MATH 412: STATISTICAL MACHINE LEARNING

Rafiki's Notes

Rafael Barroso
Ingenierie Mathematique
École Polytechnique Fédérale de Lausanne
September 23, 2025

Contents

1 Supervised Learning Basics

In this area of machine learning, we try to understand certain relations beween input-output data. If such relations are established, we then wish to generalize for new *unseen* data. Things start getting even jucier whenever we wish to take decisions based on new data, resulting in a more generalized task. In this section we explore the formalization provided by Prof. Zemel.

1. **We have:**
   - Data: $\mathcal{D}_n := \{(x_0, y_0), \ldots, (x_n, y_n)\}$
   - i.e. tuples of the form (x_i, y_i)
   - $x_i :=$ input; $y_i :=$ output
2. **We want:**
   - Given $\mathcal{D}_n$, learn relations of the x_i's with the corresponding y_i's such that we may infer something about a new unseen y' given x'.

We now define the two types of tasks cosidered inside supervised learning (amongst others).

Definition 1.1. A *prediction* task is established to be the discovery of y‘ (unseen) given x‘. A *decision* task on the other hand, focuses on producing a decision based on $(x$‘$, y$‘$)$ only with the data of x‘

For example, take into consideration a medical diagnosis. We have $x_i :=$ patient data i.g. $\{\text{weight}_i, \text{height}_i, \ldots\}$; $y_i := \{\text{positive}, \text{negative}\}$. Then, a **prediction task** would consist in predicting y‘ given x‘. A **decision task** on the other hand, would then consist on choosing how to treat patient x‘ i.g. choosing medicine $m \in \{A, B, C\}$ (we have to decide on y‘ by only seeing x‘).

We now consider the space of all possible decisions; a *learning algorithm* (sometimes called *learning scheme*) $\mathscr{A}$.

Definition 1.2. We define a learning algorithm as

$$\mathscr{A} : \mathcal{D}_n \to \hat{f}$$

where $\hat{f}$ is our decision function.

Obviously we want $\hat{f}$ to be “good” (otherwise, *nos estamos haciendo pendejos*). Hence, we must define what it means for $\hat{f}$ to be “good” i.e. what we want from $\hat{f}$.

Definition 1.3. Let $\mathcal{X}$ be the input space, then, a decision function is defined as

$$f : \mathcal{X} \to \mathcal{A}^{\mathcal{X}}$$

Note that the input space $\mathcal{X}$ is the space of all x_i's.

Ideally, as stated before, we want a "good" function (i.e. decision function) f such that $f(x) \in \mathcal{A}^{\mathcal{X}}$ is "good" when compared to an unseen y. This means that $f(x)$ must be an accurate prediction of y and it has the **smallest possible cost** whenever y occurrs. So, we compute the *loss function* l.

Definition 1.4. Let $\mathcal{Y}$ be the space of all possible outcomes, then

$$l : \mathcal{A}^{\mathcal{X}} \times \mathcal{Y} \to \mathbb{R}$$

defined by $(f(x) = a, y) \mapsto l(a, y)$. Note that this function meassures the cost of taking decision f whenever y occurrs i.e. the *risk*.

Remark 1.5. Note that all the above definitions boil down to the fact that we're trying to design a 'good' learning algorithm $\mathscr{A}$ that produces $\hat{f}$ in such a way that the risk is minimized. We formalize the definition of a learning algorithm as follows

$$\mathscr{A} : (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{A}^{\mathcal{X}} \text{ given by } \mathcal{D}_n \mapsto \hat{f}$$

Throughout the lecture, unless stated otherwise, we assume that the data is generated by a stochastic process and done so i.i.d. as random variables i.e. (X_i, Y_i).

Because of the fact that this is a statistics class, we'll start getting into the *deets* using much more of their language (statisticians have a fetish for fancy language and syntax). Hence we would like to define what the *expected cost* of taking decision f as the risk $\mathcal{R}$.

Definition 1.6. We define the risk as follows

$$\mathcal{R}(f) := \mathbb{E}\big[l(a, Y)\big]. \text{ If } \exists f^* \in \mathcal{A}^{\mathcal{X}} : \mathcal{R}(f^*) = \inf_{f \in \mathcal{A}^{\mathcal{X}}} \mathcal{R}(f)$$

then, that f^* is our juicy function we're looking for! statisticians call it the *target* function. Now, the *conditional risk* of taking f as an action given x has happened is defined as

$$\mathcal{R}(f(x) = a|x) = \mathbb{E}\big[l(a, Y)|X = x\big] = \int l(a, y) dP_{Y|X}(y|x)$$

Note that $dP_{Y|X}(y|x)$ just means we're integrating over the conditional distribution of Y given $X = x$. To simplify further (and remark the fetish statisticians posse), this just means that we're taking average the loss over all possible outcomes of Y, weighted by how likely they are given $X = x$.

Remark 1.7. Note that

$$\mathbb{E}\big[\mathcal{R}(f(X)|X)\big] = \mathbb{E}\big[\mathbb{E}\big[l(f(X),Y)|X=x\big]\big] = \mathbb{E}\big[l(f(X),Y)\big]$$

i.e. the expected value of $\mathcal{R}(f)$.

We finally make the last definition of the section, since we're interested in meassuring risks we shall compute the *excess risk* $\varepsilon(f)$ while we're at it. This number tells us how much of our risk is over the optimal ammount.

Definition 1.8. The excess risk is given by

$$\varepsilon(f) := \mathcal{R}(f) - \mathcal{R}(f^*) = \mathbb{E}\big[l(f(X),Y)\big] - \mathbb{E}\big[l(f^*(X),Y)\big]$$

$$\Rightarrow \varepsilon(f) = \mathbb{E}\big[l(f(X),Y) - l(f^*(X),Y)\big]$$

1.1 Examples

As for every new concept, one must get their hands dirty in order to fully grasp the idea that is being transmitted. We explore some basic examples as to how these things might be used in practice by taking a look at two main cases: *Ordinary Least Squares* and *Classification*. Our main goal is to derived the target function f^* given a loss function l.

