

02-620 Week 2

Machine Learning for Scientists

Aidan Jan

January 26, 2026

Univariate Linear Regression (continued)

Data

Data for univariate linear regression is

- N data points with $D = 1$ features: $x_i \in \mathbb{R}$ (or vector $x \in \mathbb{R}^N$)
- Labels: $y_i \in \mathbb{R}$ (or vector $y \in \mathbb{R}^N$)

Example: GWAS

- Analyze one SNP at a time.
- Is the SNP associated with the phenotype?

Genotype / features is a $N \times D$ matrix. N individuals and D SNPs.

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1D} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{ND} \end{bmatrix}$$

Phenotype / labels

$$y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$$

- x_{ij} is the number of mutations (alternative allele) at individual i and SNP j . E.g., when reference/alternative alleles are A/T, then 0 if AA, 1 if AT, 2 if TT.
- Phenotypes are trait values or disease status. E.g., BRCA gene expression level, blood pressure, insulin level, if having breast cancer, etc.

Model

- Linear model:

$$Y_i = \theta_1 x_i + \theta_0 + \epsilon_i, \quad i = 1, \dots, N$$

- Assumptions:

- θ, x_i are fixed (deterministic)
- $\epsilon_i \sim N(0, \sigma^2)$, IID across samples / individuals
- Parameters of interest: $\theta = (\theta_0, \theta_1)$

- Additional parameter: σ^2

Remark:

- Y_i is random due to ϵ_i , which represents random noise.
- $Y_i \sim N(\theta_1 x_i + \theta_0, \sigma^2)$, IID across samples / individuals

Terminology:

- Y_i : Dependent variable / output
- x_i : Explanatory variable / predictor / covariate / input
- θ_0 : Intercept
- θ_1 : Regression coefficient / slope
- ϵ_i : Noise

Our goal is to find values for θ_0 and θ_1 such that the line

$$Y_i = \theta_1 x_{ij} + \theta_0 + \epsilon_i$$

describes our data points the best.

In our GWAS example, we can decide the effect SNP j has on the phenotype based on the slope. If

- $\theta_1 = 0$, the line of best fit is flat, and therefore SNP j has no effect on phenotype.
- $\theta_1 > 0$, SNP j has positive effect on the phenotype.
- $\theta_1 < 0$, SNP j has negative effect on the phenotype.

Note that for regression, Y_i must be a continuous metric, meanwhile x_i can be continuous or discrete.

Learning

Estimate parameters $\hat{\theta} = (\hat{\theta}_0, \hat{\theta}_1)$ from data (x, y) .

Taxonomy of Learning Methods

Learning: Estimating Parameters θ

- \mathcal{D} : data / evidence (e.g., (X, y))

Empirical Risk Minimization (ERM) (supervised) Minimize a predefined loss function (e.g., squared error loss):

$$\hat{\theta} = \arg \min_{\theta} \sum_{i=1}^N (y_i - f(x_i; \theta))^2$$

Maximum Likelihood Estimation (MLE). θ is deterministic

$$\hat{\theta} = \arg \max_{\theta} P(\mathcal{D}; \theta) = \arg \max_{\theta} \sum_{i=1}^N \log P(X_i = x_i, Y_i = y_i; \theta)$$

Maximum a Posteriori (MAP). Θ is random

$$\begin{aligned} \hat{\theta} &= \arg \max_{\theta} P(\Theta | \mathcal{D}) \\ &= \arg \max_{\theta} \left[\log P(\Theta = \theta) + \sum_{i=1}^N \log P(Y_i = y_i, X_i = x_i | \Theta = \theta) \right] \end{aligned}$$

The question is, which strategy do we use?

- For regression, we use ERM and MLE, since θ needs to be deterministic.

Learning via ERM

First, we define the objective: $g(\theta) = \sum_{i=1}^N (y_i - \theta_1 x_i - \theta_0)^2$. Then, we take the derivatives and set to 0.

$$\begin{aligned}\frac{\partial g(\theta)}{\partial \theta_0} &= \sum_{i=1}^N 2(y_i - \theta_1 x_i - \theta_0) = 0 \\ \frac{\partial g(\theta)}{\partial \theta_1} &= -\sum_{i=1}^N 2x_i(y_i - \theta_1 x_i - \theta_0) = 0\end{aligned}$$

Solving yields:

$$\hat{\theta}_1 = \frac{\frac{1}{N} \sum_i x_i y_i - \bar{x} \bar{y}}{\frac{1}{N} \sum_i x_i^2 - \bar{x}^2}, \quad \hat{\theta}_0 = \bar{y} - \hat{\theta}_1 \bar{x}$$

where $\bar{x} = \frac{1}{N} \sum_i x_i$ and $\bar{y} = \frac{1}{N} \sum_i y_i$

Learning via MLE

In univariate linear regression:

- θ, x_i are fixed (deterministic)
- $\epsilon_i \sim N(0, \sigma^2)$ is IID (independently and identically distributed) across samples / individuals.

