

**BAKU HIGHER OIL SCHOOL**

**INFORMATION TECHNOLOGY DEPARTMENT**

**Information Security**

**Machine Learning Fall 2025**

**HOME ASSIGNMENT**

**REPORT**

**STUDENT: Huseyn Abdullayev**

**GROUP NUMBER: CS 25.1**

**LECTURER: Leyla Muradkhanli**

**Date: 05.10.2025**

The current report describes how the sales of a business can be predicted using linear regression on advertisement expenditure. Data presented encompasses advertising costs, sales value over the past 5 years. The idea is to use this information to approximate the expected sales when the company intends to spend 200 on adverts in the year 2025. Linear regression is a supervised machine learning algorithm to model the correlation between a dependent variable (here the sales) and an independent variable(s) (here, the advertisement spending). The model presumes that the two variables are related in a linear fashion, where increase in advertisement money will lead to increase in sales in a comparable way.

The mathematical form of the regression line is:

$$y = mx + b$$

where y is the predicted sales, x is the advertisement amount, m is the slope of the line (how much sales change when advertisement increases by one unit), and b is the intercept (the predicted sales when advertisement is zero).

The Python implementation used scikit-learn library. The data of the advert was put in a NumPys array X, and the sales values in another array y. Since scikit-learn wants the input data to be in 2D format, which can be given directly or be reshaped with `.reshape (-1, 1)`.

```
task.py
1  import numpy as np
2  from sklearn.linear_model import LinearRegression
3
4  # X must be 2D for scikit-learn: one column per feature
5  X = np.array([[90], [120], [150], [100], [130]]) # Ad spend
6  y = np.array([1000, 1300, 1800, 1200, 1380])      # Revenue
7
```

`LinearRegression()` model was then built and trained with the `fit` method. The model estimated the slope and the intercept automatically after training. After fitting the model, they predicted the sales given an advertisement of \$200 by calling `model.predict([[200]])`.

```
# Create and train the model
model = LinearRegression()
model.fit(X, y)

# Model parameters
print("Slope (m):", model.coef_)
print("Intercept (b):", model.intercept_)

# Predict new values
X_new = np.array([[200]]) # 2025 ad spends
y_pred = model.predict(X_new)
print("Predicted revenue:", y_pred.round(decimals=0))
```

The prediction output indicates the analyzed sales value given the trend learnt on past data.

```
Slope (m): [11.9122807]
Intercept (b): -69.6491228070181
Predicted revenue: [2313.]
```

A scatter diagram of the initial data and the regression line can also be used to visualize the line of best fit and as can be seen, the model successfully describes the relationship between the advertisement spending and sales.