1.1.1 Ordinary Least Squirts Regression (OLS)

Consider the case where both our action space $\mathcal{A}^{\mathcal{X}}$ and our output space $\mathcal{Y}$ are both the real numbers $\mathbb{R}$. The loss function in this case is then defined as $l : \mathbb{R}^2 \to \mathbb{R}$ and it is given by $l(f(x) = a, y) = (a-y)^2$. Then, the expected cost of taking decision f is the following

$$\mathcal{R}(f) = \mathbb{E}\big[l(a,y)\big] = \mathbb{E}\big[(a-y)^2\big] = \mathbb{E}\big[(y-a)^2\big]$$

We now make an educated guess (cheeky little trick) and assume that $\hat{f} = \mathbb{E}\big[Y|X\big]$ and so, we consider the following expected conditional risk:

$$\mathcal{R}(f(X)|X) = \mathbb{E}\big[l(a,y)|X\big] = \mathbb{E}\big[(Y-f(X))^2|X\big] = \mathbb{E}\big[(Y-\hat{f}(X)+\hat{f}(X)-f(X))^2|X\big]$$

Note that we also implemented a cheeky trick, it's just adding a glorified zero into the mix i.e. $-\hat{f}(X) + f^*(X) = 0$. Expanding the quadratic term inside the expectation, we obtain the following equation:

$$\mathbb{E}\big[(Y-\hat{f}(X))^2 + 2\big(Y-\hat{f}(X)\big)\big(\hat{f}(X)-f(X)\big) + (\hat{f}(X)-f(X))^2|X\big]$$

$$= \mathbb{E}\big[(Y-\hat{f}(X))^2|X\big] + \mathbb{E}\big[2\big(Y-\hat{f}(X)\big)\big(\hat{f}(X)-f(X)\big)|X\big] + \mathbb{E}\big[(F^*(X)-F(X))^2|X\big]$$

$$= \mathbb{E}\big[(Y-\hat{f}(X))^2|X\big] + 2\big(\hat{f}(X)-f(X)\big)\mathbb{E}\big[Y-\hat{f}(X)|X\big] + \mathbb{E}\big[(\hat{f}(X)-f(X))^2|X\big]$$

Remark 1.9. Note that $2\big(\hat{f}(X) - f(X)\big)\mathbb{E}\big[Y - \hat{f}(X)|X\big] = 0$ since

$$\mathbb{E}\big[Y - \hat{f}(X)|X\big] = \mathbb{E}\big[Y - \mathbb{E}\big[Y|X\big]|X\big] = \mathbb{E}\big[Y|X\big] - \mathbb{E}\big[Y|X\big] = 0$$

So, we get that the conditional risk:

$$\mathcal{R}(f(X)|X) = \mathbb{E}\big[(Y - \hat{f}(X))^2|X\big] + \mathbb{E}\big[(\hat{f}(X) - f(X))^2|X\big]$$

$$\Rightarrow \mathcal{R}(f(X)|X) = \mathcal{R}(\hat{f}(X)|X) + \mathbb{E}\big[(\hat{f}(X) - f(X))^2|X\big]$$

Now, let's think for a moment. We essentially want to find f^* (target function) such that our risk $\mathcal{R}(f(X)|X)$ is minimized whenever we take that action.

Since $\mathcal{R}(\hat{f}(X)|X)$ can't help us, we turn our but cheeks to the other term, i.e. $\mathbb{E}\big[(\hat{f}(X) - f(X))^2|X\big]$. Note that if $f = \hat{f}$, this term goes to zero, and hence our conditional risk is minimized. Voila! since there actually exists such function that minimizes our risk, we've arrived at our sensual (target) function babyyy.

$$\therefore\ f^* = \hat{f}$$

1.1.2 Classification

We now consider a classification problem where $\mathcal{A}^{\mathcal{X}} = \mathcal{Y} := \{0, \dots, K-1\}$ and the loss function is given by the 0-1 loss function i.e. $l(a, y) = \mathbb{1}_{\{a \neq y\}}$. Then, the risk of f is given by the equation

$$\mathcal{R}(f) = \mathbb{E}\big[l(f(x) = a, y)\big] = \mathbb{P}\big(f(X) \neq Y\big) = 1 - \mathbb{P}\big(f(X) = Y\big)$$

Now, if we consider the conditional risk, we obtain the following derivation

$$\mathcal{R}(f|X = x) = \mathbb{E}\big[l(f(x) = a, y)|X = x\big] = \mathbb{P}\big(f(x) \neq Y|X = x\big) = 1 - \mathbb{P}\big(f(x) = Y|X = x\big)$$

Think for a moment. This states that in order to minimize $\mathcal{R}(f|X = x)$ we must then find the highest possible value for $\mathbb{P}\big(f(x) = Y|X = x\big) = \mathbb{P}\big(Y = f(x)|X = x\big)$. So, the following equivalence holds

$$\min\mathcal{R}(f|X = x) = \max_{f \in \mathcal{A}^{\mathcal{X}}} \mathbb{P}\big(Y = f(x)|X = x\big)$$

$$\Rightarrow \arg\max_{k \in K} \mathbb{P}\big(Y = k|X = x\big) = f^*$$

And so, we've found our hard-to-get (like women playing their games) target function. That last implication basically states that we must do the following: *for each value of x, pick y that makes it the largest possible value.*

This is a great book![1]

Exercise Sheet 1

The following exercises are designed to reinforce the concepts introduced in this section. They provide both practice with fundamental techniques and opportunities to explore some extensions of the main results. Complete solutions are included to aid understanding and self-study.

