Machine learning - Master ICFP 2019-2020

Linear Least Squares Regression

Lénaïc Chizat

January 17, 2020

In this class, we introduce and analyze Linear Least Squares Regression, a tool that can be traced back to Legendre (1805) and Gauss (1809) and which remains widely used in machine learning.

1 Introduction

- We recall the goal of supervised machine learning: given some observations $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, $i = 1, \dots, n$, of inputs/outputs, features/variables (training data), given a new $x \in \mathcal{X}$, predict $y \in \mathcal{Y}$ (testing data) with a *regression* function f such that $y \approx f(x)$.
- In today's class, we consider empirical risk minimization with the square loss. This means that we choose a parameterized family of prediction functions $f_\theta : \mathcal{X} \to \mathcal{Y}$ for $\theta \in \Theta$ and minimize the empirical risk
$$\frac{1}{n}\sum_{i=1}^{n}(y_i - f_\theta(x_i))^2.$$
- When $\mathcal{X} \subset \mathbb{R}^d$, the set of affine functions is a natural default choice. To simplify notations, we assume that the first component of the covariates x_i is 1 so that it is sufficient to consider linear functions. We thus consider minimizing
$$\hat{\mathcal{R}}(w) := \frac{1}{n}\sum_{i=1}^{n}(y_i - x_i^\top w)^2.$$
- This expression can be rewritten in matrix notations. Let $y = (y_1, \dots, y_n)^\top \in \mathbb{R}^n$ be the vector of outputs and $X \in \mathbb{R}^{n\times d}$ the matrix of inputs, which rows are $x_i^\top$. It is called the *design* matrix. Its first column is a vector of 1 with our previous convention. In these notations, the empirical risk is
$$\hat{\mathcal{R}}(w) = \frac{1}{n}\|y - Xw\|_2^2. \tag{1}$$
- In the expressions above, it is possible to replace the input x_i by any (potentially non-linear) function $\Phi(x_i)$. For instance, regression with degree 2 polynomials with $d = 2$ is obtained with $\Phi(x_i) = (1, x_{i,1}, x_{i,2}, x_{i,1}^2, x_{i,2}^2, x_{i,1}x_{i,2})$.

2 Ordinary least squares estimator

We make the assumption that X is injective (i.e. the rank of X is d). In particular, the problem is over-determined and $d \leq n$.

Definition 1 *When X is injective, the minimizer of Eq.* (1) *is called the* Ordinary least Squares *(OLS) estimator.*

2.1 Closed form solution

Proposition 1 *If X is injective, then the OLS estimator exists and is unique. It is given by*

$$\hat{w} = (X^\top X)^{-1} X^\top y.$$

With the covariance *matrix $\Sigma := \frac{1}{n} X^\top X$, we have $\hat{w} = \frac{1}{n}\Sigma^{-1}X^\top y$.*

Proof Since $\hat{\mathfrak{R}}$ is coercive and continuous, it admits at least a minimizer. Moreover, it is differentiable, so a minimizer $\hat{w}$ must satisfy $\nabla\hat{\mathfrak{R}}(\hat{w}) = 0$. For all $w \in \mathbb{R}^d$, we have

$$\hat{\mathfrak{R}}(w) = \frac{1}{n}\left(\|y\|_2^2 - 2w^\top X^\top y + w^\top X^\top X w\right) \quad \text{and} \quad \nabla\hat{\mathfrak{R}}(w) = \frac{2}{n}\left(X^\top X w - X^\top y\right).$$

The condition $\nabla\hat{\mathfrak{R}}(\hat{w}) = 0$ gives the so-called *normal equations*

$$X^\top X\hat{w} = X^\top y.$$

Under the assumption that X is injective, the matrix $X^\top X$ is invertible (Exercise: show that $X^\top X$ is positive definite). So the normal equations have a unique solution $\hat{w} = (X^\top X)^{-1}X^\top y$. This shows the uniqueness of the minimizer of $\hat{\mathfrak{R}}$ as well as its closed form expression. ■

Another way to show uniqueness of the minimizer is by showing that $\hat{\mathfrak{R}}$ is strongly convex since $\nabla^2\hat{\mathfrak{R}}(w) = 2\Sigma$ for all $w \in \mathbb{R}^d$ (convexity will be studied in the following lectures).