Since $Y_i \sim N(\theta_1 x_i + \theta_0, \sigma^2)$, we get

$$f(y_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - \theta_1 x_i - \theta_0)^2}{2\sigma^2}\right)$$

We get this by plugging in the gaussian density function. From here, we can solve for $\hat{\theta}$.

$$\begin{aligned}\hat{\theta} &= \arg \max_{\theta} \sum_{i=1}^N \left(-\frac{1}{2} \log(2\pi\sigma^2) - \frac{(y_i - \theta_1 x_i - \theta_0)^2}{2\sigma^2} \right) \\ &= \arg \min_{\theta} \sum_{i=1}^N (y_i - \theta_1 x_i - \theta_0)^2\end{aligned}$$

Same optimization problem as the ERM before! Therefore, we also have the same solution for $\hat{\theta}_1$ and $\hat{\theta}_0$. See above.

Multivariate Linear Regression

The main difference between multivariate linear regression is that we have more than one feature.

Data

- N data points with $D > 1$ features: $x_i \in \mathbb{R}$ (or vector $x \in \mathbb{R}^N$)
- Labels: $y_i \in \mathbb{R}$ (or vector $y \in \mathbb{R}^N$)

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1D} \\ \vdots & \ddots & \vdots \\ x_{N1} & \cdots & x_{ND} \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$$

In our GWAS example, the data is in the same format, except we **analyze all SNPs at once**. We ask, is the SNP associated with the phenotype **given** other SNPs?

Model

Linear model:

$$Y_i = \sum_{j=1}^D \theta_j x_{ij} + \theta_0 + \epsilon_i, \quad i = 1, \dots, N$$

In this case, our equation is in matrix form: $Y_i = \theta^T \tilde{x}_i + \epsilon_i$, where $\tilde{x}_i = \begin{bmatrix} 1 \\ x_i \end{bmatrix} \in \mathbb{R}^{D+1}$

Assumptions:

- θ, x_i are fixed (deterministic)
- $\epsilon_i \sim N(0, \sigma^2)$, IID across samples / individuals

The rest is the same as the univariate model.

Learning via ERM

In multivariate linear regression:

- $f(x_i; \theta) = \theta^T \tilde{x}_i$ (recall that $\tilde{x}_i = \begin{bmatrix} 1 \\ x_i \end{bmatrix}$)
- No assumptions on ϵ_i

Define the objective: $g(\theta) = \sum_{i=1}^N (y_i - \theta^T \tilde{x}_i)^2 = (y - \tilde{X}\theta)^T (y - \tilde{X}\theta)$, where

$$\tilde{X} = \begin{bmatrix} \tilde{x}_1^T \\ \vdots \\ \tilde{x}_N^T \end{bmatrix} \in \mathbb{R}^{N \times (D+1)}$$

Take derivatives and set to 0:

$$\frac{\partial g(\theta)}{\partial \theta} = -2\tilde{X}^T (y - \tilde{X}\theta) = 0$$

Solving yields: $\hat{\theta}_1 = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T y$

Learning via MLE

In multivariate linear regression:

- θ, x_i are fixed (deterministic)
- $\epsilon_i \sim N(0, \sigma^2)$, IID across samples / individuals

Since $Y_i \sim N(\theta^T \tilde{x}_i, \sigma^2)$, $f(y_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - \theta^T \tilde{x}_i)^2}{2\sigma^2}\right)$. Again, we get this by plugging in the gaussian density formula. Solving,

$$\begin{aligned} \hat{\theta} &= \arg \max_{\theta} \sum_{i=1}^N \left(-\frac{1}{2} \log(2\pi\sigma^2) - \frac{(y_i - \theta^T \tilde{x}_i)^2}{2\sigma^2} \right) \\ &= \arg \min_{\theta} \sum_{i=1}^N (y_i - \theta^T \tilde{x}_i)^2 \end{aligned}$$

It also turns out that the ERM and MLE solutions are the same:

$$\hat{\theta}_1 = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T y$$

Sanity check: Is the Solution Consistent with the Univariate Case?