Exercise 1.1: Classification from a discrete input space

We consider a multiclass classification problem with 3 classes $\mathcal{Y} = \{1, 2, 3\}$ for data with only a single discrete descriptor in $\mathcal{X} = \{1, 2, 3, 4\}$.

We assume that the joint probability distribution $\mathbb{P}(Y = y, X = x)$, with $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$, is specified by the following table:

	$Y = 1$	$Y = 2$	$Y = 3$
$X = 1$	0.02	0.08	0.10
$X = 2$	0.05	0.40	0.15
$X = 3$	0.02	0.02	0.12
$X = 4$	0.02	0.01	0.01

1. What is the target function f^* for the 0–1 loss?
2. What are the values of $f^*(x)$ for $x = 1, 2, 3, 4$?
3. What is the value of the risk for the target function?

Answers:

As shown above, $f^*(x) = \arg\max_{y \in \mathcal{Y}} \mathbb{P}(Y = y | X = x)$.

Now, if $\mathcal{X} = \{1, 2, 3, 4\}$ and $\mathcal{Y} = \{1, 2, 3\}$, where $\mathbb{P}(Y = y | X = x)$ is given by a table, we find $f^* * (x)$ for $x \in \mathcal{X}$. We want to find the y that maximizes $\mathbb{P}(Y = y | X = x)$. For example, if $\max_y \mathbb{P}(Y = y | X = 1) = 0.1$, which occurs when $y = 3$, then $f^*(1) = 3$. So, for $x = 1, f^*(1) = 3$. If $x = 2, f^*(2) = 2$. For $x = 3, f^*(3) = 3$. For $x = 4, f^*(4) = 1$.

Finally, the risk is the probability of misclassification:

$$\mathbb{P}(f^*(X) \neq Y) = \mathbb{E}[\mathbb{1}_{f^*(X) \neq Y}] = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} \mathbb{1}_{f^*(x) \neq y} \mathbb{P}(X = x, Y = y)$$

Exercise 1.2: Recap of linear models

Let $y = X\beta + \varepsilon$, where $\mathbb{E}[\varepsilon] = 0$, $\mathrm{Var}(\varepsilon) = \sigma^2 I$, and X is a non-random full rank matrix of size $n \times p$. This setup contains the Gauss–Markov assumptions of a linear model.

1. Derive the least squares estimator $\hat{\beta} = (X^\top X)^{-1} X^\top y$.
2. Show that $\hat{\beta}$ is unbiased and that the variance of $\hat{\beta}$ is given by $\sigma^2 (X^\top X)^{-1}$.

Answers:

The model is $y = X\beta + \epsilon$, with assumptions $\mathbb{E}[\epsilon] = 0$ and $\text{Var}(\epsilon) = \sigma^2 I$. We assume X is full rank. Note that $X^T X$ is therefore invertible (positive definite). Since $\mathbb{E}[\epsilon] = 0$, we have $\mathbb{E}[Y|X] = X\beta$.

Now, the loss function is $L(\beta) = \sum_i (y_i - x_i^T \beta)^2 = (y - X\beta)^T (y - X\beta)$.

$$\begin{aligned} L(\beta) &= (y^T - (X\beta)^T)(y - X\beta) \\ &= y^T y - y^T X\beta - (X\beta)^T y + (X\beta)^T (X\beta) \\ &= y^T y - y^T X\beta - \beta^T X^T y + \beta^T X^T X\beta \\ &= y^T y - 2\beta^T X^T y + \beta^T X^T X\beta \end{aligned}$$

To find the minimum, we take the derivative with respect to β and set it to 0.

$$\frac{\partial L}{\partial \beta} = -2X^T y + 2X^T X\beta$$

Setting to 0:

$$-2X^T y + 2X^T X\beta = 0 \implies X^T X\beta = X^T y \implies \hat{\beta} = (X^T X)^{-1} X^T y$$

Now, let's find the expectation of the estimator $\hat{\beta}$ to check for bias.

$$\begin{aligned} \mathbb{E}[\hat{\beta}] &= \mathbb{E}[(X^T X)^{-1} X^T y] \\ &= (X^T X)^{-1} X^T \mathbb{E}[y] \\ &= (X^T X)^{-1} X^T \mathbb{E}[X\beta + \epsilon] \\ &= (X^T X)^{-1} X^T (X\beta + \mathbb{E}[\epsilon]) \\ &= (X^T X)^{-1} (X^T X)\beta + 0 \\ &= \beta \end{aligned}$$

Therefore, $\mathbb{E}[\hat{\beta}] = \beta$, so the estimator is unbiased. Continuing from Ex 1.2, we find the variance of $\hat{\beta}$.

$$\begin{aligned} \text{Var}(\hat{\beta}) &= \text{Var}((X^T X)^{-1} X^T y) \\ &= \text{Var}((X^T X)^{-1} X^T (X\beta + \epsilon)) \\ &= \text{Var}((X^T X)^{-1} X^T X\beta + (X^T X)^{-1} X^T \epsilon) \\ &= \text{Var}(\beta + (X^T X)^{-1} X^T \epsilon) \\ &= 0 + \text{Var}((X^T X)^{-1} X^T \epsilon) \quad \text{(since } \beta \text{ is a constant)} \end{aligned}$$

Using the rule $\text{Var}(AZ) = A\text{Var}(Z)A^T$:

$$\begin{aligned}\text{Var}(\hat{\beta}) &= ((X^TX)^{-1}X^T)\text{Var}(\epsilon)((X^TX)^{-1}X^T)^T \\ &= ((X^TX)^{-1}X^T)(\sigma^2 I)(X(X^TX)^{-1}) \\ &= \sigma^2(X^TX)^{-1}X^TX(X^TX)^{-1} \\ &= \sigma^2(X^TX)^{-1}I \\ &= \sigma^2(X^TX)^{-1}\end{aligned}$$

Exercise 1.3: Linear regression for binary classification

Consider a binary classification problem with $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y} = \mathcal{A} = \{-1, 1\}$. We model the conditional expectation of Y given X by the linear model

$$\mathbb{E}[Y \mid X] = X^\top \beta.$$

Let $x \in \mathbb{R}^n$ be a new input. So, we estimate

$$\hat{\mathbb{E}}[Y \mid X = x] = x^\top \hat{\beta},$$

where $\hat{\beta}$ is the least-squares estimate of β. We wish to estimate its class $y = f^*{*}(x)$, where $f^*{*}$ is the target function corresponding to 0–1 loss.