2.2 Geometric interpretation

Proposition 2 *The vector of predictions $X\hat{w} = X(X^\top X)^{-1}X^\top y$ is the orthogonal projection of y on* $\mathrm{im}(X)$.

Proof Let us show that $P := X(X^\top X)^{-1}X^\top$ is the orthogonal projection on $\mathrm{im}(X)$. For any $a \in \mathbb{R}^d$, it holds $PXa = X(X^\top X)^{-1}X^\top Xa = Xa$, so $Pu = u$ for all $u \in \mathrm{im}(X)$. Also, since $\mathrm{im}(X)^\perp = \mathrm{null}(X^\top)$, it holds $Pu' = 0$ for all $u' \in \mathrm{im}(X)^\perp$. These properties characterize the orthogonal projection on $\mathrm{im}(X)$. ■

Thus we can interpret the OLS estimation as doing the following:

- compute $\bar{y}$ the projection of y on the image of X,
- solve the linear system $Xw = \bar{y}$ which has a unique solution.

2.3 Numerical resolution

While the closed form $\hat{w} = (X^\top X)^{-1} X^\top y$ is convenient for analysis, inverting $X^\top X$ is sometimes unstable and has a large computational cost when d is large. The following methods are usually preferred.

QR factorization. The QR decomposition factorizes the matrix X as $X = QR$ where $Q \in \mathbb{R}^{n \times d}$ has orthonormal columns and $R \in \mathbb{R}^{d \times d}$ is upper triangular. Computing a QR decomposition is faster and more stable than inverting a matrix. One has

$$\begin{aligned}(X^\top X)\hat{w} = X^\top y &\Leftrightarrow R^\top Q^\top Q R \hat{w} = R^\top Q^\top \\ &\Leftrightarrow R^\top R \hat{w} = R^\top Q^\top y \\ &\Leftrightarrow R\hat{w} = Q^\top y.\end{aligned}$$

It only remains to solve a triangular linear system which is easy.

Gradient descent. We can completely bypass the need of matrix inversion or factorization using gradient descent. It consists in minimizing $\hat{\mathcal{R}}$ by taking an initial point $w_0 \in \mathbb{R}^d$ and iteratively going towards the minimizer by following the opposite of the gradient

$$w_{k+1} = w_k - \eta \nabla \hat{\mathcal{R}}(w_k) \quad \text{for } k \geq 0,$$

where $\eta > 0$ is the step-size. When these iterates converge, it is towards the OLS estimator since a fixed-point w satisfies $\nabla \hat{\mathcal{R}}(w) = 0$. We will study such algorithms in Lecture 6.

3 Statistical analysis of OLS

We now prove guarantees on the performance of the OLS estimator. As before, we assume that X is injective.

3.1 Linear model and risk decomposition

Any kind of guarantee requires assumptions about how the data is generated. We assume that:

- there exists a vector $w^* \in \mathbb{R}^d$ such that the relationship between input and output is for $i \in [n]$

$$Y_i = x_i^\top w^* + Z_i.$$

- Z_i are independent and identically distributed (i.i.d.) of expectation $\mathbb{E} Z_i = 0$ and variance $\mathbb{E} Z_i^2 = \sigma^2$.

As before, we assume that $x_{i,1} = 1$ for all $i \in [n]$. We write Y and Z with capital letters to remind ourselves that (from now on) they are random variables. The vector Z accounts for variabilities in the output which are due to unobserved factors or to noise. From here, there are two settings of analysis for least squares:

- *Random design.* In this setting, both the input and the output are random. This is the classical setting of supervised machine learning, where the goal is *generalization.*
- *Fixed design.* In this setting, we assume that the input data $(x_1, \ldots, x_n)$ is *not* random and we are interested in obtaining a small prediction error on those input points. Our goal is to minimize

  $$\mathcal{R}_X(w) = \mathbb{E}_Y \left[\frac{1}{n} \sum_{i=1}^{n} \|Y_i - x_i^\top w\|_2^2 \right].$$

  This assumption allows a complete analysis with basic linear algebra. It is justified in some settings, e.g. when the input is a fixed grid, but is otherwise just a simplifying assumption. It can be understood as learning the vector $Xw^* \in \mathbb{R}^d$ of best predictions, instead of a function.

