Advance Data Science

October 21, 2025

Submitted by: Sucharitha Kondaparthi

Enrollment number: 2503B05108

1. Given the following data of Temperature (°C) and Power Consumption (kWh):

Temperature (°C) (X)	**Power Consumption (kWh) (Y)**
10	300
12	310
14	320
16	330
18	345
20	360
22	370
24	390
26	420
28	450

Solution:

X	Y	X^2	XY
10	300	100	3000
12	310	144	3720
14	320	196	4480
16	330	256	5280
18	345	324	6210
20	360	400	7200
22	370	484	8140
24	390	576	9360
26	420	676	10920
28	450	784	12600
$\Sigma X=190$	**$\Sigma Y=3595$**	**$\Sigma X^2=3940$**	**$\Sigma XY=70910$**

Therefore,

Number of observations, $n = 10$
$\Sigma X = 190$
$\Sigma Y = 3595$
$\Sigma XY = 70910$
$\Sigma X^2 = 3940$

$$\text{Mean} = \frac{\text{Sum of Observations}}{\text{Total number of Observations}}$$

Mean of $X = 19$
Mean of $Y = 359.5$

(a). **Derivation of Regression Equation**

$$\boxed{Y=a+bX}$$

By Using Least Squares Method,

For a simple regression equation $Y=a+bX$

$$\boxed{b = (n*\Sigma XY - \Sigma X*\Sigma Y) / (n*\Sigma X^2 - (\Sigma X)^2)}$$

Computing Numerator,

$$\text{Numerator} = n*\Sigma XY - \Sigma X*\Sigma Y$$

$$=10*70910-(190*3595)$$

$$=709100-683050$$

$$=26050$$

Computing Denominator,

$$\text{Denominator} = n*\Sigma X^2 - (\Sigma X)^2$$

$$=10*3940 – (190)^2$$

$$=39400 – 36100$$

$$=3300$$

b = numerator/ denominator

= 26050/3300

=7.893939

$$\boxed{a = (\text{mean of } Y) – b*(\text{Mean of } X)}$$

=359.5 – (7.893939)*19
=359.5 – 149.984841
=209.515159

Therefore the regression equation is

$$\boxed{Y = (209.515159) + (7.893939)X}$$

(b). Computation of R^2

X (°C)	Y (Actual)	$\hat{Y} = 208.7879 + 7.9848X$	$(Y-\hat{Y})$	$(Y-\hat{Y})^2$
10	300	288.64	11.36	129.1
12	310	304.61	5.39	29.0
14	320	320.58	-0.58	0.34
16	330	336.55	-6.55	42.9
18	345	352.52	-7.52	56.5
20	360	368.50	-8.50	72.3
22	370	384.47	-14.47	209.5
24	390	400.44	-10.44	108.9
26	420	416.41	3.59	12.9
28	450	432.38	17.62	310.5
				$\sum(Y-Y^\wedge)^2$**=971.9**

Sum of Squares (Residual)

$SS_{res} = \sum(Y-Y^\wedge)^2$

$= 971.9$

Total Sum Of Squares

$SS_{tot} = \sum(Y-\bar{Y})^2$

$=21530.7$

$R^2 = 1 - (SS_{res}/SS_{tot})$

$=1 - 0.0451$

$= 0.9549$

Therefore,

$\boxed{R^2 = 0.9549 = 0.955\text{(approx.)}}$

2.

(a) Use Python (statsmodels) to fit model and compare.

(b) Interpret results (positive/negative slope, accuracy).

Regression Equation

$$\hat{Y} = 209.5152 + 7.8939X$$

Using stats model the findings are:

Code:

```
import pandas as pd
import statsmodels.api as sm
# Given data
temperature = [10, 12, 14, 16, 18, 20, 22, 24, 26, 28]
power = [300, 310, 320, 330, 345, 360, 370, 390, 420, 450]
# Create DataFrame
df = pd.DataFrame({'Temperature': temperature, 'Power_Consumption': power})
# Define dependent and independent variables
X = df['Temperature']
Y = df['Power_Consumption']
# Add constant term for intercept
X = sm.add_constant(X)
# Build the model
model = sm.OLS(Y, X).fit()
# Display the summary
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:      Power_Consumption   R-squared:                       0.955
Model:                            OLS   Adj. R-squared:                  0.950
Method:                 Least Squares   F-statistic:                     171.6
Date:                Wed, 22 Oct 2025   Prob (F-statistic):           1.10e-06
Time:                        18:46:21   Log-Likelihood:                -37.005
No. Observations:                  10   AIC:                             78.01
Df Residuals:                       8   BIC:                             78.61
Df Model:                           1
Covariance Type:            nonrobust
===============================================================================
                  coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------
const         209.5152     11.962     17.515      0.000     181.931     237.100
Temperature     7.8939      0.603     13.099      0.000       6.504       9.284
==============================================================================
Omnibus:                        1.026   Durbin-Watson:                   0.581
Prob(Omnibus):                  0.599   Jarque-Bera (JB):                0.781
Skew:                           0.568   Prob(JB):                        0.677
Kurtosis:                       2.236   Cond. No.                         68.7
==============================================================================
```

Key findings are:

- The Fitted Linear model has positive slope which indicates, higher temperatures are associated with the higher power consumption.
- The regression Equation is :

$$\hat{Y} = 209.5152 + 7.8939X$$

- R2 = 0.955 which indicates the proportion of variance in Y explained by X
- Therefore, the slope is positive which means power consumption increases with temperature in this dataset

