# Mathematical foundations for ML - UM6P class

Calculus

05 February 2023

## Contents

# 1 Introducing basic concepts

## 1.1 Basic Notations and Properties

**Convention:** Mathematically, the probability that a random variable $X$ takes the value $x$ is denoted $p(X = x)$. In this document, we simplify this notation to $p(X)$ to denote a distribution over the random variable $X$, or $p(x)$ to denote the distribution evaluated for the particular value $x$. This simplification applies similarly for more variables.

### 1.1.1 Fundamental Rules

**Proposition 1.1.1.** *For two random variables $X, Y$, the following fundamental rules apply:*

- ***Sum Rule:***
$$p(X) = \sum_Y p(X, Y)$$
- ***Product Rule:***
$$p(X, Y) = p(Y \mid X)p(X)$$

### 1.1.2 Independence

**Definition 1.1.** *Two random variables $X$ and $Y$ are said to be independent if and only if:*

$$P(X, Y) = P(X)P(Y)$$

### 1.1.3 Conditional Independence

**Definition 1.2.** *Let $X, Y, Z$ be random variables. We define $X$ and $Y$ to be conditionally independent given $Z$ if and only if:*

$$P(X, Y \mid Z) = P(X \mid Z)P(Y \mid Z)$$

**Proposition 1.1.2.** *If $X$ and $Y$ are conditionally independent given $Z$, then:*

$$P(X \mid Y, Z) = P(X \mid Z)$$

### 1.1.4 Independent and Identically Distributed

**Definition 1.3.** *A set of random variables is independent and identically distributed (i.i.d.) if each random variable has the same probability distribution as the others and all are mutually independent.*

### 1.1.5 Bayes' Formula

**Proposition 1.1.3.** *For two random variables $X, Y$, Bayes' formula is given by:*

$$P(X \mid Y) = \frac{P(Y \mid X)P(X)}{P(Y)}$$

## 1.2 Matrix Calculus

In this section we present some basic definitions of matrix calculus and provide a few examples.

### 1.2.1 The Gradient

**Definition 1.4.** *Suppose that $f : \mathbb{R}^{m \times n} \to \mathbb{R}$ is a function that takes as input a matrix $A$ of size $m \times n$ and returns a real value. Then the gradient of $f$ (with respect to $A \in \mathbb{R}^{m \times n}$ ) is the matrix of partial derivatives, defined as:*

$$\nabla_A f(A) \in \mathbb{R}^{m \times n} = \begin{bmatrix} \frac{\partial f(A)}{\partial A_{11}} & \frac{\partial f(A)}{\partial A_{12}} & \cdots & \frac{\partial f(A)}{\partial A_{1n}} \\ \frac{\partial f(A)}{\partial A_{21}} & \frac{\partial f(A)}{\partial A_{22}} & \cdots & \frac{\partial f(A)}{\partial A_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f(A)}{\partial A_{m1}} & \frac{\partial f(A)}{\partial A_{m2}} & \cdots & \frac{\partial f(A)}{\partial A_{mn}} \end{bmatrix}$$

*i.e., an $m \times n$ matrix with*

$$(\nabla_A f(A))_{ij} = \frac{\partial f(A)}{\partial A_{ij}}$$

Note that the size of $\nabla_A f(A)$ is always the same as the size of $A$. So if, in particular, $A$ is just a vector $x \in \mathbb{R}^n$,

$$\nabla_x f(x) = \begin{bmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{bmatrix}$$

It is very important to remember that the gradient of a function is only defined if the function is real-valued, that is, if it returns a scalar value. We can not, for example, take the gradient of $Ax, A \in \mathbb{R}^{n \times n}$ with respect to $x$, since this quantity is vector-valued.

**Proposition 1.2.1.** *Properties of partial derivatives*

- $\nabla_x(f(x) + g(x)) = \nabla_x f(x) + \nabla_x g(x)$.
- *For* $t \in \mathbb{R}, \nabla_x(tf(x)) = t\nabla_x f(x)$.

### 1.2.2 The Hessian

**Definition 1.5.** *Suppose that* $f : \mathbb{R}^n \to \mathbb{R}$ *is a function that takes a vector in* $\mathbb{R}^n$ *and returns a real number. Then the Hessian matrix with respect to* $x$*, written* $\nabla_x^2 f(x)$ *or simply as* $H$ *is the* $n \times n$ *matrix of partial derivatives,*

$$\nabla_x^2 f(x) \in \mathbb{R}^{n \times n} = \begin{bmatrix} \frac{\partial^2 f(x)}{\partial x_1^2} & \frac{\partial^2 f(x)}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 f(x)}{\partial x_2^2} & \cdots & \frac{\partial^2 f(x)}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f(x)}{\partial x_n \partial x_1} & \frac{\partial^2 f(x)}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f(x)}{\partial x_n^2} \end{bmatrix}$$

*In other words,* $\nabla_x^2 f(x) \in \mathbb{R}^{n \times n}$*, with*

$$\left(\nabla_x^2 f(x)\right)_{ij} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}$$

Note that the Hessian is always symmetric, since

$$\frac{\partial^2 f(x)}{\partial x_i \partial x_j} = \frac{\partial^2 f(x)}{\partial x_j \partial x_i}$$

Similar to the gradient, the Hessian is defined only when $f(x)$ is real-valued. It is natural to think of the gradient as the analogue of the first derivative for functions of vectors, and the Hessian as the analogue of the second derivative (and the symbols we use also suggest this relation). Finally, note that while we can take the gradient with respect to a matrix $A \in \mathbb{R}^n$, for the purposes of this class we will only consider taking the Hessian with respect to a vector $x \in \mathbb{R}^n$. This is simply a matter of convenience (and the fact that none of the calculations we do require us to find the Hessian with respect to a matrix), since the Hessian with respect to a matrix would have to represent all the partial derivatives $\partial^2 f(A)/\left(\partial A_{ij} \partial A_{k\ell}\right)$, and it is rather cumbersome to represent this as a matrix.

