Advanced Statistics

Dr. Syed Faisal Bukhari
Associate Professor
Department of Data Science
Faculty of Computing and Information Technology
University of the Punjab

Textbooks

- **Probability & Statistics for Engineers & Scientists**, Ninth Edition, Ronald E. Walpole, Raymond H. Myer

- **Elementary Statistics: Picturing the World,** 6th Edition, Ron Larson and Betsy Farber

- **Elementary Statistics**, 13th Edition, Mario F. Triola

Reference books

- **Probability and Statistical Inference, Ninth Edition,** Robert V. Hogg, Elliot A. Tanis, Dale L. Zimmerman

- **Probability Demystified**, Allan G. Bluman

- **Practical Statistics for Data Scientists: 50 Essential Concepts,** Peter Bruce and Andrew Bruce

- **Schaum's Outline of Probability,** Second Edition, Seymour Lipschutz, Marc Lipson

- **Python for Probability, Statistics, and Machine Learning,** José Unpingco

References

- **Probability & Statistics for Engineers & Scientists**, Ninth Edition, Ronald E. Walpole, Raymond H. Myer

These notes contain material from the above resource.

Prediction Interval using Python

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import linregress

# Given data
jackpot = np.array([334, 127, 300, 227, 202, 180, 164, 145,
255])
tickets = np.array([54, 16, 41, 27, 23, 18, 18, 16, 26])

# Perform linear regression
slope, intercept, r_value, p_value, std_err =
linregress(jackpot, tickets)

# Regression line equation: y = mx + b
def regression_line(x):
    return slope * x + intercept
```

```python
# Plot the data and the regression line
plt.scatter(jackpot, tickets, label='Data Points')
plt.plot(jackpot, regression_line(jackpot), color='red',
label='Regression Line')
plt.xlabel('Jackpot Amount (Millions of Dollars)')
plt.ylabel('Number of Tickets Sold (Millions)')
plt.title('Lottery Tickets Sold vs. Jackpot Amount')
plt.legend()
plt.show()

# Prediction for the jackpot amount of 625 million dollars
#i.e x_0 = 625 given
jackpot_625 = 625
predicted_tickets = regression_line(jackpot_625)

# se = sqrt((y - y_hat)^2 /(n-2))
# Compute the standard error of the estimate
se = np.sqrt(np.sum((tickets -
regression_line(jackpot))**2) / (len(jackpot) - 2))
```

```python
# Calculate the prediction interval for 95% confidence
t_value = 2.365  # for a two-tailed 95% confidence interval with 7 degrees of freedom
margin_of_error = t_value * se