1. Derive the linear model estimate of $\hat{P}(Y = 1 \mid X = x)$.

2. Show that
$$\hat{y} = \hat{f}^*(x) = 2 \cdot \mathbf{1}\{x^\top \hat{\beta} \geq 0\} - 1,$$
where $\hat{f}^*$ is the estimate of $f^*{*}$ given by plugging in the estimated conditional probabilities $\hat{P}(Y = y \mid X = x)$.

Answers:

Setup: $\mathcal{X} = \mathbb{R}^n$, $\mathcal{Y} = \{-1, 1\}$. We model the conditional expectation as $E[Y|\mathbf{x}] = \mathbf{x}^T\beta$.
(a) Derive the linear model estimate of $\hat{P}(Y = 1|\mathbf{x})$.

$$\begin{aligned}\mathbb{E}[Y|\mathbf{x}] &= \sum_{y \in \{-1,1\}} y \cdot P(Y = y|\mathbf{x}) \\ &= (1)P(Y = 1|\mathbf{x}) + (-1)P(Y = -1|\mathbf{x}) \\ &= P(Y = 1|\mathbf{x}) - P(Y = -1|\mathbf{x})\end{aligned}$$

Since $P(Y = -1|\mathbf{x}) = 1 - P(Y = 1|\mathbf{x})$,

$$\begin{aligned}\mathbb{E}[Y|\mathbf{x}] &= P(Y = 1|\mathbf{x}) - (1 - P(Y = 1|\mathbf{x})) \\ &= 2P(Y = 1|\mathbf{x}) - 1\end{aligned}$$

Hence, solving for the probability:

$$P(Y = 1|\mathbf{x}) = \frac{\mathbb{E}[Y|\mathbf{x}] + 1}{2}$$

Plugging in our linear model estimate $\hat{\mathbb{E}}[Y|\mathbf{x}] = \mathbf{x}^T\hat{\beta}$:

$$\hat{P}(Y = 1|\mathbf{x}) = \frac{\mathbf{x}^T\hat{\beta} + 1}{2}$$

(b) Show that $\hat{y} = \hat{f}^*(\mathbf{x})$ is given by $2 \cdot \mathbb{1}_{\{\mathbf{x}^T\hat{\beta} \geq 0\}} - 1$. We know that the Bayes classifier $\hat{f}^*(\mathbf{x})$ picks the class with the highest estimated probability. Thus, we predict $\hat{y} = 1$ if $\hat{P}(Y = 1|\mathbf{x}) \geq \hat{P}(Y = -1|\mathbf{x})$.

$$\frac{\mathbf{x}^T\hat{\beta} + 1}{2} \geq 1 - \frac{\mathbf{x}^T\hat{\beta} + 1}{2}$$

$$\frac{\mathbf{x}^T\hat{\beta} + 1}{2} \geq \frac{1 - \mathbf{x}^T\hat{\beta}}{2}$$

$$\mathbf{x}^T\hat{\beta} + 1 \geq 1 - \mathbf{x}^T\hat{\beta}$$

$$2\mathbf{x}^T\hat{\beta} \geq 0 \implies \mathbf{x}^T\hat{\beta} \geq 0$$

This tells us that we predict class 1 for positive values of $\mathbf{x}^T\hat{\beta}$ and class -1 for negative values. This is indeed given by the function $2 \cdot \mathbb{1}_{\{\mathbf{x}^T\hat{\beta} \geq 0\}} - 1$:

- If $\mathbf{x}^T\hat{\beta} \geq 0$, then $2(1) - 1 = 1$.
- If $\mathbf{x}^T\hat{\beta} < 0$, then $2(0) - 1 = -1$.

2 Linear Regression

In this section we discuss the method of linear regression. Here, we consider the following setting.

- We begin with a collection of data $D_n := \{(a_1, y_1), \ldots, (x_n, y_n)\}$
- We assume the data has the form (assume a model i.e. linear regression) $y = X\beta + \varepsilon$
- Input space $\mathcal{X} = \mathbb{R}^p$; Outcome space $\mathcal{Y} = \mathbb{R}$.
- Action spcae (sometimes called *hypothesis space*) $\mathcal{A}^{\mathcal{X}} := \{f_w | w \in \mathbb{R}^p\}$ where

$$f_w : x \mapsto w^\top x.$$

In this case,

$$X = \begin{bmatrix} - - - & x_1^\top & - - - \\ - - - & x_2^\top & - - - \\ & \vdots & \\ - - - & x_n^\top & - - - \end{bmatrix}, \; y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Remark 2.1. Some important facts to consider:

1. Sometimes, we want to do some data pre-processing. This means we might want to center our data i.e. $x_i^c = x_i - \bar{x}$ and we might also want to normalize.
2. Note that $X \in \mathbb{R}^{n \times p}$; the *design matrix* whose i^{th} row is $x_i^\top$.

It is important to note that linear regression is basically us assuming that our data takes the shape of our model (i.e. $y = X\beta + \varepsilon$) where we also need an *estimator* $\hat{\beta}$ i.e. a 'way' to aproximate β. In the previous section, we explored such a method to do this; recall Ordinary Least Squares (OLS).