### 1.2.3 Gradients and Hessians of Quadratic and Linear Functions

Now let's try to determine the gradient and Hessian matrices for a few simple functions.

**Exercise:**
Let $b \in \mathbb{R}^n$ and $f : x \mapsto b^T x$. Caluclate $\nabla_x f(x)$

**Solution:**
For $x \in \mathbb{R}^n$, let $f(x) = b^T x$ for some known vector $b \in \mathbb{R}^n$. Then

$$f(x) = \sum_{i=1}^{n} b_i x_i$$

so

$$\frac{\partial f(x)}{\partial x_k} = \frac{\partial}{\partial x_k} \sum_{i=1}^{n} b_i x_i = b_k$$

From this we can easily see that $\nabla_x b^T x = b$. This should be compared to the analogous situation in single variable calculus, where $\partial/(\partial x) ax = a$.

**Exercise:** N ow consider the quadratic function $f(x) = x^T A x$ for $A \in \mathbb{S}^n$. Calculate:

$$\nabla_x f(x) \quad \textbf{and} \quad \nabla_x^2 f(x)$$

**Solution:**

Remember that

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} x_i x_j$$

To take the partial derivative, we'll consider the terms including $x_k$ and $x_k^2$ factors separately:

$$\begin{aligned}
\frac{\partial f(x)}{\partial x_k} &= \frac{\partial}{\partial x_k} \sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij} x_i x_j \\
&= \frac{\partial}{\partial x_k} \left[ \sum_{i \neq k} \sum_{j \neq k} A_{ij} x_i x_j + \sum_{i \neq k} A_{ik} x_i x_k + \sum_{j \neq k} A_{kj} x_k x_j + A_{kk} x_k^2 \right] \\
&= \sum_{i \neq k} A_{ik} x_i + \sum_{j \neq k} A_{kj} x_j + 2A_{kk} x_k \\
&= \sum_{i=1}^{n} A_{ik} x_i + \sum_{j=1}^{n} A_{kj} x_j = 2 \sum_{i=1}^{n} A_{ki} x_i,
\end{aligned}$$

where the last equality follows since $A$ is symmetric (which we can safely assume, since it is appearing in a quadratic form). Note that the $k$ th entry of $\nabla_x f(x)$ is just the inner product of the $k$ th row of $A$ and $x$. Therefore, $\nabla_x x^T A x = 2Ax$. Again, this should remind you of the analogous fact in single-variable calculus, that $\partial/(\partial x) a x^2 = 2ax$.

Finally, let's look at the Hessian of the quadratic function $f(x) = x^T A x$ (it should be obvious that the Hessian of a linear function $b^T x$ is zero). In this case,

$$\frac{\partial^2 f(x)}{\partial x_k \partial x_\ell} = \frac{\partial}{\partial x_k} \left[ \frac{\partial f(x)}{\partial x_\ell} \right] = \frac{\partial}{\partial x_k} \left[ 2 \sum_{i=1}^{n} A_{\ell i} x_i \right] = 2A_{\ell k} = 2A_{k\ell}$$

Therefore, it should be clear that $\nabla_x^2 x^T A x = 2A$, which should be entirely expected (and again analogous to the single-variable fact that $\partial^2 / \left(\partial x^2\right) a x^2 = 2a$).

To recap,

**Proposition 1.2.2.** *We have the following properties:*

- $\nabla_x \left(b^T x\right) = b$
- $\nabla_x \left(x^T A x\right) = 2Ax$ *(if $A$ symmetric)*

- $\nabla_x^2 \left(x^T A x\right) = 2A$ *(if $A$ symmetric)*

## 1.3 Review on differentials

**Definition 1.6.** *Differentiable function. A function* f : $\mathbb{R}^m \to \mathbb{R}^n$ *is differentiable at* $a \in \mathbb{R}^m$ *iff there exists a linear map* $\phi_a$ *such that:*

$$f(a+h) - f(a) = \phi_a(h) + o(\|h\|)$$

*We write:*

$$\forall h \in \mathbb{R}^m \quad \phi_a(h) = df_a(h)$$

If $n = 1$ and since $R^m$ is a Hilbert space, we know that there exists $g \in \mathbb{R}^m$ such that $df_a(h) = \langle g, h \rangle$. We call $g$ the gradient of $f$ at $a$. We write: $g = \nabla f(a)$.

