 61.79464656, 76.6831291 , 78.72994281, 83.92981513,
81.05313159, 70.12812537, 53.30352807, 76.6831291 , 50.98837574,
53.33905311, 29.3590166 , 71.69661823, 54.13339791, 87.46994778,
71.58591928, 50.87767679, 49.937108 , 47.35699628, 55.35034178,
74.49812786, 44.83989817, 62.48632886, 38.8658778 , 83.09994529,
80.44465965, 78.25965841, 45.89116592, 35.85100671, 51.00782788,
63.24102478, 70.54268798, 63.35172373, 75.96395821, 77.51299894,
46.16754099, 79.08952825, 68.18397417, 66.71814381, 5.84122804,
80.44465965, 68.08131166, 69.93421605, 57.06505863, 83.54274111,
61.93283409, 73.77895697, 22.32569202, 88.16163008, 71.19884525,
76.18535612, 65.99897292, 82.49147336, 49.07171312, 64.31174466,
65.27980204, 85.17424758, 57.68156701, 81.60662635, 81.98566393,
47.53070885, 76.7857916 , 69.40091803, 55.73741581, 93.38899099,
69.53910557, 61.15868604, 57.32198157, 60.71589023, 81.95817535,
68.2194992 , 80.04954918, 59.97726721, 48.2143547 , 62.88143934,
80.8042451 , 84.19815376, 61.44309756, 56.92687109, 52.09462066,
65.06644058, 42.44153547, 79.08952825, 64.56063115, 74.13854241,
56.6779846 , 68.54355961, 63.84146026, 77.7263604 , 26.45484447,
78.95134071, 85.74719448, 62.16226845, 76.90452701, 26.77890488,
70.01742642, 65.17713953, 59.58215673, 65.6118989 , 60.68840164,
78.48105632, 58.76032334, 59.36879527, 61.269385 , 72.17493908,
66.50478235, 47.14363482, 78.95134071, 70.94995876, 47.21880874,
36.98474021, 74.82218827, 57.81975455, 53.80130105, 77.03467809,
60.2261537 , 55.13698032, 50.57306851, 72.81089959, 82.98924634,
62.154232 , 55.46104074, 86.86147585, 68.90314505, 60.24560583,
68.08131166, 69.29021908, 67.10521784, 46.67335042, 74.24924137,
72.41578912, 75.57688419, 46.67335042, 87.04734872, 62.8459143 ,
34.55085247, 80.87941902, 82.10439933, 77.04271454, 41.18906657,
46.63782539, 60.19062866, 63.73879776, 51.87322275, 78.86813034,
49.04422454, 39.67555088, 75.21729874, 56.35392419, 72.63718703,
70.72856085, 77.50496249, 65.28783849, 62.13477987, 84.77913711,
65.05840413, 45.78046697, 62.12674342, 77.04271454, 44.83989817,
```

```

65.86078539, 65.42602602, 71.33703279, 59.00920983, 89.65494902,
70.34952328, 53.5879396 , 89.4892729 , 62.29241954, 69.64980452,
84.28940058, 72.92159855, 85.27691009, 75.07107476, 57.59835664,
25.3405631 , 56.96239612, 72.05620367, 85.2494215 , 69.37342945,
58.033116 , 62.34814133, 88.65136661, 65.06644058, 88.29981762,
60.47504019, 62.12674342, 63.98768425, 69.29825553, 89.15717604,
56.24322524, 58.29003894, 69.65784097, 68.29467312, 59.03669841,
48.2143547 , 59.00920983, 69.01384401, 66.35855837, 64.06285817,
64.4499322 , 73.99231843, 77.97524689, 70.87478484, 64.31978111,
24.38054218, 90.44929383, 70.00938997, 69.79602851, 46.14005241,
66.75366884, 54.88005738, 66.14519691, 64.69881869, 58.5034004 ,
41.22459161, 67.61102727, 60.4475516 , 79.21967934, 85.97662884,
72.70020064, 49.18241208, 69.98190139, 86.72328831, 33.48013259,
84.39206308, 56.45658669, 59.00117338, 74.9603758 , 63.48991127,
73.64076943, 67.00255533, 48.93352559, 58.64962439, 44.84793462,
54.13339791, 47.6414078 , 66.60744486, 56.81617214, 83.95730372,
53.89254787, 46.92223691, 85.7197059 , 48.10365575, 87.19357271,
60.46700374, 66.25589586, 83.95730372, 41.33529056, 89.65494902])

```

```

[195]: mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
r2_sc = r2_score(y_test, y_pred)

```

```

[196]: print(f"Mean Absolute Error = ", {mae})
print(f"Mean Squared Error = ", {mse})
print(f"R2 Score = ", {r2_sc})
print("Best Parameters :", best_params)

```

Mean Absolute Error = {4.793632083170078}

Mean Squared Error = {37.01110485993378}

R2 Score = {0.8495628451174322}

Best Parameters : {'copy_X': True, 'fit_intercept': True, 'positive': True}