CSC 246 Machine Learning — Lecture 2

Linear Regression

Lecturer: Jiaming Liang — *September 3, 2026*

1 Linear Regression Model

The linear regression model assumes that the output variable y (a scalar) can be described as an affine combination of the p input variables $x_1, x_2, \ldots, x_p$ plus a noise term ε,

$$y = f(x) + \varepsilon = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_p x_p + \varepsilon = \begin{bmatrix} \theta_0 & \theta_1 & \ldots & \theta_p \end{bmatrix} \begin{bmatrix} 1 \\ x_1 \\ \vdots \\ x_p \end{bmatrix} + \varepsilon = \boldsymbol{\theta}^\top \mathbf{x} + \varepsilon.$$

We collect the training data, which consists of n data point with inputs $\mathbf{x}_i$ and outputs y_i, in the $n \times (p+1)$ matrix $\mathbf{X}$ and n-dimensional vector $\mathbf{y}$,

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix}, \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \text{ where each } \mathbf{x}_i = \begin{bmatrix} 1 \\ x_{i1} \\ x_{i2} \\ \vdots \\ x_{ip} \end{bmatrix}.$$

Note that columns of X are features and rows of X are samples. We can then conveniently write the linear regression model in a compact form

$$\mathbf{y} = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\epsilon},$$

where $\boldsymbol{\epsilon}$ is a vector of errors/noise terms. Moreover, we can also define a vector of predicted outputs for the training data $\widehat{\mathbf{y}} = \begin{bmatrix} \widehat{y}(\mathbf{x}_1) & \widehat{y}(\mathbf{x}_2) & \ldots & \widehat{y}(\mathbf{x}_n) \end{bmatrix}^\top$, which also allows a compact matrix formulation,

$$\widehat{\mathbf{y}} = \mathbf{X}\boldsymbol{\theta}.$$

Given training data $\mathcal{T} = \{\mathbf{x}_i, y_i\}_{i=1}^n$, our goal is to train a linear regression model, that is to learn $\boldsymbol{\theta}$ from $\mathcal{T}$ such that the estimate $\widehat{\mathbf{y}}$ based on the model is "close" to the lable $\mathbf{y}$. Suppose we have a loss function $L(\hat{y}, y)$ that measures the "closeness" between $\hat{y}$ and y, then we can describe the "training" problem as searching for the $\boldsymbol{\theta}$ that minimized the overall loss over training data $\mathcal{T}$. Mathematically, this can be formulated as minimizing the cost function $J(\boldsymbol{\theta})$

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \left\{ J(\boldsymbol{\theta}) = \frac{1}{n} \sum_{i=1}^{n} L\left(\widehat{y}\left(\mathbf{x}_i; \boldsymbol{\theta}\right), y_i\right) \right\}.$$

2 Least Squares and Normal Equation

For linear regression, a commonly used loss function is the squared error loss

$$L(\widehat{y}(\mathbf{x};\boldsymbol{\theta}), y) = (\widehat{y}(\mathbf{x};\boldsymbol{\theta}) - y)^2.$$

Then, we easily have the cost function

$$J(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}\left(\widehat{y}\left(\mathbf{x}_i;\boldsymbol{\theta}\right) - y_i\right)^2 = \frac{1}{n}\|\widehat{\mathbf{y}} - \mathbf{y}\|_2^2 = \frac{1}{n}\|\mathbf{X}\boldsymbol{\theta} - \mathbf{y}\|_2^2 = \frac{1}{n}\|\boldsymbol{\epsilon}\|_2^2.$$

When using the squared error loss for learning a linear regression model from $\mathcal{T}$, we thus need to solve the *least squares* problem

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{\theta}^\top \mathbf{x}_i - y_i\right)^2 = \arg\min_{\boldsymbol{\theta}} \frac{1}{n}\|\mathbf{X}\boldsymbol{\theta} - \mathbf{y}\|_2^2.$$

To solve for $\widehat{\boldsymbol{\theta}}$, we take the gradient of the cost function and set it to zero (derivation in class)

$$\nabla L(\boldsymbol{\theta}) = \mathbf{X}^\top(\mathbf{X}\boldsymbol{\theta} - \mathbf{y}) = 0.$$

Rearranging the terms, we arrive at the *normal equation*

$$\mathbf{X}^\top\mathbf{X}\widehat{\boldsymbol{\theta}} = \mathbf{X}^\top\mathbf{y}.$$

The solution $\widehat{\boldsymbol{\theta}}$ to the least squares problem, i.e., the parameter we learn from data $\mathcal{T}$ is the solution to the above linear equation. If $\mathbf{X}^\top\mathbf{X}$ is invertible, then $\widehat{\boldsymbol{\theta}}$ has the closed form expression

$$\widehat{\boldsymbol{\theta}} = \left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{y}.$$

From a linear algebra point of view, this can be seen as the problem of finding the closest vector to $\mathbf{y}$ (in an Euclidean sense) in the subspace of $\mathbb{R}^n$ spanned by the columns of $\mathbf{X}$. The solution to this problem is the orthogonal projection of $\mathbf{y}$ onto this subspace, and the corresponding $\widehat{\boldsymbol{\theta}}$ can be shown to fulfill the normal equation. Indeed, the residual $\mathbf{r} = \mathbf{y} - \widehat{\mathbf{y}} = \mathbf{y} - \mathbf{X}\widehat{\boldsymbol{\theta}}$ should be orthogonal to the span of X, that is

$$\mathbf{X}^\top\mathbf{r} = 0,$$

which is clearly equivalent to the normal equation.

