AIFS ML Lecture 4: Machine Learning Basics

Suraj Narayanan Sasikumar

Hessian AI Labs

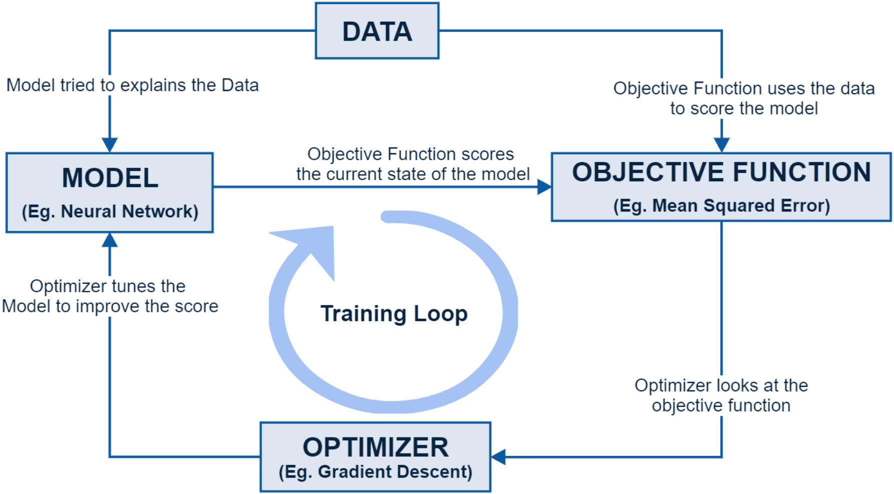
April 26, 2020

Overview

Table of Contents

Contents

Recap - Training Loop



Recap - Learning Styles

- Based on Supervision
  - **Supervised:** The dataset provides a supervisory signal in the form of a label.
  - **Unsupervised:** No explicit prediction or inference, the goal of the algorithm is to extract patterns about the underlying data generating process.
  - **Reinforcement:** The science of sequential decision making for artificial agents, such that the decisions taken by the agent in some environment maximizes a notion of cumulative reward.
- Based on Mode of Consuming Data
  - **Online:** During each iteration of the training loop, the learning algorithm processes a small subset of the dataset called a mini-batch. This allows the algorithm to learn from new data on-the-fly.
  - **Batch:** During each iteration of the training loop, the learning algorithm processes the entire dataset. These algorithms cannot learn incrementally from new data.

Recap - Learning Styles

- Based on the Type of Model
  - **Non-parametric Model:** Memory-based models whose structure is determined from the data rather than specified apriori. They cannot be parameterized by a finite number of parameters.
  - **Parametric Model:** An explicit mathematical model with finite adaptable parameters for modelling the data generating process. The model is trained to fit the data by adjusting the parameter values.

Table of Contents

Linear Regression

- Linear Regression is a machine learning algorithm that finds the best line (or planes/hyper-planes in higher dimensions) that fits the data.

Linear Regression

- Linear Regression is a machine learning algorithm that finds the best line (or planes/hyper-planes in higher dimensions) that fits the data.
- The goal of linear regression is to find the model that best predicts output label, y, given input vector, $\mathbf{x}$, using the dataset $\boldsymbol{X}$

Linear Regression

- Linear Regression is a machine learning algorithm that finds the best line (or planes/hyper-planes in higher dimensions) that fits the data.
- The goal of linear regression is to find the model that best predicts output label, y, given input vector, $\mathbf{x}$, using the dataset $\boldsymbol{X}$
- Components of the Linear Regression algorithm

Linear Regression

- Linear Regression is a machine learning algorithm that finds the best line (or planes/hyper-planes in higher dimensions) that fits the data.
- The goal of linear regression is to find the model that best predicts output label, y, given input vector, $\mathbf{x}$, using the dataset $\boldsymbol{X}$
- Components of the Linear Regression algorithm
  - **Data:** $\boldsymbol{X}$

Linear Regression

- Linear Regression is a machine learning algorithm that finds the best line (or planes/hyper-planes in higher dimensions) that fits the data.
- The goal of linear regression is to find the model that best predicts output label, y, given input vector, $\mathbf{x}$, using the dataset $\boldsymbol{X}$
- Components of the Linear Regression algorithm
  - **Data:** $\boldsymbol{X}$
  - **Model:** $\hat{y} = \mathbf{w}^T\mathbf{x} + w_0 x_0$, where $x_0 = 1$ and w_0 is the y-intercept.