Remark 2.2. The *risk* of a predictor $f_w(x) = w^\top x$ under squared loss is

$$R(w) = \mathbb{E}\big[(Y - w^\top X)^2\big],$$

where the expectation is taken with respect to the (unknown) distribution of (X, Y). Since this expectation cannot be computed directly, we approximate it by the *empirical risk*:

$$\hat{R}_n(w) = \frac{1}{n} \sum_{i=1}^{n} (y_i - w^\top x_i)^2.$$

This corresponds to replacing the population expectation with the sample average. By the law of large numbers, $\hat{R}_n(w) \to R(w)$ as $n \to \infty$. Thus, empirical risk minimization provides a practical way to approximate population risk.

When we consider this in a vectorized form, we obtain that

$$\hat{R}_n(w) = \frac{1}{n}\sum_{i=1}^{n}(y_i - w^\top x_i)^2 = \frac{1}{n}||y - Xw||_2^2$$

In order to minimize this risk, i.e. solve $\min_{w\in\mathbb{R}^d}\hat{\mathcal{R}}(\hat{f_w})$ we consider that

$$\hat{R}_n(w) = \frac{1}{n}\left(w^\top X^\top X w - 2w^\top X^\top y + ||y||_2^2\right)$$

is a *differentiable convex* function, whose minima are then characterized by the equation

$$X^\top X w - X^\top y = 0.$$

Thus, if $X^\top X$ is invertible, the equation above has a unique solution (the one we found in the previous section! 1.1.2) and thus, our *empirical risk minimizer* $\hat{f}$ is given by

$$\hat{f} : x\prime \mapsto x\prime^\top \left(X^\top X\right)^{-1} X^\top y$$

Definition 2.3. A function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be *differentiably convex* if it is both differentiable and convex.

- **Differentiable:** f is differentiable at $x \in \mathbb{R}^d$ if the gradient $\nabla f(x)$ exists, i.e. there is a linear map $g : \mathbb{R}^d \to \mathbb{R}$ such that

$$\lim_{h\to 0}\frac{f(x+h) - f(x) - g(h)}{\|h\|} = 0.$$

- **Convex:** f is convex if for all $x, y \in \mathbb{R}^d$ and $t \in [0, 1]$,

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y).$$

  Equivalently, if f is differentiable, convexity is equivalent to the first-order condition

$$f(y) \geq f(x) + \nabla f(x)^\top (y - x) \quad \forall x, y \in \mathbb{R}^d.$$

- **Strict convexity:** f is strictly convex if for all $x \neq y$ and $t \in (0, 1)$,

$$f(tx + (1-t)y) < tf(x) + (1-t)f(y).$$

  In the differentiable case, this implies that the first-order inequality is strict whenever $y \neq x$.

Thus, a differentiably convex function is one that admits a gradient and satisfies the convexity inequalities.

slight problemo amigo: Whenever $p > n$, $X^\top X$ is not invertible, this gives our equation multiple solutions (not unique anymore).

Linear regression v. Affine regression

$$f_w(x) = w^\top x \text{ v. } f_{w,b}(x) = w^\top x + b = \hat{w}^\top \hat{x}$$

where

$$\hat{w} = \begin{bmatrix} w \\ b \end{bmatrix}, \ \hat{x} = \begin{bmatrix} x \\ 1 \end{bmatrix}$$

Remark 2.4. Wowoweeewa, this shows that an affine regression of dimention p is just a linear regresion of dimention $p+1$! Then, these two models are equivalent if we do not *regularize* (this is true because b is usually not regularized).

Hat matrix and porn (geometry of linear regression)

If $X \in \mathbb{R}^{n \times p}$ has full column rank, then the OLS estimator is

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

so that for the training data we obtain

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

with

$$H = X(X^\top X)^{-1} X^\top.$$

Let $r = \text{rank}(X)$. Consider the eigenvalue decomposition of $XX^\top$ in reduced form:

$$XX^\top = USU^\top,$$

where

- $U \in \mathbb{R}^{n \times r}$ is an orthonormal matrix (its columns form an orthonormal basis for $\text{Im}(X)$),
- $S \in \mathbb{R}^{r \times r}$ is a diagonal matrix with strictly positive entries.

Then one can show that

$$H = UU^\top.$$

Geometric interpretation: The hat matrix H (called hat matrix cuz' it maps y to $\hat{y}$ hence adding a hat to it) is the orthogonal projector onto $\text{Im}(X)$. The geometry produced by this projection may be observed in the following figure 1

This means:

- The fitted vector $\hat{y} = Hy$ is the projection of y onto the subspace $\text{Im}(X)$ (the span of the columns of X).
- The residual vector $y - \hat{y}$ lies in $\text{Im}(X)^\perp$.

In the case $X = [x^{(1)}\ x^{(2)}] \in \mathbb{R}^{n\times 2}$, the column space $\mathrm{Im}(X)$ is the plane spanned by $x^{(1)}$ and $x^{(2)}$. The observed vector y is projected onto this plane, producing $\hat{y} = Hy$, and the residual $y - \hat{y}$ is orthogonal to it.

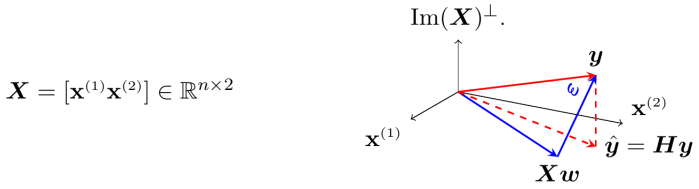


Figure 1: Geometry of linear regression

Optimality of least squares linear regression

Assuming that $y = X\beta + \varepsilon$ where $\mathrm{rank}(X) = p$ i.e. full rank matrix and *decorrelated centered noise* $\mathbb{E}[\varepsilon] = 0;\ \mathbb{E}[\varepsilon^\top \varepsilon] = \sigma^2 I$.