lower_bound = predicted_tickets - margin_of_error
upper_bound = predicted_tickets + margin_of_error
# Output prediction interval
print(f"Predicted number of tickets for $625 million jackpot: {predicted_tickets:.2f} million")
print(f"95% Prediction Interval: ({lower_bound:.2f} million, {upper_bound:.2f} million)")
```

#Predicted number of tickets for $625 million jackpot: 97.98 million

#95% Prediction Interval: (87.49 million, 108.48 million)

Linear Regression Model Using Matrices

In fitting a **multiple linear regression model**, particularly when the **number of variables exceeds two**, a knowledge of matrix theory can facilitate the mathematical manipulations considerably. Suppose that the experimenter has ***k* independent** variables $x_1, x_2, . . . , x_k$ and ***n* observations** $y_1, y_2, . . . , y_n$, each of which can be expressed by the equation

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \ldots + \beta_k x_{ki} + \epsilon_i$$

This model essentially represents ***n* equations** describing how the **response values** are generated in the scientific process. Using matrix notation, we can write the following equation:

General Linear Model

$\mathbf{y} = \mathbf{X}\beta + \boldsymbol{\epsilon_i}$

where

$$\text{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \text{X} = \begin{bmatrix} 1 & x_{11} & x_{21} & \cdots & x_{k1} \\ 1 & x_{12} & x_{22} & \cdots & x_{k2} \\ \vdots & \vdots & \vdots & & \\ 1 & x_{1n} & x_{21} & \cdots & x_{kn} \end{bmatrix}, \beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix}, \epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

The least squares estimating equations, **$(X'\mathrm{X})\mathrm{b} = X'\mathrm{y}$** ,are

$$X'\mathrm{X} = \begin{bmatrix} \mathrm{n} & \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{2i} & \cdots & \sum_{i=1}^{n} x_{ki} \\ \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{1i}^{2} & \sum_{i=1}^{n} x_{1i}\, x_{2i} & \cdots & \sum_{i=1}^{n} x_{1i}\, x_{ki} \\ \vdots & \vdots & \vdots & & \\ \sum_{i=1}^{n} x_{ki} & \sum_{i=1}^{n} x_{ki}\, x_{1i} & \sum_{i=1}^{n} x_{ki}\, x_{2i} & \cdots & \sum_{i=1}^{n} x_{ki}^{2} \end{bmatrix}$$

$$X'\mathrm{y} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_{1i}\, y_i \\ \vdots \\ \sum_{i=1}^{n} x_{ki}\, y_i \end{bmatrix}$$

If the **matrix A is nonsingular**, we can write the **solution for the regression coefficients** as

$$\mathbf{b} = (X'\mathrm{X})^{-1} X'\mathrm{y}$$

Example: The percent survival rate of sperm in a certain type of animal semen, after storage, was measured at various combinations of concentrations of three materials used to increase chance of survival. The data are given in the below Table. Estimate the multiple linear regression model for the given data.

y (% survival)	x_1 (weight %)	x_2 (weight %)	x_3 (weight %)
25.5	1.74	5.3	10.8
31.2	6.32	5.42	9.4
25.9	6.22	8.41	7.2
38.4	10.52	4.63	8.5
18.4	1.19	11.6	9.4
26.7	1.22	5.85	9.9
26.4	4.1	6.62	8
25.9	6.32	8.72	9.1
32	4.08	4.42	8.7
25.2	4.15	7.6	9.2
39.7	10.15	4.83	9.4

$$\boldsymbol{X}'\mathbf{X} = \begin{bmatrix} \text{n} & \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{2i} & \sum_{i=1}^{n} x_{3i} \\ \sum_{i=1}^{n} x_{1i} & \sum_{i=1}^{n} x_{1i}^2 & \sum_{i=1}^{n} x_{1i}\, x_{2i} & \sum_{i=1}^{n} x_{1i}\, x_{3i} \\ \sum_{i=1}^{n} x_{2i} & \sum_{i=1}^{n} x_{1i}\, x_{2i} & \sum_{i=1}^{n} x_{2i}^2 & \sum_{i=1}^{n} x_{2i}\, x_{3i} \\ \sum_{i=1}^{n} x_{3i} & \sum_{i=1}^{n} x_{1i}\, x_{3i} & \sum_{i=1}^{n} x_{2i}\, x_{3i} & \sum_{i=1}^{n} x_{3i}^2 \end{bmatrix}$$

***k = 3* independent** variables x_1, x_2, x_3 and ***n = 11* observations** y_1, y_2,…, y_{11}

$$\boldsymbol{X}'\mathbf{X} = \begin{bmatrix} \text{n} & \sum_{i=1}^{11} x_{1i} & \sum_{i=1}^{11} x_{2i} & \sum_{i=1}^{11} x_{3i} \\ \sum_{i=1}^{11} x_{1i} & \sum_{i=1}^{11} x_{1i}^2 & \sum_{i=1}^{11} x_{1i}\, x_{2i} & \sum_{i=1}^{11} x_{1i}\, x_{3i} \\ \sum_{i=1}^{11} x_{2i} & \sum_{i=1}^{11} x_{1i}\, x_{2i} & \sum_{i=1}^{11} x_{2i}^2 & \sum_{i=1}^{11} x_{2i}\, x_{3i} \\ \sum_{i=1}^{11} x_{3i} & \sum_{i=1}^{11} x_{1i}\, x_{3i} & \sum_{i=1}^{11} x_{2i}\, x_{3i} & \sum_{i=1}^{11} x_{3i}^2 \end{bmatrix}$$

