**Machine Learning**

Lecture # 2

**Linear Regression**

Linear Regression is a fundamental concept in machine learning and statistics. It is used to model the relationship between a dependent variable (also known as the target or response) and one or more independent variables (also known as features or predictors). The primary objective of linear regression is to predict the value of the dependent variable by fitting a linear equation to the observed data.

**1. Definition of Linear Regression**

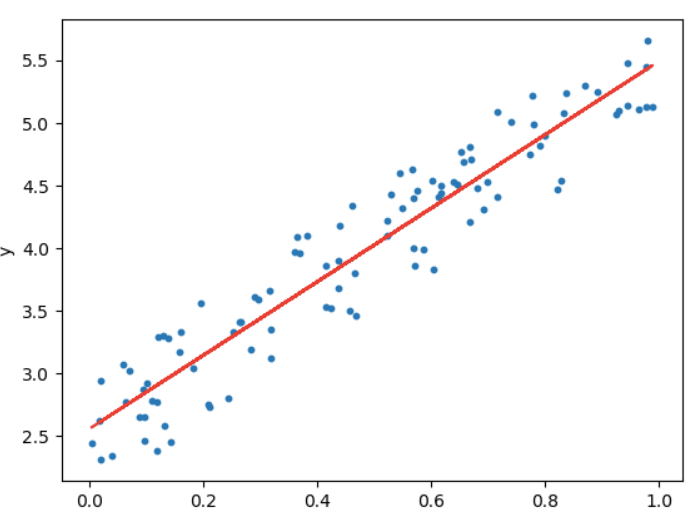
Linear regression assumes that the relationship between the dependent variable y and the independent variable x can be described by a linear equation of the form:

y=θ₀+ θ₁x

Where:

* y: Dependent variable (what we want to predict).
* x: Independent variable (input or feature).
* θ₀​: y-intercept (the value of y when x=0).
* θ₁​: Slope or coefficient (how much y changes when x changes).
* ​θ₀ and θ₁ are the parameters to be learned from the data.

In simple linear regression, there's only one independent variable, while in multiple linear regression, there are several independent variables, and the equation expands accordingly.



**2. Fundamental Concepts in Linear Regression**

To understand linear regression thoroughly, we need to break down its fundamental concepts:

**2.1 Linearity Assumption**

Linear regression assumes that the relationship between the dependent and independent variables is linear. This means the change in the dependent variable is proportional to the change in the independent variable.

**Example**: If you are predicting a person’s salary based on their years of experience, linear regression assumes that a small increase in experience leads to a proportional increase in salary.

**2.2 Parameters (θ₀ and θ₁​)**

The parameters θ₀ and θ₁​ are coefficients that determine the behavior of the regression line. The goal is to find the values of these parameters that minimize the difference between the predicted value (y^) and the actual observed value (y).

* θ₀**​**: This is the intercept of the line, the point where the line crosses the y-axis when x=0. It represents the predicted value of y when all independent variables are zero.
* θ₁**​**: This is the slope, representing how much the dependent variable y changes for each unit change in the independent variable x.

**2.3 Residuals**

The residual is the difference between the actual observed value and the predicted value:

Residual= y−y^

Where y^​ is the predicted value from the model.

* A good linear regression model will have small residuals, meaning the predicted values are close to the actual values.

**2.4 Line of Best Fit**

The line of best fit is the linear equation that minimizes the residuals, meaning it is the line that best represents the data. The objective of linear regression is to find the line of best fit by minimizing the cost function.

**3. Hypothesis Function**

In machine learning, the hypothesis function is used to predict the output based on input features. For linear regression, the hypothesis function is the same as the linear equation:

y^=hθ(x)=θ₀+θ₁x

Where:

* y ^​: Predicted value.
* hθ(x): Hypothesis function.
* θ₀​ and θ₁​: Parameters or coefficients of the linear model.
* x: Input variable (independent variable).

**Goal**: The goal is to find the best values for θ₀​ and θ₁​ so that the hypothesis function hθ(x) accurately predicts y.

**4. Univariate Linear Regression**

Univariate linear regression refers to linear regression with only one independent variable. The term "univariate" means that there is only one feature x predicting the output y.

The equation for univariate linear regression is:

y=θ₀+θ₁​x

Where:

* x is the single independent variable.
* y is the dependent variable.
* θ₀​ is the intercept, and θ₁​ is the slope or coefficient.

**Example**: Suppose you want to predict the price of a house based on its size (in square feet). If x is the size of the house and y is the price, univariate linear regression will find the best straight line that predicts house prices from their sizes.

**Steps in Univariate Linear Regression:**

**1. Data Collection**

The first step in univariate linear regression is gathering data. The data should include:

* **One independent variable (x)**: This is the predictor variable used to predict the outcome. For example, in predicting house prices, the independent variable could be the size of the house in square feet.
* **One dependent variable (y)**: This is the target variable or the outcome we want to predict. In the house price example, the dependent variable would be the house price.

