

Task 1

Linear regression is a mathematical technique which we use to show relationship between depended variable and one or more independent variables by fitting a linear equation to observed data. The model will assume that there is already an existing linear relationship between these variables, and the depended variable is a function of the independed variables, and we use parameters to minimize the difference between observed and predicted values.

Linear equation can be expressed as: $Y = B_0 + B_1X + e$, where

Y is the depended variable

X is the independed variable

B_0 is the y intercept

B_1 is the slope

And e is the error.

This model is used in many areas, such as predictions, estimations, and analysis between multiple variables.

Practical Example

For example, linear aggression can be used for a retail company, where you are interested to understand the relationship between the amount spent on ads and how it affects the company's sales. We would need to collect data for some period of time, preferably for a long period of time to receive better results. Next, we would do a model fitting and interpret the data to understand the relationship, that is, how much the spending affects the sales revenue. Lastly, we would do a prediction, where we analyze the fitting model to do predictions for future, such as next year or five years.

```
import numpy as np
#data
ad_price = np.array([100, 200, 300, 400, 500]).reshape(-1, 1)
sales = np.array([1500, 2500, 3300, 4200, 5000])
model = LinearRegression()
X_train, X_test, y_train, y_test = train_test_split(advertising_expenditures, sales_revenue,
test_size=0.2, random_state=42)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
interception = model.intercept_
slope = model.coef_[0]

#predict future results
ad_price_700 = 700
```

```
sales_prediction_700 = model.predict([[ad_price_700]])
ad_price_5k = 5000
sales_prediction_5k = model.predict([[ad_price_5k]])
print("interception is:",interception,"slope is: ",slope,"predicted sales for 700$
is",int(sales_prediction_700[0]),"predicted sales for 5000$ is",int(sales_prediction_5k[0]))
```

```
interception is: 639.9999999999991 slope is: 8.800000000000002 predicted sales for 700$ is 6800 predicted sales for 5000$ is 44640
```

In [24]:

Based on the outputs, we can see that the interception is ~640, slope is 8.8, and the predicted sales for 700\$ is 6800 units, and for 5000\$ of ads is 44640 units.