**Exercise:** i f $f \mapsto a^\top x + b$, calculate $\nabla f(x)$

**Solution:** if $f \mapsto a^\top x + b$ then we have :

$$f(x+h) - f(x) = a^\top h$$

and thus

$$\nabla f(x) = a$$

**Exercise:** if $fx \mapsto x^\top A x$, calculate $\nabla f(x)$

**Solution:**

$$\begin{aligned} f(x+h) - f(x) &= (x+h)^T A(x+h) - x^\top A x \\ &= x^\top A h + h^\top A x + o(\|h\|) \end{aligned}$$

The gradient is then :

$$\nabla f(x) = \left(A + A^\top\right) x$$

**Exercise:**
if $f : A$ positive definite $\mapsto \log\det(A)$, calculate $\nabla f(A)$

**Solution:**

- **Method 1**

$$|A| = \sum_{i=1}^{n} (-1)^{i+j} A_{ij} \left|A_{\setminus i,\setminus j}\right| \quad (\text{ for any } j \in 1, \ldots, n)$$

So,

$$\begin{aligned}\frac{\partial}{\partial A_{k\ell}}|A| &= \frac{\partial}{\partial A_{k\ell}} \sum_{i=1}^{n} (-1)^{i+j} A_{ij} \left|A_{\setminus i,\setminus j}\right| \\ &= (-1)^{k+\ell} \left|A_{\setminus k,\setminus \ell}\right| \\ &= (\text{adj}(A))_{\ell k}\end{aligned}$$

From this it immediately follows from the properties of the adjoint that

$$\nabla_A |A| = (\text{adj}(A))^T = |A| A^{-T}$$

Note that we have to restrict the domain of $f$ to be the positive definite matrices, since this ensures that $|A| > 0$, so that the log of $|A|$ is a real number. In this case we can use the chain rule (nothing fancy, just the ordinary chain rule from single-variable calculus) to see that

$$\frac{\partial \log|A|}{\partial A_{ij}} = \frac{\partial \log|A|}{\partial |A|} \frac{\partial |A|}{\partial A_{ij}} = \frac{1}{|A|} \frac{\partial |A|}{\partial A_{ij}}$$

From this it should be obvious that

$$\nabla_A \log|A| = \frac{1}{|A|} \nabla_A |A| = A^{-1}$$

where we can drop the transpose in the last expression because $A$ is symmetric. Note the similarity to the single-valued case, where $\partial/(\partial x) \log x = 1/x$.

- **Method 2**

  Let's define $\widetilde{H} = \left(A^{-\frac{1}{2}}\right) H A^{-\frac{1}{2}}$.

  $\widetilde{H}$ is symmetric, so it can be written as :

$$\widetilde{H} = U \Lambda U^\top$$

  where U is an orthogonal matrix and $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_d)$.

  We have:

$$\begin{aligned}
\log(\det(A+H)) &= \log\left(\det\left(A^{\frac{1}{2}}\left(I + A^{-\frac{1}{2}} H A^{-\frac{1}{2}}\right) A^{\frac{1}{2}}\right)\right) \\
&= \log(\det(A)) + \log(\det(I + \widetilde{H})) \\
&= \log(\det(A)) + \sum_{i=1}^{n} \log(1+\lambda_i) \\
&= \log(\det(A)) + \sum_{i=1}^{n} \lambda_i + o(\|H\|) \\
&= \log(\det(A)) + \operatorname{tr}(\widetilde{H}) + o(\|H\|) \\
&= \log(\det(A)) + \operatorname{tr}\left(\left(A^{-\frac{1}{2}}\right) H A^{-\frac{1}{2}}\right) + o(\|H\|) \\
&= \log(\det(A)) + \operatorname{tr}\left(\left(A^{-1}\right)^\top H\right) + o(\|H\|) \\
&= \log(\det(A)) + \left\langle A^{-1}, H \right\rangle + o(\|H\|)
\end{aligned}$$

  We deduce the gradient of $\log(\det(A))$:

$$\nabla \log(\det(A)) = A^{-1}$$

## 1.4 Review on Lagrange duality

- **Lagrangian**

**Definition 1.7.** *Consider the following convex optimization problem:*

$$\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}), \text{ subject to } \mathbf{A}\mathbf{x} = \mathbf{b}$$

*where $f$ is a convex function, $\mathcal{X} \subset \mathbb{R}^p$ is a convex set included in the domain of $f$, $\mathbf{A} \in \mathbb{R}^{n \times p}, \mathbf{b} \in \mathbb{R}^n$.*

*The Lagrangian associated with this optimization problem is defined as*

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{x} - \mathbf{b})$$

*The vector $\boldsymbol{\lambda} \in \mathbb{R}^n$ is called the Lagrange multiplier vector.*

- **Lagrange dual function**

**Definition 1.8.** *The Lagrange dual function is defined as*

$$g(\boldsymbol{\lambda}) = \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) \tag{1}$$

*The problem of maximizing $g(\boldsymbol{\lambda})$ with respect to $\boldsymbol{\lambda}$ is known as the Lagrange dual problem.*

- **Max-min inequality**

**Proposition 1.4.1.** *For any $f : \mathbb{R}^n \times \mathbb{R}^m$ and any $w \in \mathbb{R}^n$ and $z \in \mathbb{R}^m$, we have*

$$\begin{aligned} f(w,z) \leq \max_{z \in Z} f(w,z) &\implies \min_{w \in W} f(w,z) \leq \min_{w \in W} \max_{z \in Z} f(w,z) \\ &\implies \max_{z \in Z} \min_{w \in W} f(w,z) \leq \min_{w \in W} \max_{z \in Z} f(w,z). \end{aligned}$$

