Linear Models
Advanced Machine Learning

Yadder Aceituno

October 29, 2025

1 Content of Course

1. Linear or Logistic

2. Decision Trees

3. SVM

4. Neural Networks

5. Representational Issues

6. Deep Learning

7. Gaussian Processes

2 Lecture 1: Linear Regression

2.1 Linear Regression

Given a set of data points $< x_i, y_i >$, the goal is to find the weights θ such that $X_i\theta = Y_i$.
Where:

$$X = \langle 1, x_1, x_2, \ldots, x_m \rangle$$
$$Y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_m x_m$$

2.2 Maximum Likelihood Estimation

Assuming that data points are independent and identically distributed, the goal is to find the parameters θ that maximize the likelihood of the data with the next equation:

$$\theta = argmax_{\theta_0, \theta_1, \ldots, \theta_m} \prod_{i=1}^{N} p(y_i | x_i; \theta)$$

Where

$$p(y_i | x_i; \theta) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - \mu_i)^2}{2\sigma^2}}$$
$$\mu_i = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_m x_m$$

If we assume that the data points are independent and identically distributed, then the probability of observing all data points is the product of their individual probabilities. This is the joint likelihood. This method can derive into the Mean Squared Error (MSE). This can be proven as follows:

Because it's equivalent to maximize the likelihood, we can maximize the log-likelihood:

$$\theta = argmax_{\theta_0,\theta_1,\ldots,\theta_m} \log \left(\prod_{i=1}^{N} p(y_i|x_i;\theta) \right)$$

Because the logaritm of a product is the sum of the logaritms, we can rewrite the equation as:

$$\theta = argmax_{\theta_0,\theta_1,\ldots,\theta_m} \sum_{i=1}^{N} \log p(y_i|x_i;\theta)$$

Further we can transform the product in the logaritm into a sum of logaritms:

$$\theta = argmax_{\theta_0,\theta_1,\ldots,\theta_m} \sum_{i=1}^{N} \log \left(\frac{1}{\sqrt{2\pi\sigma^2}} \right) + \log \left(e^{-\frac{(y_i-\mu_i)^2}{2\sigma^2}} \right)$$

Here, the first term in the sum is a constant that doesn't depend on θ, so it can be ignored. In the second term, the logaritm of the exponential is the exponential itself, so we can rewrite the equation as:

$$\theta = argmax_{\theta_0,\theta_1,\ldots,\theta_m} \sum_{i=1}^{N} -\frac{(y_i-\mu_i)^2}{2\sigma^2}$$

Which is the same as minimizing the Mean Squared Error (MSE).

2.3 Normal Equation

Method to find the optimal parameters. The equation is: $\theta = (X^T X)^{-1} X^T y$ Minimaze the new cost function: $J(\theta) = \frac{1}{2m}(X\theta - y)^T(X\theta - y)$

2.4 Gradient Descent

Method to find the optimal parameters. Start with random parameters and iteratively update them to minimize the cost function. Keep changing the parameters until the cost function is minimized. The equation to repeat until convergence is:

$$\theta_j := \theta_j - \alpha\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x_i) - y_i)x_{ij}$$

Where α is the learning rate. And x_{ij} is the j-th feature of the i-th training example. There's an strategy where the learning rate can start with a high value and then decrease over time.

Some strategies:

- Batch Gradient Descent

- Stochastic Gradient Descent (SGD)

- Mini-batch Gradient Descent

2.5 Learning rate

If the learning rate is too small, the algorithm will take a long time to converge. If the learning rate is too large, the algorithm may not converge.

2.6 Overfitting

High number of features means high number of heights

- Reduce the number of features

- Regularization

3   Logistic Regression

Cast a binary classification problem as a regression problem. It uses the sigmoid function to map the output to a probability.

Sigmoid function: $\sigma(z) = \frac{1}{1+e^{-z}}$

Cost function:

$$Cost(h_\theta(x), y) = -y \log(h_\theta(x)) - (1 - y) \log(1 - h_\theta(x))$$

4   Regression as a shallow network

Same math as modern

Summary: - Linear Regression: $h_\theta(x) = \theta_0 + \theta_1 x$ - Logistic Regression: $h_\theta(x) = \sigma(\theta_0 + \theta_1 x)$ - Softmax regression vs k-binary classifiers

Lecture 2

Bias per layer or per input (neuron)

Random for weights is important because if no the neurons will behave the same.

* element wise product (operation between two matrices of the same size)

Hopfield Network - Binary Units - Symmetric Connections - Energy Function