In today's class, we consider the fixed design setting.

Proposition 3 (Risk decomposition) *Under the linear model and fixed design assumptions, for any random variable $w \in \mathbb{R}^d$ (typically an estimator of w^*), the excess risk can be decomposed as*

$$\mathbb{E}[\mathcal{R}_X(w)] - \mathcal{R}^* = \underbrace{\|\mathbb{E}[w] - w^*\|_\Sigma^2}_{\textit{Bias}} + \underbrace{\mathbb{E}\Big[\|w - \mathbb{E}[w]\|_\Sigma^2\Big]}_{\textit{Variance}}$$

where $\Sigma := \frac{1}{n} X^\top X$ is the input covariance and $\|w\|_\Sigma^2 := w^\top \Sigma w$.

Proof Applying the result of Lecture 1 (Section 4), the Bayes predictor f^* for the square loss satisfies $f^*(x_i) = \mathbb{E}\left[Y_i\right] = x_i^\top w^*$ and the Bayes risk is $\mathcal{R}^* = \frac{1}{n}\sum_{i=1}^n \mathbb{E}\Big[(Y_i - f^*(x_i))^2\Big] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}[Z_i^2] = \sigma^2$. Now for any fixed $w \in \mathbb{R}^d$, we have

$$\begin{aligned}
\mathcal{R}_X(w) &= \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{Y_i}\Big[(Y_i - x_i^\top w)^2\Big] = \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{Y_i}\Big[(Y_i - x_i^\top w^* + x_i^\top w^* - x_i^\top w)^2\Big] \\
&= \frac{1}{n}\sum_{i=1}^n \Big(\mathbb{E}_{Y_i}\Big[(Y_i - x_i^\top w^*)^2\Big] + \mathbb{E}_{Y_i}\Big[(Y_i - x_i^\top w^*)(x_i^\top w^* - x_i^\top w)\Big] + \mathbb{E}_{Y_i}\Big[(x_i^\top w^* - x_i^\top w)^2\Big]\Big) \\
&= \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{Y_i}[Z_i^2] \qquad\qquad + \ 0 \qquad\qquad\qquad\qquad + \frac{1}{n}\sum_{i=1}^n \Big(x_i^\top (w - w^*)\Big)^2 \\
&= \sigma^2 + \frac{1}{n}(w - w^*)X^\top X(w - w^*) = \mathcal{R}^* + \|w - w^*\|_\Sigma^2.
\end{aligned}$$

Thus the excess risk of any linear predictor is $\mathcal{R}_X(w) - \mathcal{R}^* = \|w - w^*\|_\Sigma^2$. To reach our conclusion, we now consider a random w and perform the usual bias/variance decomposition:

$$\begin{aligned}
\mathbb{E}[\mathcal{R}_X(w)] - \mathcal{R}^* &= \mathbb{E}\Big[\|w - \mathbb{E}[w] + \mathbb{E}[w] - w^*\|_\Sigma^2\Big] \\
&= \mathbb{E}\Big[\|w - \mathbb{E}[w]\|_\Sigma^2\Big] + \mathbb{E}\Big[(w - \mathbb{E}[w])^\top \Sigma(\mathbb{E}[w] - w^*)\Big] + \mathbb{E}\Big[\|\mathbb{E}[w] - w^*\|_\Sigma^2\Big] \\
&= \mathbb{E}\Big[\|w - \mathbb{E}[w]\|_\Sigma^2\Big] + \|\mathbb{E}[w] - w^*\|_\Sigma^2.
\end{aligned}$$

■

- The quantity $\|w\|_\Sigma$ is called the Mahalanobis distance norm (it is a true norm whenever Σ is positive definite). It is the norm on the parameter space induced by the input data.
- It can be seen from the proof that the same risk decomposition holds in the random design setting, with $\Sigma = \mathbb{E}[xx^\top]$ the *population covariance.*