Solution for univariate regression:

$$\hat{\theta}_1 = \frac{\frac{1}{N} \sum_i x_i y_i - \bar{x}\bar{y}}{\frac{1}{N} \sum_i x_i^2 - \bar{x}^2}, \quad \hat{\theta}_0 = \bar{y} - \hat{\theta}_1 \bar{x}$$

Solution for multivariate regression:

$$\hat{\theta}_1 = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T y$$

- Approach 1: prove they are the same theoretically (not covered in this class)
- Approach 2: check via a small numerical example. (Write some code.)

Sample Size N vs. Feature Number D

Solution for multivariate regression:

$$\hat{\theta}_1 = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T y$$

Given D , what's the smallest N such that the solution above is possible?

To make $\tilde{X}^T \tilde{X}$ invertible, we need $N \geq D + 1$ and the columns of \tilde{X} be linearly independent.

Limitations of Multivariate Linear Regression

Multivariate linear regression (ERM / MLE)

$$\hat{\theta} = \arg \min_{\theta} \sum_{i=1}^N (y_i - \theta^T \tilde{x}_i)^2$$

Issue: when $N \leq D + 1$, the solution is not unique. See above section relating N and D .

Solution: add regularization to encode a preference (or a “prior”) over θ .

Regularized Regression

These methods add a regularization term to fix the problem described above.

- **Ridge regression:** $\hat{\theta} = \arg \min_{\theta} \sum_{i=1}^N (y_i - \theta^T \tilde{x}_i)^2 + \lambda \|\beta\|_2^2, \quad \lambda > 0$
- **LASSO regression:** $\hat{\theta} = \arg \min_{\theta} \sum_{i=1}^N (y_i - \theta^T \tilde{x}_i)^2 + \lambda \|\beta\|_1, \quad \lambda > 0$

where $\beta = (\theta_1, \dots, \theta_D)$ (and recall $\theta = (\theta_0, \theta_1, \dots, \theta_D)$)

Note: Both regularization terms favor smaller β . Additionally, there is no regularization on θ_0 , since we can eliminate that by centering the data.

Removing the Intercept via Data Centering

Center the data:

$$x_j \leftarrow x_i - \bar{x}_j, \quad \text{for } j = 1, \dots, D, \text{ and } y \leftarrow y - \bar{y}$$

After centering, the intercept is zero ($\theta_0 = 0$) and can be omitted.

Regularized regression after centering:

- Ridge regression: $\hat{\beta} = \arg \min_{\beta} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \|\beta\|_2^2$
- LASSO regression: $\hat{\beta} = \arg \min_{\beta} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \|\beta\|_1$

where $\beta = (\theta_1, \dots, \theta_D)$

Ridge Regression

Model

Frequentist Model: $(\theta_1, \dots, \theta_D)$ deterministic; $\beta = (\theta_1, \dots, \theta_D)$

$$\hat{\beta} = \arg \min_{\beta} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \|\beta\|_2^2$$

Bayesian model: $(\Theta_1, \dots, \Theta_D)$ random

$$Y_i = \Theta_{(1:)}^T x_i + \epsilon_i, \quad i = 1, \dots, N$$

- Assumptions:
 - x_i are fixed (deterministic)
 - $\Theta_{(1:)} = (\Theta_1, \dots, \Theta_D) \sim N(0, \sigma_0^2 I_D)$
 - $\epsilon_i \sim N(0, \sigma^2)$, IID
- Hyperparameters: σ^2, σ_0^2

Remark:

- Marginally, Y_i is random due to both $\Theta_{(1:)}$ and ϵ_i
- $Y_i | \Theta_{(1:)} = \theta_{(1:)} \sim N(\theta_{(1:)}^T x_i, \sigma^2)$, IID

Learning via ERM for the Frequentist Model

Empirical Risk Minimization (ERM)

$$\hat{\theta} = \arg \min_{\theta} \sum_{i=1}^N (y_i - \theta^T \tilde{x}_i)^2 + \lambda \|\theta\|_2^2, \quad \lambda > 0$$

Define the objective:

$$g(\beta) = (y - X\beta)^T (y - X\beta) + \lambda \beta^T \beta$$

Take derivatives and set to 0:

$$\frac{\partial g(\beta)}{\partial \beta} = -2X^T(y - X\beta) + 2\lambda\beta = 0$$

Solving yields: $\hat{\beta} = (X^T X + \lambda I_D)^{-1} X^T y$, where I_D is a D by D identity matrix.

