Demystifying Linear Regression: Part I

Rishabh Iyer

March 19, 2024

1 Introduction to Regression Problems

Regression analysis predicts the relationship between a dependent variable (target or label) and one or more independent variables (features). It's widely used for predicting a continuous quantity, such as stock prices, house prices, or temperatures, based on various features. Linear Regression is a subclass of regression problems where the relationship between the target (label) and features is linear, i.e., the label is a linear function of the features. Figure 1 illustrates a simple dataset where x-axis is the feature and the y-axis is the label. The goal of linear regression is to find a straight line (amongst multiple possible straight lines) that best fits the dataset.

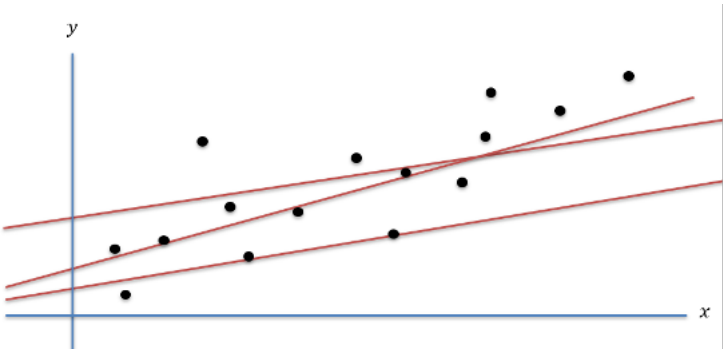

Figure 1: Illustrating Linear Regression: The goal is to fit the best straight line on a dataset. The x-axis is a single feature and the y-axis is the label.

1.1 Applications of Regression

Linear regression models find extensive applications across various fields where the relationship between a dependent variable and one or more independent variables needs to be modeled. Here are some elaborated examples:

- **Stock Price Prediction:** Financial analysts use linear regression to predict future stock prices by considering a range of features such as historical

stock prices, trading volume, economic indicators (e.g., interest rates, inflation rates), and market sentiment indicators. For instance, the model might predict the stock price of a company based on its earnings per share (EPS) and the price-to-earnings (P/E) ratio over the past quarters, allowing investors to make informed decisions.

- **House Price Prediction:** Real estate companies and investors use linear regression to estimate the market value of properties. Features might include the property's square footage, the number of bedrooms and bathrooms, the age of the house, proximity to schools and amenities, and neighborhood crime rates. By analyzing these factors, the model can help in setting competitive house prices and identifying undervalued properties.

- **Sales Forecasting:** Businesses often use linear regression to forecast sales based on factors such as advertising spend across different media channels, seasonal trends, and economic conditions. For example, a retail company might predict monthly sales using variables like advertising budget, the number of new store openings, and the unemployment rate, enabling more effective budgeting and strategic planning.

- **Energy Consumption Analysis:** Utility companies can apply linear regression to predict energy consumption patterns of households or industries based on historical consumption data, weather conditions, and pricing policies. Such models are crucial for optimizing energy production, determining pricing strategies, and planning for future energy demands.

- **Healthcare:** In medical research, linear regression models are used to understand the relationship between various risk factors and health outcomes. For example, a study might explore how different lifestyle factors (such as diet, physical activity, and smoking) influence blood pressure levels or the risk of developing certain diseases, aiding in the development of targeted health interventions.

These applications demonstrate the versatility of linear regression in extracting meaningful insights from data, aiding decision-making processes across different domains.

2 A Deep Dive into Linear Regression

TODO: Reference blog article 2 that discusses the five step recipie for creating ML models.

Linear Regression is a cornerstone of machine learning, offering a simple yet effective approach to regression. Let's explore its components in detail, using housing price prediction as an illustrative example.

2.1 Step 1: Data Preparation and Gathering

In linear regression, x represents the features (in d dimensions) or independent variables influencing the prediction, while y (a real number) denotes the target or dependent variable we aim to predict.

In the context of house price prediction, x would represent the feature vector comprising various attributes of a house: $x = [x_1, x_2, x_3, x_4, \ldots, x_d]^T$ could represent:

- x_1: Square footage of the house
- x_2: Number of bedrooms
- x_3: Number of bathrooms
- x_4: Age of the house in years
- ...
- x_d: Distance to the nearest school

Next, we collect a training dataset comprising of a large number of x, y pairs:

$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), (x^{(3)}, y^{(3)}), \cdots, (x^{(M)}, y^{(M)})\}$$

M is the total number of training examples in the dataset. Taking the example of house price prediction, we have M homes in our training dataset of recently sold homes (within a specific pre-determined period). For example, we might want to consider all homes sold in the last one month.