Linear Regression

- Linear Regression is a machine learning algorithm that finds the best line (or planes/hyper-planes in higher dimensions) that fits the data.
- The goal of linear regression is to find the model that best predicts output label, y, given input vector, $\mathbf{x}$, using the dataset $\boldsymbol{X}$
- Components of the Linear Regression algorithm
  - **Data:** $\boldsymbol{X}$
  - **Model:** $\hat{y} = \mathbf{w}^T\mathbf{x} + w_0 x_0$, where $x_0 = 1$ and w_0 is the y-intercept.
  - **Loss Function:** Minimize Mean Squared Error

$$\text{MSE} = \frac{1}{N}||\hat{\mathbf{y}} - \mathbf{y}||_2^2$$

Linear Regression

- Linear Regression is a machine learning algorithm that finds the best line (or planes/hyper-planes in higher dimensions) that fits the data.
- The goal of linear regression is to find the model that best predicts output label, y, given input vector, $\mathbf{x}$, using the dataset $\boldsymbol{X}$
- Components of the Linear Regression algorithm
  - **Data:** $\boldsymbol{X}$
  - **Model:** $\hat{y} = \mathbf{w}^T\mathbf{x} + w_0 x_0$, where $x_0 = 1$ and w_0 is the y-intercept.
  - **Loss Function:** Minimize Mean Squared Error

$$\text{MSE} = \frac{1}{N}||\hat{\mathbf{y}} - \mathbf{y}||_2^2$$

  - **Optimizer:** Closed Form Solution

$$\mathbf{w}^* = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\mathbf{y}$$

Data

- Each record (data-point) in the dataset is represented as a row in a matrix $\boldsymbol{X}$
- If there is N records each having D features, the matrix $\boldsymbol{X}$ has shape $N \times D$

$$\boldsymbol{X} = \begin{bmatrix} \leftarrow & \mathbf{x}_1 & \rightarrow \\ \leftarrow & \mathbf{x}_2 & \rightarrow \\ & \vdots & \\ \leftarrow & \mathbf{x}_N & \rightarrow \end{bmatrix}_{N \times D}$$

- Each record $\mathbf{x}_i$ has D elements, which corresponds to the D features of a data-point.

Model

- As the name suggests, *linear* regression uses a model that is both linear in the parameters and inputs.

$$\hat{y} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \ldots w_D x_D + b$$

Model

- As the name suggests, *linear* regression uses a model that is both linear in the parameters and inputs.

$$\hat{y} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \ldots w_D x_D + b$$

- The vector $\mathbf{w} = (w_1, \ldots, w_D)$ is the adaptable parameters of the model.

Model

- As the name suggests, *linear* regression uses a model that is both linear in the parameters and inputs.

$$\hat{y} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \ldots w_D x_D + b$$

- The vector $\mathbf{w} = (w_1, \ldots, w_D)$ is the adaptable parameters of the model.
- The vector $\mathbf{x} = (x_1, \ldots, x_D)$ is the input to the model, ie a single data-point.

Model

- As the name suggests, *linear* regression uses a model that is both linear in the parameters and inputs.

$$\hat{y} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \dots w_D x_D + b$$

- The vector $\mathbf{w} = (w_1, \dots, w_D)$ is the adaptable parameters of the model.
- The vector $\mathbf{x} = (x_1, \dots, x_D)$ is the input to the model, ie a single data-point.
- The term b is the y-intercept, or often called as the bias.

Model

- As the name suggests, *linear* regression uses a model that is both linear in the parameters and inputs.

$$\hat{y} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \dots w_D x_D + b$$

- The vector $\mathbf{w} = (w_1, \dots, w_D)$ is the adaptable parameters of the model.
- The vector $\mathbf{x} = (x_1, \dots, x_D)$ is the input to the model, ie a single data-point.
- The term b is the y-intercept, or often called as the bias.
- If b is written as $w_0 x_0$ where x_0 is set to 1, then:

$$\hat{y} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \dots w_D x_D + w_0 x_0$$