*The last inequality is known as the max-min inequality.*

$$\max_{z \in Z} \min_{w \in W} f(w,z) \leq \min_{w \in W} \max_{z \in Z} f(w,z) \tag{2}$$

- **Duality**

**Proposition 1.4.2.**

$$\max_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) = \begin{cases} f(\mathbf{x}) & \text{if } \mathbf{A}\mathbf{x} = \mathbf{b} \\ +\infty & \text{otherwise} \end{cases}$$

*Which gives us*

$$\min_{\mathbf{x}} f(\mathbf{x}) = \min_{\mathbf{x}} \max_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) \tag{3}$$

- **Weak and strong duality**

Now from 1, 2 and 3 we have

$$\max_{\boldsymbol{\lambda}} g(\boldsymbol{\lambda}) = \max_{\boldsymbol{\lambda}} \min_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) \leq \min_{\mathbf{x}} \max_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) = \min_{\mathbf{x}} f(\mathbf{x}) \tag{4}$$

The inequality 4 says that the optimal value $d^*$ of the Lagrange dual problem always lower-bounds the optimal value $p^*$ of the original problem. This property is called the **weak duality**.

If the equality $d^* = p^*$ holds, then we say that the strong duality holds. Strong duality means that the order of the minimization over $\mathbf{x}$ and the maximization over $\boldsymbol{\lambda}$ can be switched without affecting the result.

- **Slater's constraint qualification lemma.**

  **Lemma 1.4.3.** *If there exists an* $\mathbf{x}$ *in the relative interior of* $\mathcal{X} \cap \{\mathbf{Ax} = \mathbf{b}\}$ *then strong duality holds.*

  (Note that by definition $\mathcal{X}$ is included in the domain of $f$ so that if $\mathbf{x} \in \mathbf{X}$ then $f(\mathbf{x}) < \infty$.)

  For a more general problem and more details about Lagrange duality, please refer to [1] (chapter 5).

# 2 Parameter estimation by maximum likelihood

## 2.1 Statistical Models

**Definition 2.1.** *A (parametric) statistical model* $\mathcal{P}_\Theta$ *is a collection of probability distributions (or a collection of probability density functions) defined on the same space and parameterized by parameters* $\theta$ *belonging to a set* $\Theta \subset \mathbb{R}^p$*. Formally:*

$$\mathcal{P}_\Theta = \{p_\theta(\cdot) \mid \theta \in \Theta\}$$

### 2.1.1 Bernoulli model

Consider a binary random variable $X$ that can take the value 0 or 1 . If $p(X = 1)$ is parametrized by $\theta \in [0, 1]$ :

$$\begin{cases} \mathbb{P}(X = 1) = \theta \\ \mathbb{P}(X = 0) = 1 - \theta \end{cases}$$

then a probability distribution of the Bernoulli model can be written as

$$p(X = x; \theta) = \theta^x (1 - \theta)^{1-x}$$

and we can write

$$X \sim \text{Ber}(\theta).$$

The Bernoulli model is the collection of these distributions for $\theta \in \Theta = [0, 1]$.

### 2.1.2 Binomial model

A binomial random variable $\text{Bin}(\theta, N)$ is defined as the value of the sum of $n$ i.i.d. Bernoulli r.v. with parameter $\theta$. The distribution of a binomial random variable $N$ is

$$\mathbb{P}(N = k) = \left( \begin{array}{c} n \\ k \end{array} \right) \theta^k (1 - \theta)^{n-k} \tag{5}$$

The set $\Theta$ is the same as for the Bernoulli model.

### 2.1.3 Multinomial model

Consider a discrete random variable $C$ that can take one of $K$ possible values $\{1, 2, \ldots, K\}$. The random variable $C$ can be represented by a $K$-dimensional random variable $X = (X_1, X_2, \ldots, X_K)^T$ for which the event $\{C = k\}$ corresponds to the event

$$\{X_k = 1 \text{ and } X_l = 0, \forall l \neq k\}.$$

If we parametrize $\mathbb{P}(C = k)$ by a parameter $\pi_k \in [0, 1]$, then by definition we also have

$$\mathbb{P}(X_k = 1) = \pi_k \quad \forall k = 1, 2, \ldots, K,$$

with $\sum_{k=1}^{K} \pi_k = 1$. The probability distribution over $\mathbf{x} = (x_1, \ldots, x_k)$ can be written as

$$p(\mathbf{x}; \boldsymbol{\pi}) = \prod_{k=1}^{K} \pi_k^{x_k} \tag{6}$$

where $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_K)^T$. We will denote $\mathcal{M}(1, \pi_1, \ldots, \pi_K)$ such a discrete distribution. The corresponding set of parameters is

$$\Theta = \left\{ \boldsymbol{\pi} \in \mathbb{R}^+ \mid \sum_{k=1}^{K} \pi = 1 \right\}$$

Now if we consider $n$ independent observations of a $\mathcal{M}(1, \boldsymbol{\pi})$ multinomial random variable $X$, and we denote by $N_k$ the number of observations for which $x_k = 1$, then the joint distribution of $N_1, N_2, \ldots, N_K$ is called a multinomial $\mathcal{M}(n, \boldsymbol{\pi})$ distribution. It takes the form:

$$p\left(n_1, n_2, \ldots, n_K; \boldsymbol{\pi}, n\right) = \frac{n!}{n_1! n_2! \ldots n_K!} \prod_{k=1}^{K} \pi_k^{n_k}$$

and we can write

$$\left(N_1, \ldots, N_K\right) \sim \mathcal{M}\left(N, \pi_1, \pi_2, \ldots, \pi_K\right)$$

The multinomial $\mathcal{M}(n, \boldsymbol{\pi})$ is to the $\mathcal{M}(1, \boldsymbol{\pi})$ distribution, as the binomial distribution is to the Bernoulli distribution. In the rest of this course, when we will talk about multinomial distributions, we will always refer to a $\mathcal{M}(1, \boldsymbol{\pi})$ distribution.