3.2 Statistical properties of the OLS estimator

We now analyze the properties of the OLS estimator. We divide the

Proposition 4 (Estimation properties of OLS) *Under the linear model and fixed design assumptions, the OLS estimator $\hat{w} = (X^\top X)^{-1}X^\top Y$ has the following properties:*

1. *it is unbiased $\mathbb{E}\hat{w} = w^*$,*
2. *its variance is $\mathrm{var}(\hat{w}) = \mathbb{E}\Big[(\hat{w} - w^*)(\hat{w} - w^*)^\top\Big] = \frac{\sigma^2}{n}\Sigma^{-1}$ (Σ^{-1} is often called the* precision *matrix).*

Proof

1. Since $\mathbb{E}Y = Xw^*$, we have directly $\mathbb{E}\hat{w} = (X^\top X)^{-1}X^\top Xw^* = w^*$.
2. It follows that $\hat{w} - w^* = (X^\top X)^{-1}X^\top Z$. Thus, using that $\mathbb{E}ZZ^\top = \sigma^2 I$, it holds

$$\mathrm{var}\,\hat{w} = \mathbb{E}\Big[(X^\top X)^{-1}X^\top ZZ^\top X(X^\top X)^{-1}\Big] = \sigma^2(X^\top X)^{-1}(X^\top X)(X^\top X)^{-1} = \sigma^2(X^\top X)^{-1} = \frac{\sigma^2}{n}\Sigma^{-1}.$$

■

Proposition 5 (Risk of OLS) *Under the linear model and fixed design assumptions, the excess risk of OLS is*

$$\mathbb{E}\left[\mathcal{R}_X(\hat{w})\right] - \mathcal{R}^* = \frac{\sigma^2 d}{n}.$$

Proof

Using the risk decomposition of Proposition 3 and the fact that $\mathbb{E}[\hat{w}] = w^*$, we have

$$\mathbb{E}\left[\mathcal{R}_X(\hat{w})\right] - \mathcal{R}^* = \mathbb{E}\|\hat{w} - w^*\|_\Sigma^2.$$

Using the identity $\hat{w} - w^* = (X^\top X)^{-1}X^\top Z$, we get

$$\begin{aligned}
\mathbb{E}[\mathcal{R}_X(\hat{w})] - \mathcal{R}^* &= \mathbb{E}\|(X^\top X)^{-1}X^\top Z\|_\Sigma^2 \\
&= \frac{1}{n}\mathbb{E}\Big[Z^\top X(X^\top X)^{-1}X^\top X(X^\top X)^{-1}X^\top Z\Big] \\
&= \frac{1}{n}\mathbb{E}\Big[Z^\top X(X^\top X)^{-1}X^\top Z\Big] \\
&= \frac{1}{n}\mathbb{E}\Big[Z^\top PZ\Big] = \frac{\sigma^2 d}{n}
\end{aligned}$$

where we used that $P = X(X^\top X)^{-1}X^\top$ is the orthogonal projection on $\mathrm{im}(X)$, which is d dimensional. ■

Exercise: what is the expected empirical risk $\mathbb{E}[\hat{\mathcal{R}}_X(\hat{w})]$? Solution: $\mathbb{E}[\hat{\mathcal{R}}_X(\hat{w})] = \frac{n-d}{n}\sigma^2$. In particular, when $n > d$, an unbiased estimator of the noise variance σ^2 is given by $\frac{\|Y-X\hat{w}\|_2^2}{n-d}$.

3.3 Gaussian noise model

If we make the stronger assumption that the noise is Gaussian, i.e. $Z_i \sim \mathcal{N}(0, \sigma^2)$, then the least mean square estimator of w^* coincides with the maximum likelihood estimator. The likelihood of Y is

$$P(Y|w,\sigma^2) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-(Y_i - x_i^\top w)^2/(2\sigma^2)\Big).$$

Taking the logarithm and removing constants, the maximum likelihood estimators $(\tilde{w}, \tilde{\sigma}^2)$ minimize

$$\frac{1}{2\sigma^2}\sum_{i=1}^n (Y_i - x_i^\top w)^2 + n\log(\sigma).$$

We see that $\tilde{w} = \hat{w}$.