Theorem 2.5. Gauss-Markov Theorem: Assuming the previous statements, $\hat{\beta} = \left(X^\top X\right)^{-1} X^\top y$ is the *best linear unbiased estimator* (BLUE). That is, for any other *unbiased* estimator $\tilde{\beta}$ we have

$$\mathrm{Cov}(\tilde{\beta}) = \mathrm{Cov}(\hat{\beta}) + K_{\tilde{\beta}}$$

where $K_{\tilde{\beta}}$ is a *possitive semi-definite* matrix.

Remark 2.6 (Positive (semi)definiteness)**.** A symmetric matrix $M \in \mathbb{R}^{d\times d}$ is called

- *positive semidefinite (psd)* if

$$z^\top M z \ \geq\ 0 \quad \forall z \in \mathbb{R}^d,$$

  written $M \succeq 0$.

- *positive definite (pd)* if

$$z^\top M z \ >\ 0 \quad \forall z \in \mathbb{R}^d \setminus \{0\},$$

  written $M \succ 0$.

Equivalently:

- $M \succeq 0$ if and only if all eigenvalues of M are nonnegative,
- $M \succ 0$ if and only if all eigenvalues of M are strictly positive.

Gaussian conditional model and least squares regression

We can model the conditional distribution of Y given X as

$$Y \mid X \sim \mathcal{N}(\beta^\top X, \sigma^2).$$

- Likelihood for one observation (x_i, y_i):

$$p(y_i \mid x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(y_i - \beta^\top x_i)^2\right).$$

- Negative log-likelihood:

$$-\ell(\beta, \sigma^2) = \sum_{i=1}^{n} -\log p(y_i \mid x_i) = \frac{n}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \beta^\top x_i)^2.$$

Minimization over both β and σ^2 yields:

$$\min_{\sigma^2,\beta} \ \frac{n}{2}\log(2\pi\sigma^2) + \frac{1}{2\sigma^2}\|y - X\beta\|_2^2.$$

Remark 2.7. For fixed σ^2, minimizing in β is equivalent to the usual least squares problem:

$$\min_{\beta} \ \frac{1}{2\sigma^2}\|y - X\beta\|_2^2.$$

Optimizing over σ^2 gives the maximum-likelihood estimate:

$$\hat{\sigma}^2_{\text{MLE}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{\beta}_{\text{MLE}}^\top x_i)^2.$$

Properties if the model is well-specified

Assume

$$y = X\beta^* + \varepsilon,$$

with full column rank design matrix ($\text{rank}(X) = p$, thus $n \geq p$) and i.i.d. centered Gaussian noise $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$.

Proposition 2.8 (Distributional properties)**.** Under these assumptions:

- $\hat{\beta} = (X^\top X)^{-1}X^\top y \ \sim \ \mathcal{N}\big(\beta^*, \sigma^2(X^\top X)^{-1}\big)$
- $S^2 = \frac{1}{n-p}\|y - \hat{y}\|_2^2 \ \sim \ \sigma^2 \cdot \frac{1}{n-p}\chi^2_{n-p}$
- $\hat{\beta} \perp\!\!\!\perp S^2$

Remark 2.9. These distributional facts are the foundation for classical inference tools: ANOVA, t-tests, and confidence intervals. They are only valid if the Gaussian linear model is correctly specified (i.e. the noise is truly Gaussian and homoscedastic).

3 Overfitting, regularization and complexity

In this section, we explore how nature can "trick" us when we wield the *tools of god*—namely, polynomials—to fit our models to data. Polynomial regression is still a linear regression model (linear in the parameters), but by expanding the feature space with polynomial terms we can capture much more *sensual*, non-linear structures in x.

This apparent power comes with a catch: high-degree polynomials are so flexible that they can bend to pass through almost every training point, including noise, without actually capturing the underlying structure we care about. In other words, complexity can become our downfall: more degrees do not always mean better generalization. (Moral: do not be a greedy sonnovabitch!)

Polynomial regression

Polynomial regression is an instance of linear regression:

$$Y = w_0 + w_1 X + w_2 X^2 + \cdots + w_p X^p + \varepsilon,$$

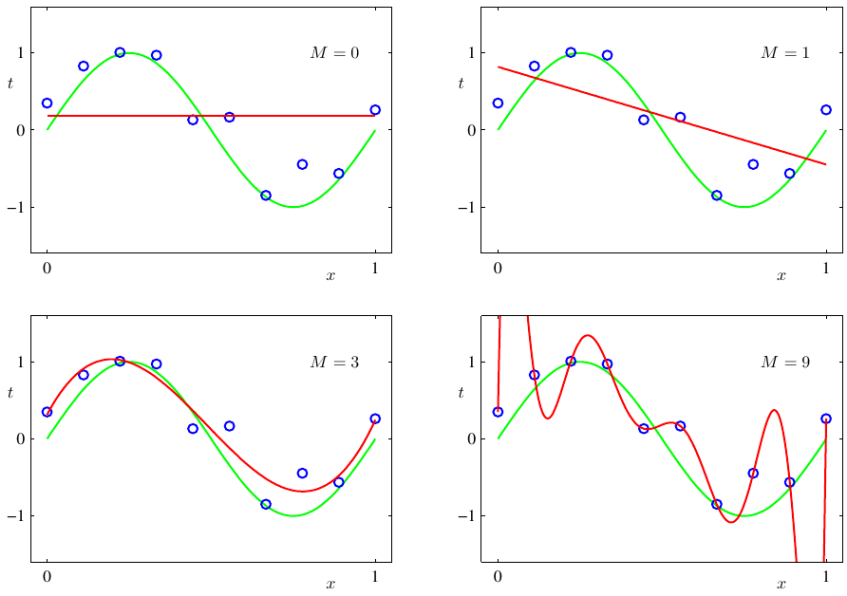
where

$$\min_w \frac{1}{2n} \sum_{i=1}^{n} \Big(y_i - (w_0 + w_1 x_i + w_2 x_i^2 + \cdots + w_p x_i^p) \Big)^2,$$