### 2.1.4 Gaussian models

The Gaussian distribution is also known as the normal distribution. In the case of a scalar variable $X$, the Gaussian distribution can be written in the form

$$\mathcal{N}\left(x; \mu, \sigma^2\right) = \frac{1}{\left(2\pi\sigma^2\right)^{1/2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{7}$$

where $\mu$ is the mean and $\sigma^2$ is the variance. For a $d$-dimensional vector $\mathbf{x}$, the multivariate Gaussian distribution takes the form

$$\mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{d/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})\right) \tag{8}$$

where $\boldsymbol{\mu}$ is a $d$-dimensional vector, $\boldsymbol{\Sigma}$ is a $d \times d$ symmetric positive definite matrix, and $|\boldsymbol{\Sigma}|$ denotes the determinant of $\boldsymbol{\Sigma}$. It is a well-known property that the parameter $\boldsymbol{\mu}$ is equal to the expectation of $X$ and that the matrix $\Sigma$ is the covariance matrix of $X$, which means that $\boldsymbol{\Sigma}_{ij} = \mathbb{E}\left[\left(X_i - \mu_i\right)\left(X_j - \mu_j\right)\right]$.

## 2.2 Maximum Likelihood Estimation

**Definition 2.2.** *Maximum likelihood estimation (MLE) is a method of estimating the parameters of a statistical model. Suppose we have a sample $x_1, x_2, \ldots, x_n$ of $n$ independent and identically distributed observations, coming from a distribution $p\left(x_1, x_2, \ldots, x_n; \theta\right)$ where $\theta$ is an unknown parameter (both $x_i$ and $\theta$ can be vectors). As the name suggests, the MLE finds the parameter $\hat{\theta}$ under which the data $x_1, x_2, \ldots, x_n$ are most likely:*

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} p\left(x_1, x_2, \ldots, x_n; \theta\right)$$

*The probability on the right-hand side in the above equation can be seen as a function of $\theta$ and can be denoted by $\mathcal{L}(\theta)$ :*

$$\mathcal{L}(\theta) = p\left(x_1, x_2, \ldots, x_n; \theta\right)$$

This function is called the likelihood.

As $x_1, x_2, \ldots, x_n$ are independent and identically distributed, we have

$$\mathcal{L}(\theta) = \prod_{i=1}^{n} p\left(x_i; \theta\right)$$

In practice it is often more convenient to work with the logarithm of the likelihood function, called the log-likelihood:

$$\ell(\theta) = \log \mathcal{L}(\theta) = \log \prod_{i=1}^{n} p\left(x_i; \theta\right) = \sum_{i=1}^{n} \log p\left(x_i; \theta\right) \tag{9}$$

Next, we will apply this method for the models presented previously. We assume that all the observations are independent and identically distributed in all of the remainder of this lecture.

## 2.3 Exercises:

**Exercise:** MLE for the Bernoulli model

**Solution:**

Consider $n$ observations $x_1, x_2, \ldots, x_n$ of a binary random variable $X$ following a Bernoulli distribution $\text{Ber}(\theta)$. From 5 and 9 we have

$$\begin{aligned}
\ell(\theta) &= \sum_{i=1}^{n} \log p\left(x_i; \theta\right) \\
&= \sum_{i=1}^{n} \log \theta^{x_i} (1-\theta)^{1-x_i} \\
&= N \log(\theta) + (n - N) \log(1-\theta)
\end{aligned}$$

where $N = \sum_{i=1}^{n} x_i$.

As $\ell(\theta)$ is strictly concave, it has a unique maximizer, and since the function is in addition differentiable, its maximizer $\hat{\theta}$ is the zero of its gradient $\nabla \ell(\theta)$ :

$$\nabla \ell(\theta) = \frac{\partial}{\partial \theta} \ell(\theta) = \frac{N}{\theta} - \frac{n - N}{1 - \theta}.$$

It is easy to show that $\nabla \ell(\theta) = 0 \iff \theta = \frac{N}{n}$. Therefore we have

$$\hat{\theta} = \frac{N}{n} = \frac{x_1 + x_2 + \cdots + x_n}{n}.$$

**Exercise:**
MLE for the Multinomial model

**Solution:** Consider $N$ observations $X_1, X_2, \ldots, X_N$ of a discrete random variable $X$ following a multinomial distribution $\mathcal{M}(1, \boldsymbol{\pi})$, where $\boldsymbol{\pi} = (\pi_1, \pi_2, \ldots, \pi_K)^T$. We denote $\mathbf{x}_i (i = 1, 2, \ldots, N)$ the $K$-dimensional vectors of 0 s and 1 s representing $X_i$, as presented in Section 2.1.3.