3.Using Python, perform Linear Regression on the dataset attached in excel format.

```
# Step 1: Import necessary libraries

import pandas as pd

import numpy as np

from sklearn.linear_model import LinearRegression

from sklearn.metrics import mean_absolute_error, mean_squared_error

import matplotlib.pyplot as plt
```

```
# Step 2: Load dataset
# 🔷 Replace with your actual file path
file_path = r"/content/drive/MyDrive/ADS/ASS_1/Experience_Salary.xlsx"
df = pd.read_excel(file_path)

# Step 3: Separate variables
X = df[['Experience_Years']]   # Independent variable
Y = df['Salary_USD']           # Dependent variable

# Step 4: Create and train model
model = LinearRegression()
model.fit(X, Y)

# Step 5: Get regression parameters
a = model.intercept_
b = model.coef_[0]

print(f"Intercept (a): {a:.2f}")
print(f"Slope (b): {b:.2f}")

# Step 6: Predictions
Y_pred = model.predict(X)

# Step 7: Model accuracy (R²)
r2 = model.score(X, Y)
print(f"R² (Coefficient of Determination): {r2:.4f}")

# Step 8: Error metrics
mae = mean_absolute_error(Y, Y_pred)
```

```
mse = mean_squared_error(Y, Y_pred)
rmse = np.sqrt(mse)

print(f"Mean Absolute Error (MAE): {mae:.2f}")
print(f"Mean Squared Error (MSE): {mse:.2f}")
print(f"Root Mean Squared Error (RMSE): {rmse:.2f}")

# Step 9: Add predictions and residuals to dataframe
df['Predicted_Salary'] = Y_pred
df['Residuals'] = Y - Y_pred

# Step 10: Plot Regression Line
plt.figure(figsize=(7,5))
plt.scatter(X, Y, color='blue', label='Actual Data')
plt.plot(X, Y_pred, color='red', label='Regression Line')
plt.xlabel('Experience (Years)')
plt.ylabel('Salary (USD)')
plt.title('Experience vs Salary (Linear Regression)')
plt.legend()
plt.show()

# Step 11: Residual vs Fitted (Predicted) Plot
plt.figure(figsize=(7,5))
plt.scatter(Y_pred, df['Residuals'], color='purple')
plt.axhline(y=0, color='black', linestyle='--')
plt.xlabel('Predicted (Fitted) Values')
plt.ylabel('Residuals (Y - Ŷ)')
plt.title('Residuals vs Predicted Values')
plt.show()
```

Output :

Intercept (a): 31180.95

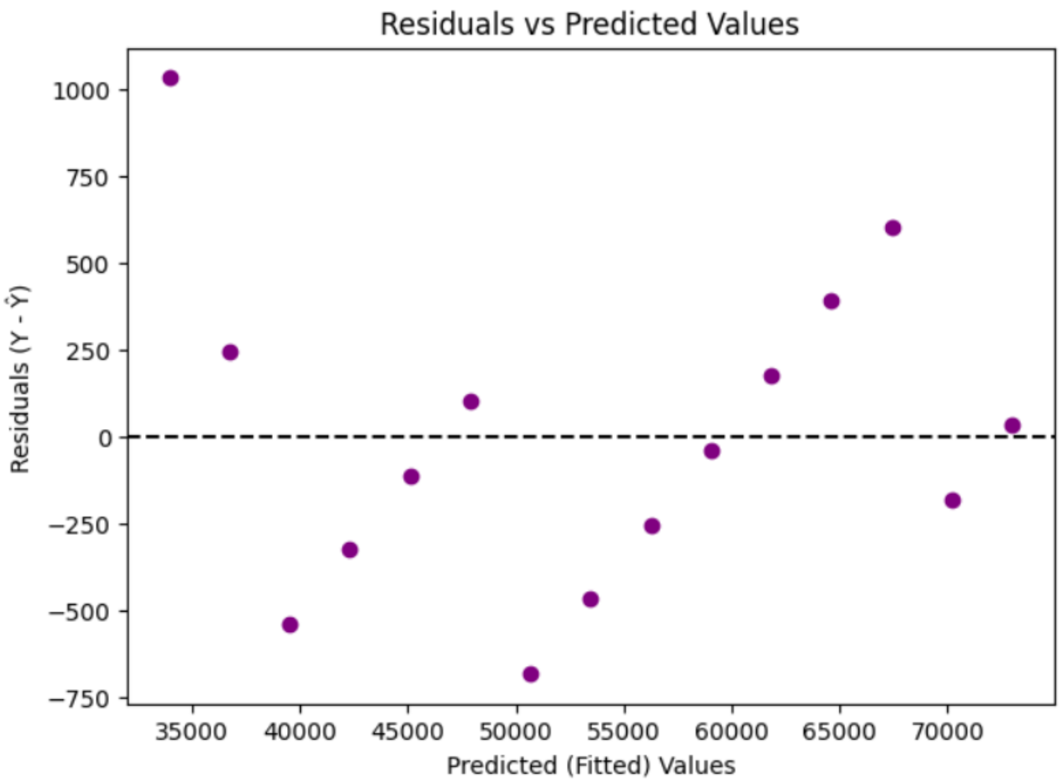
Slope (b): 2785.71

R^2 (Coefficient of Determination): 0.9987

Mean Absolute Error (MAE): 345.40

Mean Squared Error (MSE): 191746.03

Root Mean Squared Error (RMSE): 437.89


Residuals vs Predicted Values
Residuals (Y - Ŷ)
Predicted (Fitted) Values
1000
750
500
250
0
−250
−500
−750
35000
40000
45000
50000
55000
60000
65000
70000