Note: $X^T X + \lambda I_D$ is always invertible for $\lambda > 0$ (positive definite)

Learning via MAP for the Bayesian Model

Maximum a Posteriori (MAP)

$$\hat{\beta} = \arg \max_{\beta} \left[\log P(\Theta_{(1:)} = \beta) + \sum_{i=1}^N \log P(Y_i = y_i | \Theta_{(1:)} = \beta) \right]$$

- x_i are fixed (deterministic); $\beta = (\theta_1, \dots, \theta_D)$
- $\Theta_{(1:)} = (\Theta_1, \dots, \Theta_D) \sim N(0, \sigma_0^2 I_D)$; $\epsilon_i \sim N(0, \sigma^2)$, IID

Since

$$\Theta_{(1:)} \sim N(0, \sigma_0^2 I_D), f(\Theta_{(1:)} = \beta) = (2\pi\sigma_0^2)^{-\frac{D}{2}} \exp\left(-\frac{\beta^T \beta}{2\sigma_0^2}\right)$$

Since

$$[Y_i | \Theta_{(1:)} = \beta] \sim N(\beta^T x_i, \sigma^2), f(y_i | \Theta_{(1:)} = \beta) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_i - \beta^T x_i)^2}{2\sigma^2}\right)$$

We get these from plugging in the Gaussian density and Multivariate Gaussian density functions.

Then,

$$\begin{aligned} \hat{\beta} &= \arg \max_{\beta} -\frac{D}{2} \log(2\pi\sigma_0^2) - \frac{\beta^T \beta}{2\sigma_0^2} - \sum_{i=1}^N \left(\frac{1}{2} \log(2\pi\sigma^2) + \frac{(y_i - \beta^T x_i)^2}{2\sigma^2} \right) \\ &= \arg \min_{\beta} \frac{\sigma^2}{\sigma_0^2} \beta^T \beta + \sum_{i=1}^N (y_i - \beta^T x_i)^2 \quad (\text{Dropping terms independent of } \beta) \\ &= \arg \min_{\beta} \lambda \beta^T \beta + \sum_{i=1}^N (y_i - \beta_i^T)^2 \quad (\text{Letting } \lambda = \frac{\sigma^2}{\sigma_0^2}) \end{aligned}$$

Same optimization problem as the ERM before, giving $\hat{\beta} = (X^T X + \lambda I_D)^{-1} X^T y$

Learning Summary

1. Compute mean and center data:

- Means: $\hat{y}, \hat{x}_i, \dots, \hat{x}_D$
- Centering: $x_j \leftarrow x_i - \bar{x}_j$, for $j = 1, \dots, D$, and $y \leftarrow y - \bar{y}$

2. Learning (no intercept) ($\lambda = \frac{\sigma^2}{\sigma_0^2}$):

$$\hat{\beta} = (X^T X + \lambda I_D)^{-1} X^T y$$

3. Recover the original model with intercept

$$Y_i = \hat{\theta}_0 + \sum_{j=1}^D \hat{\theta}_j x_{ij}$$

where $\hat{\theta}_j = \hat{\beta}_j$ and $\hat{\theta}_0 = \bar{y} - \sum_{j=1}^D \bar{x}_j \hat{\theta}_j$

Note: Small σ_0^2 or large λ mean strong pull of the estimates towards zero.

LASSO Regression

Model

Similar to Ridge regression, except we are regularizing using the L1 norm instead of the L2 norm.

Frequentist model: $(\theta_1, \dots, \theta_D)$ deterministic; $\beta = (\theta_1, \dots, \theta_D)$

$$\hat{\beta} = \arg \min_{\beta} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \|\beta\|_1, \quad \lambda > 0$$