From 9 and 6 we have:

$$\begin{aligned}
\ell(\boldsymbol{\pi}) &= \sum_{i=1}^{N} \log p\left(\mathbf{x}_i; \boldsymbol{\pi}\right) \\
&= \sum_{i=1}^{N} \log \left( \prod_{k=1}^{K} \pi_k^{x_{ik}} \right) \\
&= \sum_{i=1}^{N} \sum_{k=1}^{K} x_{ik} \log \pi_k \\
&= \sum_{k=1}^{K} n_k \log \pi_k
\end{aligned}$$

where $n_k = \sum_{i=1}^{N} x_{ik}$ ( $n_k$ is therefore the number of observations of $x_k = 1$ ).

We need to maximize this quantity subject to the constraint:

$$\sum_{k=1}^{K} \pi_k = 1$$

We need to minimize

$$f(\boldsymbol{\pi}) = -\ell(\boldsymbol{\pi}) = -\sum_{k=1}^{K} n_k \log \pi_k$$

subject to the constraint $\mathbf{1}^T \boldsymbol{\pi} = 1$.

The Lagrangian of this problem is

$$L(\boldsymbol{\pi}, \lambda) = -\sum_{k=1}^{K} n_k \log \pi_k + \lambda \left( \sum_{k=1}^{K} \pi_k - 1 \right)$$

Clearly, as $n_k \geq 0 (k = 1, 2, \ldots, K)$, $f$ is convex and this problem is a convex optimization problem. Moreover, it is trivial that there exist $\pi_1, \pi_2, \ldots, \pi_K$ such that $\pi_k > 0$ ( $k =$
$1, 2, \ldots, K$) and $\sum_{k=1}^{K} \pi_k = 1$, so by Slater's constraint qualification, the problem has strong duality property. Therefore, we have

$$\min_{\boldsymbol{\pi}} f(\boldsymbol{\pi}) = \max_{\lambda} \min_{\boldsymbol{\pi}} L(\boldsymbol{\pi}, \lambda)$$

As $L(\boldsymbol{\pi}, \lambda)$ is convex with respect to $\boldsymbol{\pi}$, to find $\min_{\boldsymbol{\pi}} L(\boldsymbol{\pi}, \lambda)$, it suffices to take derivatives with respect to $\pi_k$. This yields

$$\frac{\partial L}{\partial \pi_k} = -\frac{n_k}{\pi_k} + \lambda = 0, k = 1, 2, \ldots, K.$$

or

$$\pi_k = \frac{n_k}{\lambda}, k = 1, 2, \ldots, K$$

Substituting these into the constraint $\sum_{k=1}^{K} \pi_k = 1$ we get $\sum_{k=1}^{K} n_k = \lambda$, yielding $\lambda = N$. From this and (1.24) we get finally

$$\hat{\pi}_k = \frac{n_k}{N}, k = 1, 2, \ldots, K$$

Remark: $\hat{\pi}_k$ is the fraction of the $N$ observations for which $x_k = 1$.

**Exercise:**
MLE for the univariate Gaussian model

**Solution:**
Consider $n$ observations $x_1, x_2, \ldots, x_n$ of a random variable $X$ following a Gaussian distribution $\mathcal{N}\left(\mu, \sigma^2\right)$. From 9 and 7 we have:

$$\begin{aligned}
\ell\left(\mu, \sigma^2\right) &= \sum_{i=1}^{n} \log p\left(x_i; \mu, \sigma^2\right) \\
&= \sum_{i=1}^{n} \log \left[ \frac{1}{\left(2\pi\sigma^2\right)^{1/2}} \exp\left( -\frac{\left(x_i - \mu\right)^2}{2\sigma^2} \right) \right] \\
&= -\frac{n}{2} \log(2\pi) - \frac{n}{2} \log\left(\sigma^2\right) - \frac{1}{2} \sum_{i=1}^{n} \frac{\left(x_i - \mu\right)^2}{\sigma^2}
\end{aligned}$$

We need to maximize this quantity with respect to $\mu$ and $\sigma^2$. By taking derivative with respect to $\mu$ and then $\sigma^2$, it is easy to obtain that the pair $\left(\hat{\mu}, \hat{\sigma}^2\right)$, defined by

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i$$
$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} (x_i - \hat{\mu})^2$$

is the only stationary point of the likelihood. One can actually check (for example computing the Hessian w.r.t. $(\mu, \sigma^2)$ that this actually a maximum.

**Exercise:**
MLE for the multivariate Gaussian model

**Solution:**
Let $X \in \mathbb{R}^d$ be a Gaussian random vector, with mean vector $\mu \in \mathbb{R}^d$ and a covariance matrix $\Sigma \in \mathbb{R}^{d\times d}$ (positive definite). We have from 8

$$p(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}}} \frac{1}{\sqrt{\det \Sigma}} \exp\left(\frac{-(x-\mu)^\top \Sigma^{-1}(x-\mu)}{2}\right)$$

Let $x_1, \ldots, x_n$ be a i.i.d. sample.
As shown in 9, the log-likelihood is given by:

$$\begin{aligned}
\ell(\mu, \Sigma) &= \log p\left(x_1, \ldots, x_n; \mu, \Sigma\right) \\
&= \log \prod_{i=1}^{n} p\left(x_i \mid \mu, \Sigma\right) \\
&= -\left(\frac{nd}{2}\log(2\pi) + \frac{n}{2}\log(\det \Sigma) + \frac{1}{2}\sum_{i=1}^{n} \left(x_i - \mu\right)^\top \Sigma^{-1} \left(x_i - \mu\right)\right)
\end{aligned}$$

In this case, one should be careful that these log-likelihoods are not concave with respect to the pair of parameters $(\mu, \Sigma)$. They are concave w.r.t. $\mu$ when $\Sigma$ is fixed but they are not even concave with respect to $\Sigma$ when $\mu$ is fixed.