$$\boldsymbol{X}'\mathbf{y} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_{1i}\, y_i \\ \vdots \\ \sum_{i=1}^{n} x_{ki}\, y_i \end{bmatrix}$$

$$\boldsymbol{X}'\mathbf{y} = \begin{bmatrix} \sum_{i=1}^{11} y_i \\ \sum_{i=1}^{11} x_{1i}\, y_i \\ \sum_{i=1}^{11} x_{2i}\, y_i \\ \sum_{i=1}^{11} x_{3i}\, y_i \end{bmatrix}$$

.

y	x_1	x_2	x_3	x_1x_2	x_1x_3	x_2x_3	y^2	x_1^2	x_2^2	x_3^2
25.5	1.74	5.3	10.8	9.222	18.792	173.2998	650.25	3.0276	28.09	116.64
31.2	6.32	5.42	9.4	34.2544	59.408	2034.985	973.44	39.9424	29.3764	88.36
25.9	6.22	8.41	7.2	52.3102	44.784	2342.66	670.81	38.6884	70.7281	51.84
38.4	10.52	4.63	8.5	48.7076	89.42	4355.434	1474.56	110.6704	21.4369	72.25
18.4	1.19	11.6	9.4	13.804	11.186	154.4115	338.56	1.4161	134.56	88.36
26.7	1.22	5.85	9.9	7.137	12.078	86.20069	712.89	1.4884	34.2225	98.01
26.4	4.1	6.62	8	27.142	32.8	890.2576	696.96	16.81	43.8244	64
25.9	6.32	8.72	9.1	55.1104	57.512	3169.509	670.81	39.9424	76.0384	82.81
32	4.08	4.42	8.7	18.0336	35.496	640.1207	1024	16.6464	19.5364	75.69
25.2	4.15	7.6	9.2	31.54	38.18	1204.197	635.04	17.2225	57.76	84.64
39.7	10.15	4.83	9.4	49.0245	95.41	4677.428	1576.09	103.0225	23.3289	88.36
35.7	1.72	3.12	7.6	5.3664	13.072	70.14958	1274.49	2.9584	9.7344	57.76
26.5	1.7	5.3	8.2	9.01	13.94	125.5994	702.25	2.89	28.09	67.24
377.5	59.43	81.82	115.4	360.6621	522.078	728.31	11400.15	394.7255	576.7264	1035.96

.

y	x_1	x_2	x_3	x_1y	x_2y	x_3y
25.5	1.74	5.3	10.8	44.37	135.15	275.4
31.2	6.32	5.42	9.4	197.184	169.104	293.28
25.9	6.22	8.41	7.2	161.098	217.819	186.48
38.4	10.52	4.63	8.5	403.968	177.792	326.4
18.4	1.19	11.6	9.4	21.896	213.44	172.96
26.7	1.22	5.85	9.9	32.574	156.195	264.33
26.4	4.1	6.62	8	108.24	174.768	211.2
25.9	6.32	8.72	9.1	163.688	225.848	235.69
32	4.08	4.42	8.7	130.56	141.44	278.4
25.2	4.15	7.6	9.2	104.58	191.52	231.84
39.7	10.15	4.83	9.4	402.955	191.751	373.18
35.7	1.72	3.12	7.6	61.404	111.384	271.32
26.5	1.7	5.3	8.2	45.05	140.45	217.3
377.5	**59.43**	**81.82**	**115.4**	**1877.567**	**2246.661**	**3337.78**

$$X'X = \begin{bmatrix} n & \sum_{i=1}^{11} x_{1i} & \sum_{i=1}^{11} x_{2i} & \sum_{i=1}^{11} x_{3i} \\ \sum_{i=1}^{11} x_{1i} & \sum_{i=1}^{11} x_{1i}^2 & \sum_{i=1}^{11} x_{1i}\, x_{2i} & \sum_{i=1}^{11} x_{1i}\, x_{3i} \\ \sum_{i=1}^{11} x_{2i} & \sum_{i=1}^{11} x_{1i}\, x_{2i} & \sum_{i=1}^{11} x_{2i}^2 & \sum_{i=1}^{11} x_{2i}\, x_{3i} \\ \sum_{i=1}^{11} x_{3i} & \sum_{i=1}^{11} x_{1i}\, x_{3i} & \sum_{i=1}^{11} x_{2i}\, x_{3i} & \sum_{i=1}^{11} x_{3i}^2 \end{bmatrix}$$

$$\boldsymbol{X'X} = \begin{bmatrix} 11 & 59.43 & 81.82 & 115.4 \\ 59.43 & 394.7255 & 360.6621 & 522.078 \\ 81.82 & 360.6621 & 576.7264 & 728.31 \\ 115.4 & 522.078 & 728.31 & 1035.96 \end{bmatrix}$$

$$X'y = \begin{bmatrix} \sum_{i=1}^{11} y_i \\ \sum_{i=1}^{11} x_{1i}\, y_i \\ \sum_{i=1}^{11} x_{2i}\, y_i \\ \sum_{i=1}^{11} x_{3i}\, y_i \end{bmatrix}$$

$$\boldsymbol{X'y} = \begin{bmatrix} 377.5 \\ 1877.567 \\ 2246.661 \\ 3337.78 \end{bmatrix}$$

$$(X'X)^{-1}= \begin{bmatrix} 8.0648 & -0.0826 & -0.0942 & -0.7905 \\ -0.0826 & 0.0085 & 0.0017 & 0.0037 \\ -0.0826 & 0.0017 & 0.0166 & -0.0021 \\ -0.7905 & 0.0037 & -0.0021 & 0.0886 \end{bmatrix}$$

$\mathbf{b} = (X'\mathbf{X})^{-1}X'\mathbf{y}$, the **estimated regression coefficients** are obtained as

$$b = \begin{bmatrix} 8.0648 & -0.0826 & -0.0942 & -0.7905 \\ -0.0826 & 0.0085 & 0.0017 & 0.0037 \\ -0.0826 & 0.0017 & 0.0166 & -0.0021 \\ -0.7905 & 0.0037 & -0.0021 & 0.0886 \end{bmatrix}_{4\times 4} \begin{bmatrix} 377.5 \\ 1877.567 \\ 2246.661 \\ 3337.78 \end{bmatrix}_{4\times 1}$$

$$b = \begin{bmatrix} 39.1574 \\ 1.0161 \\ -1.8616 \\ -0.3433. \end{bmatrix}_{4\times 1}$$

$b_0 = 39.1574$, $b_1 = 1.0161$, $b_2 = -1.8616$, $b_3 = -0.3433$. Hence, our estimated regression equation is

$\hat{y} = 39.1574 + 1.0161x_1 - 1.8616x_2 - 0.3433x_3$