(Not covered in this class) Bayesian model $(\Theta_1, \dots, \Theta_D)$ random

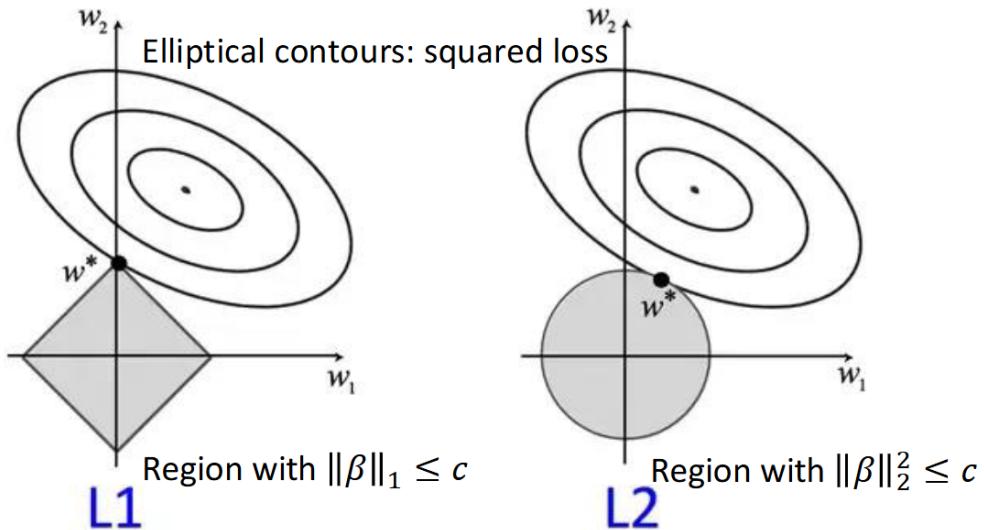
$$Y_i = \Theta_{(1:)}^T x_i + \epsilon_i, \quad i = 1, \dots, N$$

- Assumptions:
 - x_i are fixed (deterministic)
 - $\Theta_{(1:)} \sim \text{Laplace}(0, b)$ (double exponential prior)
 - $\epsilon_i \sim N(0, \sigma^2)$, IID
- Hyperparameters: σ^2, b

L1 vs. L2 Regularization

- L1 (LASSO) favors sparse solutions
- L2 (ridge) favors dense solutions with small overall magnitude

L1 Norm Regularization Introduces Sparsity



The loss contours hit the $L1$ constraint at corners, where one coefficient is exactly zero.

Sparsity

- One common assumption to make is **sparsity**
- **Makes statistical sense:** Learning a few non-zero regression weights is an easier problem than learning a large number of non-zero regression weights
- **Makes biological sense:** each phenotype is likely to be associated with a small number of SNPs, rather than all the SNPs

In a mathematical sense,

- Consider least squares linear regression problem
- Sparsity means most of the regression coefficients are zero.
- $\hat{\beta} = \arg \min_{\beta} \|Y - X\beta\|^2$ subject to $\sum_{j=1}^p \mathbb{I}[|\beta_j| > 0] \leq C$
 - $\mathbb{I}[A]$ is an indicator function, 1 if the condition A is satisfied, 0 otherwise.
- But solving this problem is computationally intractable.

L1 Regularization (LASSO)

An alternative problem,

$$\hat{\beta} = \arg \min_{\beta} \|Y - X\beta\|^2 + \lambda \|\beta\|_1$$

where $\|\beta_j\|_1 = \sum |\beta_j|$, is a form of the problem that is easier to solve!

- Unlike the constrained form above, this problem is not intractable.

LASSO Regression: Learning via ERM for the frequentist model

Empirical Risk Minimization (ERM)

$$\hat{\beta} = \arg \min_{\beta} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \|\beta\|_1, \quad \lambda > 0$$

No closed-form solution.

LASSO regularization is typically solved using numerical optimization methods, such as coordinate descent or proximal gradient methods.

- Pathwise coordinate descent [Friedman 2007 *Ann Appl Stat*]
 - In each iteration, update each β_j , and iterate until convergence

LASSO Regression: Learning Summary

1. Compute means and center data:

- Means: $\bar{y}, \bar{x}_1, \bar{x}_D$
- Centering: $y \leftarrow y - \bar{y}$, and $x_j \leftarrow x_j - \bar{x}_j$, for $j = 1, \dots, D$

2. Learning (no intercept):

- Solve the following problem using numerical methods:

$$\arg \min_{\beta} \sum_{i=1}^N (y_i - \beta^T x_i)^2 + \lambda \|\beta\|_1$$

3. Recover the original model with intercept

$$Y_i = \hat{\theta}_0 + \sum_{j=1}^D \hat{\theta}_j x_{ij}$$

where $\hat{\theta}_j = \hat{\beta}_j$ and $\hat{\theta}_0 = \bar{y} - \sum_{j=1}^D \bar{x}_j \hat{\theta}_j$

Larger λ enforces stronger sparsity.

Summary

- Univariate linear regression ($D = 1$)
 - ERM or MLE (Gaussian Noise)
- Multivariate linear regression ($D > 1$, X full column rank)
 - ERM or MLE
- Regularized regression (high-dimensional or ill-conditioned settings)
 - Ridge regression: L2 penalty encourages small overall magnitude; ERM or MAP (Gaussian prior)
 - LASSO regression: L1 penalty encourages sparse solutions; ERM or MAP (Laplace prior)