Remember that the function we want to differentiate is :

$$\ell(\mu, \Sigma) = -\left(\frac{nd}{2}\log(2\pi) + \frac{n}{2}\log(\det \Sigma) + \frac{1}{2}\sum_{i=1}^{n} \left(x_i - \mu\right)^\top \Sigma^{-1} \left(x_i - \mu\right)\right)$$

Let us first differentiate $\ell(\mu, \Sigma)$ w.r.t. $\mu$.
We need to differentiate :

$$\Psi : \ \mu \mapsto (x_i - \mu)^\top \Sigma^{-1} (x_i - \mu)$$

Which is equal to $g \circ f$ where :

$$\begin{aligned} g : \mathbb{R}^d &\to \quad \mathbb{R} \\ y &\mapsto y^\top \Sigma^{-1} y \end{aligned}$$

and

$$\begin{aligned} f : \mathbb{R}^d &\to \mathbb{R}^d \\ \mu &\mapsto \mu - x_i \end{aligned}$$

Reminder : Composition of differentials

$$d\Psi_a(h) = d(g \circ f)_a(h) = (dg)_{f(a)} \circ df_a(h) = (dg)_{f(a)} (df_a(h))$$

We have:

$$\forall a, h \in \mathbb{R}^d \quad f(a+h) = f(a) + h$$

Thus;

$$\forall a, h \in \mathbb{R}^d \quad df_a(h) = h$$

Moreover, from 1.3:

$$\forall y \in \mathbb{R}^d \quad \nabla g(y) = \left(\Sigma^{-1} + \left(\Sigma^{-1}\right)^\top\right) y = 2\Sigma^{-1} y$$

As a result,

$$\begin{aligned} d(g \circ f)_a(h) &= (dg)_{f(a)} (h) \\ &= \langle \nabla g (f(a)), h \rangle \\ &= \left\langle 2\Sigma^{-1} (a - x_i), h \right\rangle \\ &= \langle \nabla \psi(a), h \rangle \end{aligned}$$

So,

$$\nabla \psi(\mu) = 2\Sigma^{-1} (\mu - x_i)$$

And therefore,

$$\begin{aligned}\nabla_\mu \ell\left(\mu, \Sigma^{-1}\right) &= -\frac{1}{2}\sum_{i=1}^{n} 2\Sigma^{-1}\left(\mu - x_i\right) \\ &= \Sigma^{-1}\left(\sum_{i=1}^{n} x_i - n\mu\right) \\ &= \Sigma^{-1} n(\bar{x} - \mu)\end{aligned}$$

where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$
The gradient is equal to 0 iff :

$$\widehat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Let us now differentiate $\ell$ w.r.t. $\Sigma^{-1}$. Let $A = \Sigma^{-1}$. We have :

$$\ell(\mu, \Sigma) = -\left(\frac{nd}{2}\log(2\pi) - \frac{n}{2}\log(\det A) + \frac{1}{2}\sum_{i=1}^{n}\left(x_i - \mu\right)^\top A\left(x_i - \mu\right)\right)$$

The last term is a real number, so it equal to its trace.
Thus :

$$\ell(\mu, \Sigma) = -\left(\frac{nd}{2}\log(2\pi) - \frac{n}{2}\log(\det A) + \frac{n}{2}\operatorname{tr}(A\widetilde{\Sigma})\right) \tag{10}$$

where

$$\widetilde{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - \mu\right)\left(x_i - \mu\right)^\top$$

is the empirical covariance matrix.
Let $\Phi : A \mapsto \frac{n}{2}\operatorname{tr}(A\widetilde{\Sigma})$.
We have :

$$\begin{aligned}\forall A, H \in \mathbb{R}^d \quad \Phi(A + H) - \Phi(A) &= \frac{n}{2}\operatorname{tr}(H\widetilde{\Sigma}) \\ &= \operatorname{tr}\left(\left(\frac{n}{2}\widetilde{\Sigma}\right)^\top H\right) \\ &= \left\langle \frac{n}{2}\widetilde{\Sigma}, H\right\rangle\end{aligned}$$

The gradient of the last term in eq 10 is then :

$$\nabla\Phi(A) = \frac{n}{2}\widetilde{\Sigma}$$

Let $\Xi : A \mapsto \log(\det(A))$.
From exercise 1.3 we have the gradient of the second term in eq 10:

$$\nabla\Xi(A) = A^{-1}$$

And the gradient of $\ell$ in 10 w.r.t. A is :

$$\nabla_A(\ell) = \frac{n}{2}A^{-1} - \frac{n}{2}\widetilde{\Sigma}$$

It is equal to zero iff :

$$\widehat{\Sigma} = \widetilde{\Sigma}$$

when $\widetilde{\Sigma}$ is invertible.
Finally we have shown that the pair

$$\hat{\mu} = \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad \text{and } \hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^\top$$

is the only stationary point of the likelihood. One can actually check (for example computing the Hessian w.r.t. $(\mu, \Sigma)$ that this actually a maximum.