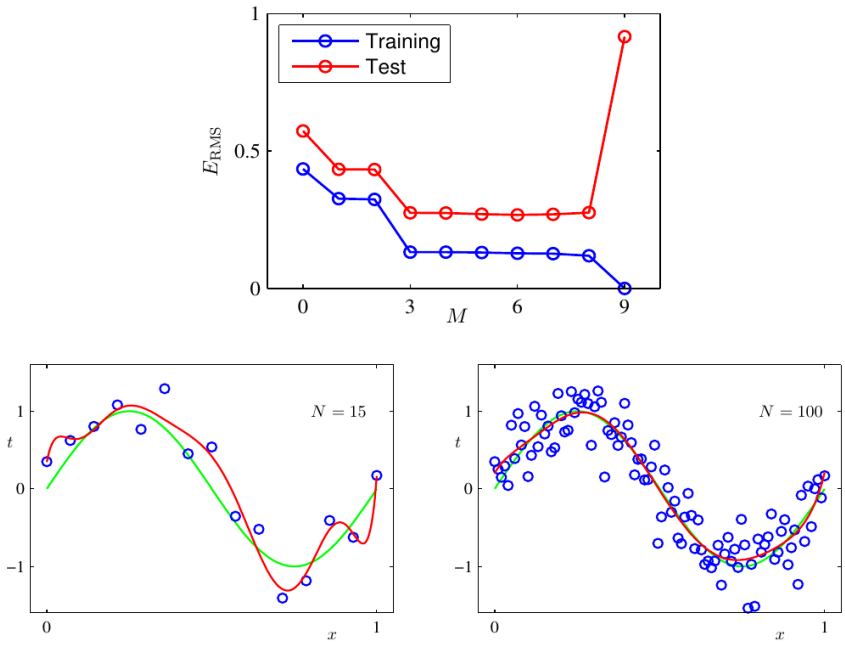
with $\deg(Y) = p = M$.

We present some examples for these type of regressions and choose different *degrees* for our polynomial in order to see explicitly the trickery we've discussed.

- $M = 0, 1, 3, 9$



Overfitting: symptoms and characteristics



This trickery we've talked about is formally known as *overfitting*. Most of the times, it posseses some of the characteristics we descibe below (which are clearly visible from our figures above).

- Training error decreasing *monotonically* (i.e. consistently decreasing) as complexity increases.
- Test error decreasing initially, then increasing as the model begins to fit noise.

Overfitting and generalization

In machine learning/statistics, the main concern is the *generalization ability* of the predictor.

Fitting the data perfectly does *not always* mean poor generalization. For example: deep neural networks in computer vision often fit perfectly and still generalize well.

However, fitting perfectly *is problematic* if:

- The data is noisy and the model fits the noise.
- The model must become overly complex in order to fit the data.

So, we confront a big and intimidating wall which may as well be reduced if we're able to answer the following question: How do we measure **complexity**?

Tikhonov regularization

Courtesy of Andrey N. Tikhonov (1906–1993).

Regularized Empirical Risk Minimization (ERM) objective:

$$\min_{f \in S} \hat{R}_n(f) + \lambda \|f\|^2,$$

where λ is the *regularization* coefficient or *hyperparameter*.

Proposition 3.1. How to know if the above objective is well posed?

- If $\hat{R}_n$ is convex:
  1. $\implies$ objective is strongly convex and coercive for any $\lambda > 0$.
  2. $\implies$ solution exists and is unique.
  3. $\implies$ $\lambda \mapsto \hat{f}_\lambda$ is a continuous function.
- If $\hat{R}_n$ is only bounded below:
  1. $\implies$ objective is coercive and at least one solution exists.
- If $\hat{R}_n$ is C^2 with bounded curvature:
  1. $\implies$ regularization eliminates small local minima.

Well, since that is a mouthfull, let us digress with an insightfull remark.

Remark 3.2 (Strong convexity and coercivity)**.** Let $f : \mathbb{R}^d \to \mathbb{R}$ be differentiable.

- **Convex:** f is convex if for all $x, y \in \mathbb{R}^d$ and $t \in [0, 1]$,

$$f(tx + (1-t)y) \;\leq\; tf(x) + (1-t)f(y).$$

- **Strongly convex:** f is *μ-strongly convex* ($\mu > 0$) if

$$f(y) \;\geq\; f(x) + \nabla f(x)^\top (y - x) + \frac{\mu}{2}\|y - x\|^2 \quad \forall x, y \in \mathbb{R}^d.$$

  Intuitively, the graph of f always curves upwards at least as much as a quadratic bowl of curvature μ. Strong convexity implies that the minimizer of f is *unique*.

- **Coercive:** f is coercive if

$$\|x\| \to \infty \quad \implies \quad f(x) \to +\infty.$$

  This condition ensures that a minimizer cannot "escape to infinity", so a minimizer exists (at least one).

In ridge regression: the regularized risk

$$\hat{R}_n(w) + \frac{\lambda}{2}\|w\|^2$$

is λ-strongly convex and coercive for any $\lambda > 0$, hence it always admits a unique solution.

Ridge regression

Applying Tikhonov regularization to OLS:

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

Normal equations:

$$\left(\tfrac{1}{n}X^\top X + \lambda I\right)w = \tfrac{1}{n}X^\top y,$$

so the unique solution is:

$$\hat{w}_{\text{ridge}} = \frac{1}{n}\left(\tfrac{1}{n}X^\top X + \lambda I\right)^{-1} X^\top y.$$

Remark 3.3 (Normal equations)**.** In ordinary least squares we minimize

$$\min_{w\in\mathbb{R}^p} \|y - Xw\|_2^2.$$

Setting the gradient to zero yields

$$X^\top Xw = X^\top y,$$

a system of linear equations called the *normal equations*. They are called "normal" because the residual vector $r = y - X\hat{w}$ is orthogonal (normal) to the column space of X, **not** because of any assumption about normally distributed noise.