2.2 Step 2: Hypothesis Function

The hypothesis function for linear regression is:

$$h_{w,b}(x) = w^T x + b = \sum_{i=1}^{d} w_i x_i + b$$

Here, w is the weight vector that assigns importance to each feature. The bias term b accounts for the base price independent of the features. Specifically $w = [w_1, w_2, w_3, w_4, \ldots, w_d]^T$ contains the weights or coefficients that quantify the impact of each feature on the house's price. Similarly, b is the bias term, representing the base price of a house when all features are zero (which is more of a theoretical construct since features like square footage cannot be zero).

In the context of house price prediction, the hypothesis function models the relationship between the house's characteristics (features) and its selling price (target variable). For instance:

$$\begin{aligned} h_{w,b}(x) =& w_1 \cdot (\text{Square footage}) + w_2 \cdot (\text{Number of bedrooms}) \\ &+ w_3 \cdot (\text{Number of bathrooms}) + \ldots + w_d \cdot (\text{Distance to school}) + b \end{aligned}$$

This model allows us to input the characteristics of a house and predict its selling price based on the learned weights w and bias b, which are adjusted during the training process to minimize prediction errors.

2.3 Step 3: The Loss Function

The Mean Squared Error (MSE) loss function is a critical component in many machine learning models, including linear regression. It is mathematically defined as:

$$L(w, b) = \frac{1}{2M} \sum_{i=1}^{M} (h_{w,b}(x^{(i)}) - y^{(i)})^2$$

where:

- $L(w, b)$ is the loss function value, representing the "cost" or "error" of the model predictions.
- M is the number of training examples.
- $h_{w,b}(x^{(i)})$ is the predicted value for the i-th example, given by the hypothesis function.
- $y^{(i)}$ is the actual value (target) for the i-th example.
- The factor of $\frac{1}{2}$ is often included for mathematical convenience, simplifying derivative calculations.

The MSE loss function measures the average of the squares of the errors between the predicted and actual values. In the context of house price prediction:

- Each term $(h_{w,b}(x^{(i)}) - y^{(i)})^2$ represents the squared error for a single house's predicted price versus its actual selling price. Squaring ensures that errors are positive and emphasizes larger errors more than smaller ones, as the square of a larger number is much greater than the square of a smaller number.
- By averaging these squared errors across all training examples, the MSE provides a single measure of model performance. The lower the MSE, the closer the model's predictions are to the actual prices, indicating a better fit to the data.

The reasons why MSE makes sense as a loss function are:

- **Differentiability:** The MSE function is smooth and differentiable. This property is crucial because it allows the use of optimization algorithms like gradient descent to find the model parameters (w and b) that minimize the MSE.

- **Interpretability:** The MSE has a clear interpretation as the "average squared prediction error," making it easy to understand and communicate.
- **Sensitivity to Outliers:** Squaring the errors means that larger discrepancies between predicted and actual values have a disproportionately large effect on the MSE. This sensitivity can be both an advantage and a disadvantage, depending on the context. In scenarios where it's critical to avoid large errors (e.g., predicting house prices where overestimations or underestimations can have significant financial implications), the MSE's emphasis on larger errors helps to fine-tune the model to reduce these occurrences.

In summary, the MSE loss function is a foundational component of linear regression models, offering a balance between mathematical convenience, interpretability, and a strong incentive to minimize large errors in predictions.

2.4 Step 4: Optimization Algorithm

To minimize the Mean Squared Error (MSE) in linear regression, we employ the gradient descent algorithm. This iterative optimization technique updates the model's parameters, w and b, to reduce the MSE loss function's value progressively.

The gradient descent algorithm updates the parameters w and b as follows:

1. Initialize w and b with random values or zeros.
2. Compute the gradient of the loss function $L(w, b)$ with respect to each parameter.
3. Update each parameter by subtracting the product of the learning rate α and its gradient:
$$w := w - \alpha \nabla_w L$$
$$b := b - \alpha \nabla_b L$$
4. Repeat steps 2 and 3 until the loss function converges to a minimum or a predefined number of iterations is reached.

2.4.1 Computing the Gradient Using the Chain Rule

TODO: Refer to the section on computing gradients and basics of derivatives/gradients.

The gradient of the MSE loss function with respect to the parameters w and b can be computed using the chain rule, a fundamental technique in calculus that allows the differentiation of composite functions.