**Key considerations during data collection:**

* Ensure that the data is relevant and representative of the problem you're trying to solve.
* Larger datasets generally lead to better models, as they allow the model to capture more variation.

**2. Data Preprocessing**

Before applying the linear regression model, the data must be cleaned and prepared. Common preprocessing steps include:

* **Handling missing values**: If the dataset has missing data points, these need to be addressed. Common approaches include removing rows with missing data or imputing missing values (e.g., replacing them with the mean or median).
* **Outlier detection**: Outliers can disproportionately influence the regression line, leading to inaccurate predictions. Outliers should either be removed or carefully examined for their impact.
* **Feature scaling**: Though not always necessary for univariate linear regression, it’s good practice to scale or normalize the independent variable, especially if it has a wide range of values. This can help improve the model's performance and speed up the optimization process.

**3. Define the Hypothesis Function**

The next step is to define the hypothesis function, which represents the linear relationship between the independent variable and the dependent variable.

For univariate linear regression, the hypothesis function is:

y^ = θ₀+θ₁​x

This equation defines the straight line we are trying to fit through the data points in the dataset.

**4. Initialize Model Parameters**

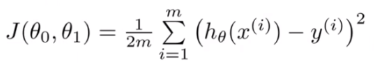
The parameters θ₀ (intercept) and θ₁ (slope) need to be initialized before training the model. Typically, they are initialized to zero or small random values.

* θ₀ (Intercept): This parameter shifts the line up or down on the y-axis.
* θ₁ (Slope): This parameter controls the steepness or incline of the line, dictating how changes in x impact y.

The goal of training is to find the optimal values for these parameters that minimize the error in the model’s predictions.

**5. Minimize the Cost Function**

The cost function measures how well the hypothesis function fits the data. In linear regression, the most commonly used cost function is the Mean Squared Error (MSE). The cost function for univariate linear regression is given by:



Where:

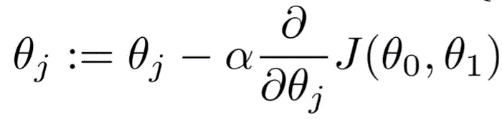
* J(θ₀,θ₁): Cost function (we aim to minimize this).
* m: Number of training examples.
* hθ(x(i)): Predicted value for the i-th example (using the hypothesis function).
* y(i): Actual value for the iii-th example.

The goal of linear regression is to find the values of θ₀​ and θ₁​ that minimize this cost function, which essentially minimizes the difference between the predicted and actual values. To find the optimal parameters (θ₀ and θ₁), we minimize the cost function, which measures how well the model fits the training data.

**6. Gradient Descent**

Gradient descent is a method used to minimize the cost function by iteratively adjusting the model's parameters (θ₀​ and θ₁​) in the direction of the steepest descent (i.e., towards the minimum cost).

**Update Rule**:



Where:

* θj (for j=0,1) are the parameters being updated.
* α is the learning rate, which determines the size of the steps taken toward the minimum.
* ∂/∂θj J(θ₀,θ₁​) is the derivative of the cost function with respect to θj ​.

The process continues until the cost function converges to a minimum, meaning that the algorithm has found the optimal parameters for the regression model.

#### 7. Model Training

Once the gradient descent algorithm has run through multiple iterations, the model parameters (θ₀ and θ₁) will converge to their optimal values, minimizing the cost function.

* The line of best fit is now established, and the model can make predictions for new data points.
* During training, you can monitor the cost function to ensure it’s decreasing as expected. If it oscillates or diverges, you may need to adjust the learning rate or other parameters.

**8. Make Predictions**

After training the model and finding the best-fit line, you can use the model to make predictions for new, unseen data points. The prediction is made using the hypothesis function:

y^ = θ₀+θ₁​x

Given a new value of **x**, the model will compute the predicted value of **y** using the learned parameters.

**9. Model Evaluation**

To assess the performance of the model, you need to evaluate how well it predicts the dependent variable on unseen data. If the model performs well on the training data but poorly on unseen data, it may be overfitting, and additional techniques like regularization may be needed.

**Example of Univariate Linear Regression:**

Let's say you're trying to predict a student's final exam score based on their hours of study. You collect data on hours studied (independent variable x) and final exam scores (dependent variable y) for several students. Using univariate linear regression, you can model the relationship between hours of study and exam scores as a straight line:

y^=θ₀+θ₁

Where θ₀​ is the intercept (predicted exam score when no hours are studied) and θ₁​ is the slope (increase in score per additional hour of study).

By minimizing the cost function (using Gradient Descent), the algorithm will find the best values for θ₀​ and θ₁​, allowing you to make predictions for new students based on their study hours.