Exercise: what is $\tilde{\sigma}^2$ the maximum likelihood of σ^2? Solution: $\tilde{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n (Y_i - x_i^\top \hat{w})^2$ (using the result of the previous exercise, observe that this estimator is biased towards 0).

4 Ridge least squares regression

- When d/n approaches 1, the prediction for x_i is barely better than Y_i by Proposition 5. Also when $d > n$, then $X^\top X$ is not invertible and the normal equations admit a linear subspace of solutions. These behaviors of OLS in high dimension (d large) are undesirable.
- Several solutions exist to fix these issues. The most common is to regularize the least squares objective, either by adding $\|w\|_1$ (*LASSO* regression, see Lecture 7) or $\|w\|_2^2$ (*ridge* regression, this lecture) to the empirical risk.

Definition 2 (Ridge least-squares regression) *For a regularization parameter $\lambda > 0$, we define the ridge least squares estimator $\hat{w}_\lambda$ as the minimizer of*

$$\min_{w\in\mathbb{R}^d} \frac{1}{n}\|Y - Xw\|_2^2 + \lambda\|w\|_2^2.$$

Proposition 6 *Let us define $\Sigma_\lambda = \frac{1}{n}X^\top X + \lambda I$, and we recall that $\Sigma = \frac{1}{n}X^\top X$. It holds*

$$\hat{w}_\lambda = \frac{1}{n}\Sigma_\lambda^{-1}X^\top Y.$$

Proof Left as an exercise (similar to the proof of Proposition 1). ■

As for the OLS, we can analyze the statistical properties of this estimator under the linear model and fixed design assumptions. To simplify the derivations and without loss of generality, we assume that the axes are aligned with the eigenvectors of Σ and we have $\Sigma = \mathrm{diag}(\lambda_1, \dots, \lambda_d)$. In this section, we do not assume that X is injective, so $\lambda_j \geq 0$ is potentially 0 and d may be larger than n. Note that we have $\Sigma_\lambda = \mathrm{diag}(\lambda_1 + \lambda, \dots, \lambda_d + \lambda)$

Proposition 7 *Under the linear model assumption, the ridge least squares estimator* $\hat{w}_\lambda = \frac{1}{n}\Sigma_\lambda^{-1}X^\top Y$ *has the following excess risk*

$$\mathbb{E}\left[\mathcal{R}_X(\hat{w}_\lambda)\right] - \mathcal{R}^* = \sum_{j\geq 1}(w_j^*)^2 \frac{\lambda_j}{1+\lambda_j/\lambda} + \frac{\sigma^2}{n}\sum_{j\geq 1}\frac{\lambda_j^2}{(\lambda_j+\lambda)^2}.$$

Observe how this converges to the OLS estimator (when it is defined) as $\lambda \to 0$.

Proof We use the risk decomposition of Proposition 3 into a bias B and variance V terms. Since it holds $\mathbb{E}[\hat{w}_\lambda] = \frac{1}{n}\Sigma_\lambda^{-1}X^\top X w^* = \Sigma_\lambda^{-1}\Sigma w^*$, it follows

$$\begin{aligned}
B &= \|\mathbb{E}[\hat{w}_\lambda] - w^*\|_\Sigma^2 \\
&= (w^*)^\top(\Sigma_\lambda^{-1}\Sigma - I)\Sigma(\Sigma_\lambda^{-1}\Sigma - I)w^* \\
&= \sum_{j\geq 1}(w_j^*)^2\left(\frac{\lambda_j}{\lambda_j+\lambda} - 1\right)^2 \lambda_j = \sum_{j\geq 1}(w_j^*)^2\frac{\lambda_j}{1+\lambda_j/\lambda}.
\end{aligned}$$