Or more succinctly,

$$\hat{y} = \mathbf{w}^T \mathbf{x}$$

Where,

$\mathbf{x} = (x_1, \dots, x_D, x_0)$

$\mathbf{w} = (w_1, \dots, w_D, w_0)$

Loss Function

- When we use **mean squared-error** (MSE) to measure the loss function, linear regression is called ordinary least squares (OLS).
- The "error" in Mean squared-error is the Euclidean distance between the true label and the prediction of the model. For data-point $\mathbf{x}_i$ the prediction is $\hat{y}_i = \mathbf{w}^T\mathbf{x}_i$, so the error is:

$$\hat{y}_i - y_i$$

  Where, y_i is the true label (supervisor signal) from the dataset.
- So the mean of all the squared-errors is given by,

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)$$

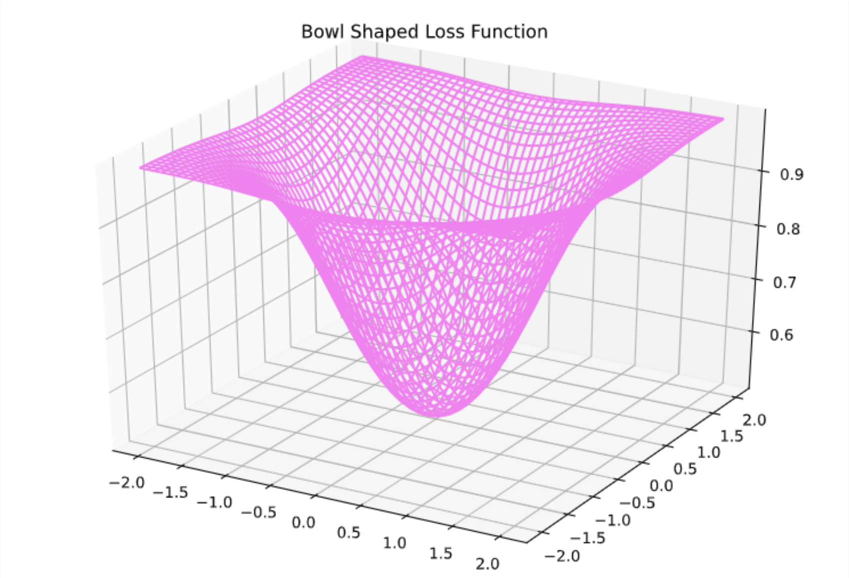
$$\text{MSE} = \frac{1}{N}||\hat{\mathbf{y}} - \mathbf{y}||_2^2$$

  Where,

  $\hat{\mathbf{y}} = (\hat{y}_1, \ldots, \hat{y}_N)$

  $\mathbf{y} = (y_1, \ldots, y_N)$

Loss function Illustration



Optimizer

- The goal of the optimizer is to set a value to **w** such that the loss is a minimum a possible. But how to find the minimum of the mean square-error (MSE) loss function?

Optimizer

- The goal of the optimizer is to set a value to $\mathbf{w}$ such that the loss is a minimum a possible. But how to find the minimum of the mean square-error (MSE) loss function?
- Calculus tell us that at the minimum of a function, its gradient is zero. So all we need to do is solve for $\mathbf{w}$ when the gradient of MSE loss function is zero,

$$\nabla_{\mathbf{w}} \frac{1}{N} ||\hat{\mathbf{y}} - \mathbf{y}||_2^2 = 0$$

Optimizer

- The goal of the optimizer is to set a value to **w** such that the loss is a minimum a possible. But how to find the minimum of the mean square-error (MSE) loss function?
- Calculus tell us that at the minimum of a function, its gradient is zero. So all we need to do is solve for **w** when the gradient of MSE loss function is zero,

$$\nabla_{\mathbf{w}} \frac{1}{N} ||\hat{\mathbf{y}} - \mathbf{y}||_2^2 = 0$$

- When you solve the equation, we'll get,

$$\mathbf{w}^\star = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \mathbf{y}$$

Optimizer

- The goal of the optimizer is to set a value to **w** such that the loss is a minimum a possible. But how to find the minimum of the mean square-error (MSE) loss function?
- Calculus tell us that at the minimum of a function, its gradient is zero. So all we need to do is solve for **w** when the gradient of MSE loss function is zero,

$$\nabla_{\mathbf{w}} \frac{1}{N} ||\hat{\mathbf{y}} - \mathbf{y}||_2^2 = 0$$

- When you solve the equation, we'll get,

$$\mathbf{w}^\star = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \mathbf{y}$$

- Putting $\mathbf{w}^\star$ in the model gives you the linear regression solution.

$$\hat{y} = {\mathbf{w}^\star}^T \mathbf{x}$$