Question: What does it mean in practice if $\mathbf{X}^\top\mathbf{X}$ is not invertible?

Since $\text{rank}(\mathbf{X}^\top\mathbf{X}) = \text{rank}(\mathbf{X})$ (left as a HW problem), $\mathbf{X}^\top\mathbf{X}$ is not invertible if and only if the columns of $\mathbf{X}$ (i.e., feature vectors) are linearly dependent. In practice, this means that there is redundancy among the features. For example,

- two features are identical;

- one feature is a linear combination of other features;
- there are more features than samples, i.e., $p + 1 > n$.

For instance, suppose

$$x_3 = x_1 + x_2.$$

Then

$$\theta_1 x_1 + \theta_2 x_2 + \theta_3 x_3 = (\theta_1 + \theta_3)x_1 + (\theta_2 + \theta_3)x_2.$$

Therefore, different choices of $\boldsymbol{\theta}$ can give exactly the same prediction.

Importantly, $\mathbf{X}^\top \mathbf{X}$ being non-invertible does not mean that the least-squares problem has no solution. Rather, the solution may not be unique. The least-squares objective

$$J(\boldsymbol{\theta}) = \frac{1}{n}\|\mathbf{X}\boldsymbol{\theta} - \mathbf{y}\|_2^2$$

is still convex. Its Hessian is

$$\nabla^2 J(\boldsymbol{\theta}) = \frac{2}{n}\mathbf{X}^\top \mathbf{X}.$$

If $\mathbf{X}^\top \mathbf{X}$ is singular, some eigenvalues of the Hessian are zero, so the objective is not strongly convex. In particular, if

$$\mathbf{v} \in \text{null}(\mathbf{X}),$$

then $\mathbf{X}\mathbf{v} = 0$, and hence

$$J(\boldsymbol{\theta} + t\mathbf{v}) = J(\boldsymbol{\theta}) \qquad \text{for any } t \in \mathbb{R}.$$

Thus, the objective is flat along directions in the null space of $\mathbf{X}$.

One common way to address this issue is ridge regression, which adds an ℓ_2 regularization term:

$$\min_{\boldsymbol{\theta}} \frac{1}{n}\|\mathbf{X}\boldsymbol{\theta} - \mathbf{y}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_2^2, \qquad \lambda > 0.$$

The Hessian then becomes

$$\frac{2}{n}\mathbf{X}^\top \mathbf{X} + 2\lambda \mathbf{I},$$

which is positive definite. Therefore, the problem becomes strongly convex and has a unique solution. We will come back to the idea of regularization in the next lecture.

3 Maximum Likelihood Perspective

The least squares formulation can also be derived from a statistical perspective using *maximum likelihood estimation* (MLE).

Assume the linear regression model

$$y_i = \boldsymbol{\theta}^\top \mathbf{x}_i + \varepsilon_i, \qquad \varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2).$$

Then

$$y_i \mid \mathbf{x}_i, \boldsymbol{\theta} \sim \mathcal{N}\left(\boldsymbol{\theta}^\top \mathbf{x}_i, \sigma^2\right).$$

Since the noise terms are independent, the likelihood of the observed training outputs is

$$p(\mathbf{y} \mid \mathbf{X}; \boldsymbol{\theta}) = \prod_{i=1}^{n} p(y_i \mid \mathbf{x}_i; \boldsymbol{\theta}).$$

Therefore,

$$p(\mathbf{y} \mid \mathbf{X}; \boldsymbol{\theta}) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2\right).$$

MLE chooses the parameter that makes the observed data most likely:

$$\widehat{\boldsymbol{\theta}}_{\text{MLE}} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{y} \mid \mathbf{X}; \boldsymbol{\theta}).$$

Taking the negative logarithm,

$$-\log p(\mathbf{y} \mid \mathbf{X}; \boldsymbol{\theta}) = \frac{1}{2\sigma^2}\|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|_2^2 + \text{constant}.$$

Hence,

$$\widehat{\boldsymbol{\theta}}_{\text{MLE}} = \arg\min_{\boldsymbol{\theta}} \|\mathbf{X}\boldsymbol{\theta} - \mathbf{y}\|_2^2.$$

Note that this is the same optimization problem as in the least squares problem up to a constant. Thus, under the Gaussian noise assumption,

$$\text{MLE} \quad \Longleftrightarrow \quad \text{least squares}.$$

In this sense, the squared error loss is not just an arbitrary choice: it corresponds to assuming Gaussian noise in the linear regression model. Different assumptions on the noise distribution lead to different loss functions.