Linear vs affine regression and regularization

$$f_w(x) = w^\top x \quad \text{vs} \quad f_{w,b}(x) = w^\top x + b = \tilde{w}^\top \tilde{x},$$

with

$$\tilde{w} = \begin{bmatrix} w \\ b \end{bmatrix}, \qquad \tilde{x} = \begin{bmatrix} x \\ 1 \end{bmatrix}.$$

Thus, an affine model in dimension p is a linear model in dimension $p+1$.

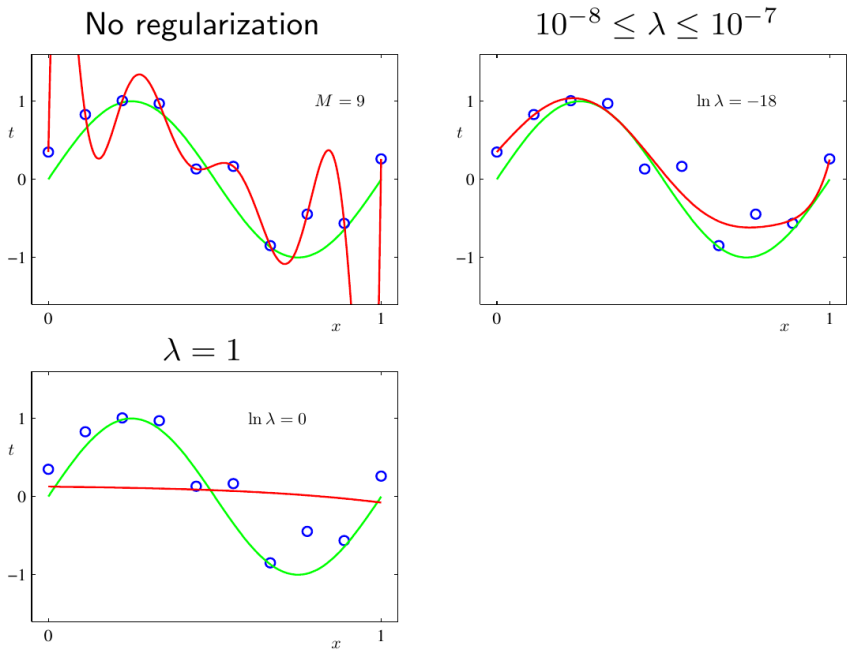
Difference: When regularizing, usually b is not penalized. So minimizing

$$\min_{w\in\mathbb{R}^p} \frac{1}{2n}\|y - Xw + b1\|_2^2 + \frac{\lambda}{2}\|w\|_2^2$$

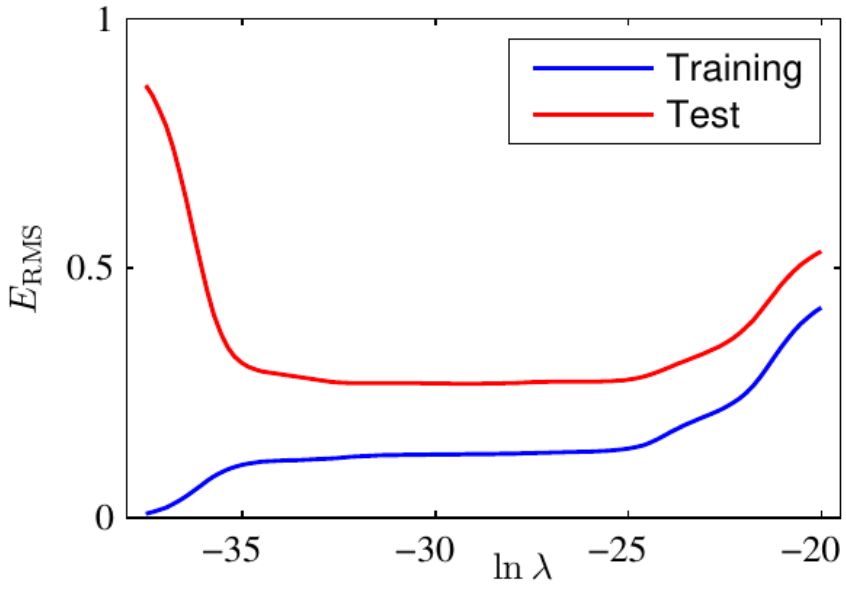
is not equivalent to penalizing $\tilde{w}$.

Polynomial regression with ridge

We now explore an example of polynomial regression for the parameters $n = 10$, degree $M = 9$ (look how different the behaviour is from not regularizing!). We also compare no regularization with that of ridge regularization for different values of λ.



Also, note the difference between our train v. test error evolution with and without regularization. It is now more stable for greater degrees of the polynomial.



Controlling complexity of the hypothesis space

Explicit control:

- number of variables,
- max polynomial degree,
- spline degree and number of knots,
- max resolution in wavelet approximation,

- bandwidth in RKHS.

Implicit control:

- via regularization,
- Bayesian formulations,
- optimization algorithm itself,
- randomization, etc.

The complexity of the predictor often results from a trade-off between goodness of fit and complexity. **Model selection problem:** how to choose the right level of complexity?

Risk decomposition: approximation–estimation trade-off

Let f^* be the target function, $f_S^* = \arg\min_{f \in S} R(f)$, and $\hat{f}_S$ the ERM predictor in S.

$$R(\hat{f}_S) - R(f^*) \;=\; \underbrace{R(\hat{f}_S) - R(f_S^*)}_{\text{estimation error}} + \underbrace{R(f_S^*) - R(f^*)}_{\text{approximation error}} .$$

This is sometimes called the *bias–variance trade-off*.

Approximation–estimation trade-off

There is generally a compromise:

- fitting the training data well,
- avoiding a too complex model to ensure generalization.

However, this view has been challenged by modern neural networks.

References

[1] D. S. Judson, *Abstract Algebra: Theory and Applications*, 3rd ed. Orthogonal Publishing L3C, 2019.