```python
import pandas as pd
import statsmodels.api as sm

# Data from the previous example
data = {
    'y': [25.5, 31.2, 25.9, 38.4, 18.4, 26.7, 26.4, 25.9, 32.0, 25.2, 39.7,
35.7, 26.5],
    'x1': [1.74, 6.32, 6.22, 10.52, 1.19, 1.22, 4.10, 6.32, 4.08, 4.15,
10.15, 1.72, 1.70],
    'x2': [5.30, 5.42, 8.41, 4.63, 11.60, 5.85, 6.62, 8.72, 4.42, 7.60,
4.83, 3.12, 5.30],
    'x3': [10.80, 9.40, 7.20, 8.50, 9.40, 9.90, 8.00, 9.10, 8.70, 9.20,
9.40, 7.60, 8.20]
}

# Create a DataFrame
df = pd.DataFrame(data)

# Add a constant term to the independent variables
X = sm.add_constant(df[['x1', 'x2', 'x3']])

# Fit the multiple linear regression model
model = sm.OLS(df['y'], X).fit()

# Display the regression results
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.912
Model:                            OLS   Adj. R-squared:                  0.882
Method:                 Least Squares   F-statistic:                     30.98
Date:                Fri, 01 Dec 2023   Prob (F-statistic):           4.50e-05
Time:                        04:55:01   Log-Likelihood:                -25.533
No. Observations:                  13   AIC:                             59.07
Df Residuals:                       9   BIC:                             61.33
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         39.1573      5.887      6.651      0.000      25.840      52.475
x1             1.0161      0.191      5.323      0.000       0.584       1.448
x2            -1.8616      0.267     -6.964      0.000      -2.466      -1.257
x3            -0.3433      0.617     -0.556      0.592      -1.739       1.053
==============================================================================
==============================================================================
Omnibus:                        2.087   Durbin-Watson:                   1.568
Prob(Omnibus):                  0.352   Jarque-Bera (JB):                1.548
Skew:                           0.730   Prob(JB):                        0.461
Kurtosis:                       2.148   Cond. No.                         123.
==============================================================================
```