Given the MSE loss function for linear regression:

$$L(w, b) = \frac{1}{2M} \sum_{i=1}^{M} (h_{w,b}(x^{(i)}) - y^{(i)})^2$$

where $h_{w,b}(x^{(i)}) = w^T x^{(i)} + b$, the gradients are:

1. Gradient with respect to w:

$$\nabla_w L = \frac{1}{M} \sum_{i=1}^{M} (h_{w,b}(x^{(i)}) - y^{(i)}) x^{(i)}$$

2. Gradient with respect to b:

$$\nabla_b L = \frac{1}{M} \sum_{i=1}^{M} (h_{w,b}(x^{(i)}) - y^{(i)})$$

These gradients represent the slope of the loss function with respect to each parameter and guide how w and b should be adjusted to minimize the loss. The chain rule enables the decomposition of the derivative of the loss function into simpler parts, facilitating the computation of these gradients.

By iteratively applying these updates, gradient descent seeks to find the values of w and b that minimize the MSE, thereby training the linear regression model to fit the data as closely as possible.

2.5 Evaluation Metrics

Evaluation metrics are crucial for assessing the performance of regression models. Two of the most commonly used metrics are the Root Mean Square Error (RMSE) and the Coefficient of Determination (R^2).

- **Mean Absolute Error (MAE):** The MAE measures the average magnitude of the errors between the predicted values and the actual values, without considering their direction. It is defined as:

$$\text{MAE} = \frac{1}{M} \sum_{i=1}^{M} |h_{w,b}(x^{(i)}) - y^{(i)}|$$

  Intuition: MAE provides a straightforward measure of prediction accuracy by averaging the absolute differences between predicted and actual values. Unlike RMSE, MAE does not square the errors, resulting in a more linear penalization of errors.

  What MAE Tells Us: MAE offers a clear representation of the average error magnitude, making it easy to interpret. It is particularly useful when you want to understand the error scale directly without the squaring effect present in RMSE.

  Ideal Value: The ideal value for MAE is 0, indicating no prediction error. Similar to RMSE, what constitutes a "good" MAE value depends on the context of the problem and the scale of the target variable.

- **Root Mean Square Error (RMSE):** The RMSE measures the average magnitude of the errors between the predicted values by the model and the actual values from the data. It is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(h_{w,b}(x^{(i)}) - y^{(i)})^2}$$

  Intuition: RMSE gives us a sense of how large the errors are being made by our model on average. A smaller RMSE value indicates a better fit to the data, as it means the predictions are closer to the actual values.

  What RMSE Tells Us: RMSE quantifies the model's prediction error in the same units as the target variable, making it straightforward to interpret. It penalizes larger errors more than smaller ones due to the squaring of the errors.

  Ideal Value: The ideal value of RMSE is 0, which would indicate perfect predictions with no errors. However, in practice, a "good" RMSE value depends on the context and the scale of the target variable.

- **Coefficient of Determination (R^2):** The R^2 metric provides a measure of how well the observed outcomes are replicated by the model, based on the proportion of total variation of outcomes explained by the model. It is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{M}(y^{(i)} - h_{w,b}(x^{(i)}))^2}{\sum_{i=1}^{M}(y^{(i)} - \bar{y})^2}$$

  where $\bar{y}$ is the mean of observed values.

  Intuition: R^2 tells us the proportion of the variance in the dependent variable that is predictable from the independent variables. A higher R^2 value indicates that the model explains a large portion of the variance in the target variable.

  What R^2 Tells Us: R^2 is a relative measure of fit; it provides insight into the amount of variance captured by the model. It helps in comparing the explanatory power of regression models.

  Ideal Value: The ideal value of R^2 is 1, indicating that the model perfectly explains the variance in the target variable. An R^2 value of 0 means the model does not explain any of the variance, while a negative R^2 value indicates a model that performs worse than a horizontal line representing the mean of the target variable.

Each of these metrics – MAE, RMSE, and R^2 provide unique insights into the model's performance. MAE is particularly valuable for its robustness to outliers and direct interpretability, making it a critical metric alongside the other two. RMSE provides a measure of the error magnitude and R^2 indicating the proportion of variance explained by the model. Striving for a low RMSE and MAE and a high R^2 value is generally desirable, but the context and specific

objectives of the modeling task should guide the interpretation and importance of these metrics.

3 Conclusion

Linear regression is a fundamental technique in machine learning, offering a simple yet powerful tool for predicting continuous variables. Understanding its underlying principles, from hypothesis functions and loss functions to optimization algorithms and evaluation metrics, is essential for effectively applying this method in practice. In the Part II of Linear Regression, we will discuss extensions of linear regression (polynomial regression, beyond the square loss, and regularization).