For the variance term, using the fact that $ZZ^\top = \sigma^2 I$ we have

$$\begin{aligned}
V &= \mathbb{E}\Big[\|\hat{w}_\lambda - \mathbb{E}[\hat{w}_\lambda]\|_\Sigma^2\Big] \\
&= \mathbb{E}\Big[\Big\|\frac{1}{n}\Sigma_\lambda^{-1}X^\top Z\Big\|_\Sigma^2\Big] \\
&= \mathbb{E}\Big[\frac{1}{n^2}\,\mathrm{tr}\left(Z^\top X\Sigma_\lambda^{-1}\Sigma\Sigma_\lambda^{-1}X^\top Z\right)\Big] \\
&= \mathbb{E}\Big[\frac{1}{n^2}\,\mathrm{tr}\left(X^\top ZZ^\top X\Sigma_\lambda^{-1}\Sigma\Sigma_\lambda^{-1}\right)\Big] \\
&= \frac{\sigma^2}{n}\,\mathrm{tr}\left(\Sigma\Sigma_\lambda^{-1}\Sigma\Sigma_\lambda^{-1}\right) \\
&= \frac{\sigma^2}{n}\sum_{j\geq 1}\frac{\lambda_j^2}{(\lambda_j+\lambda)^2}.
\end{aligned}$$

The proposition follows by summing the bias and variance terms. ■

Choosing λ in theory. Based on this expression for the risk, we can tune the parameter λ to obtain a potentially better bound than with the OLS (which corresponds to $\lambda = 0$).

Proposition 8 *With the choice* $\lambda^* = \frac{\sigma\sqrt{\mathrm{tr}(\Sigma)}}{\|w^*\|_2\sqrt{n}}$, *we have*

$$\mathbb{E}\left[\mathcal{R}_X(\hat{w}_{\lambda^*})\right] - \mathcal{R}^* \leq \frac{\sigma\sqrt{\mathrm{tr}(\Sigma)}\|w^*\|_2}{\sqrt{n}}.$$

Proof Let us upper bound the bias and variance terms using the inequality $(a+b)^2 \geq 2ab$ for $a, b \geq 0$. We have

$$B \leq \sum_{j\geq 1} (w_j^*)^2 \frac{\lambda_j}{2\lambda_j/\lambda} = \frac{\lambda\|w^*\|_2^2}{2}$$

$$V \leq \frac{\sigma^2}{n}\sum_{j\geq 1}\frac{\lambda_j^2}{2\lambda_j\lambda} = \frac{\sigma^2\,\mathrm{tr}\,\Sigma}{\lambda n}.$$

Plugging in λ^* (which was chosen to minimize the upper bound on $B+V$) gives the result. ■

- Observe that if we write $R = \max_i \|x_i\|_2^2$, then we have

  $$\mathrm{tr}(\Sigma) = \sum_{j\geq 1}\Sigma_{jj} = \frac{1}{n}\sum_{i=1}^n\sum_{j\geq 1} x_{i,j}^2 = \frac{1}{n}\sum_{i=1}^n \|x_i\|_2^2 \leq R^2.$$

  Thus in the excess risk bound, the dimension d plays no role and it could even be infinite (given R and $\|w^*\|_2$ remain finite). This type of bounds are called *dimension free* bounds.

- Comparing this bound with that of the OLS estimator, we see that it converges slower to 0 as a function of n (from n^{-1} to $n^{-1/2}$) but it has a milder dependence on the noise (from σ^2 to σ).

- The value of λ^* involves quantities which we typically do not know in practice (σ and $\|w^*\|_2$).

Choosing λ in practice. The regularization λ is an example of a *hyper-parameter*. This term refers broadly to any quantity that influences the behavior of a machine learning algorithm and that is left to choose by the practitioner. While theory often offers guidelines and qualitative understanding on how to best chose the hyper-parameters, their precise numerical value depends on quantities which are often difficult to know or even guess. In practice, we typically resort to the two following approaches, which are attempts to estimate the test performance without using the test set:

- *Validation.* We divide the training data into two categories: a training set and a validation set. Given a choice of hyper-parameters, we run the algorithm on the training set and then measure the performance on the validation set. We repeat this process with different hyper-parameters and we select the hyper-parameters with the best validation performance.

- *Cross-validation.* A similar, but slightly more elaborate method is to partition the training data into k smaller data sets (typically $k = 5$). Each of these subsets plays successively the role of the validation set while the other form the training set. This gives, for each choice of hyper-parameters, k validation performances that can be averaged to choose the best hyper-parameter.