# 3 Linear Regression

## 3.1 Introduction

When dealing with two random variables $X$ and $Y$, one can use a generative model, i.e. which models the joint distribution $p(X, Y)$, or one can use instead a conditional model (often considered equivalent to the slightly different concept of discriminative model), which models the conditional probability of the output, given the input $p(Y \mid X)$. The two following models, linear regression or a logistic regression, are conditional models.

## 3.2 The Linear Regression model

**Exercise:**
Let's assume that $Y \in \mathbb{R}$ depends linearly on $X \in \mathbb{R}^p$. Let $w \in \mathbb{R}^p$ be a weighting vector and $\sigma^2 > 0$. We make the following assumption:

$$Y \mid X \sim \mathcal{N}\left(\mathbf{w}^\top X, \sigma^2\right)$$

which can be rewritten as

$$Y = \mathbf{w}^\top X + \epsilon$$

with $\epsilon \sim \mathcal{N}\left(0, \sigma^2\right)$. Note that if there is an offset $w_0 \in \mathbb{R}^p$, that is, if $Y = \mathbf{w}^\top X + w_0 + \epsilon$, one can always redefine a weighting vector $\tilde{\mathbf{w}} \in \mathbb{R}^{p+1}$ such that

$$Y = \tilde{\mathbf{w}}^\top \begin{pmatrix} x \\ 1 \end{pmatrix} + \epsilon$$

Let $\mathcal{D} = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ be a training set of i.i.d. random variables. Each $y_i$ is a label (a decision) on observation $\mathbf{x}_i$.

We consider the conditional distribution of all outputs given all inputs, which is a product of terms because of the independence of the pairs forming the training set:

$$p\left(y_1, \cdots, y_n \mid x_1, \cdots, x_n; \mathbf{w}, \sigma^2\right) = \prod_{i=1}^{n} p\left(y_i \mid \mathbf{x}_i; \mathbf{w}, \sigma^2\right)$$

The associated log-likelihood has the following expression:

$$-l\left(\mathbf{w}, \sigma^2\right) = -\sum_{i=1}^{n} \log p\left(y_i \mid \mathbf{x}_i\right) = \frac{n}{2} \log\left(2\pi\sigma^2\right) + \frac{1}{2} \sum_{i=1}^{n} \frac{\left(y_i - \mathbf{w}^\top \mathbf{x}_i\right)^2}{\sigma^2}$$

The minimization problem with respect to $w$ can now be reformulated as:

$$\text{find} \quad \hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \mathbf{w}^\top \mathbf{x}_i\right)^2$$

Define the so-called design matrix $\mathbf{X}$ as

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_n^\top \end{pmatrix} \in \mathbb{R}^{n \times p}$$

and denote by $\mathbf{y}$ the vector of coordinates $(y_1, \cdots, y_n)$.

The minimization problem over $w$ can be rewritten in a more compact way as:

$$\text{find} \quad \hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2.$$

**Solution:**

Let $f : w \mapsto \frac{1}{2n} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2$

We have:

$$\begin{aligned} f(\mathbf{w}) &= \frac{1}{2n}\|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 \\ &= \frac{1}{2n}\left(\mathbf{y}^\top\mathbf{y} - 2\mathbf{w}^\top\mathbf{X}^\top\mathbf{y} + \mathbf{w}^\top\mathbf{X}^\top\mathbf{X}\mathbf{w}\right) \end{aligned}$$

$f$ is strictly convex if and only if its Hessian matrix is invertible. This is never the case when $n < p$ (in this case, we deal with underdetermined problems). Most of the time, the Hessian matrix is invertible when $n \geq p$. When this is not the case, we often use the Tikhonov regularization, which adds a penalization of the $\ell_2$-norm of $w$ by minimizing $f(\mathbf{w}) + \lambda\|\mathbf{w}\|^2$ with some hyperparameter $\lambda > 0$.

The gradient of $f$ is

$$\nabla f(\mathbf{w}) = \frac{1}{n}\mathbf{X}^\top(\mathbf{X}\mathbf{w} - \mathbf{y}) = 0 \Longleftrightarrow \mathbf{X}^\top\mathbf{X}\mathbf{w} = \mathbf{X}^\top\mathbf{y}$$

The equation $\mathbf{X}^\top\mathbf{X}\mathbf{w} = \mathbf{X}^\top\mathbf{y}$ is known as the normal equation.

- If $\mathbf{X}^\top\mathbf{X}$ is invertible.

  Then the optimal weighting vector is

  $$\hat{\mathbf{w}} = \left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{y} = \mathbf{X}^\dagger\mathbf{y}$$

  where $\mathbf{X}^\dagger = \left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top$ is the Moore-Penrose pseudo-inverse of $X$.

- If $\mathbf{X}^\top\mathbf{X}$ is not invertible:

  The solution is not unique anymore, and for any $\mathbf{h} \in \ker(\mathbf{X}), \hat{\mathbf{w}} = \left(\mathbf{X}^\top\mathbf{X}\right)^\dagger\mathbf{X}^\top\mathbf{y} + \mathbf{h}$ is an admissible solution.

  In that case however it would be necessary to use regularization.

Now, let's differentiate $l\left(\mathbf{w}, \sigma^2\right)$ with respect to $\sigma^2$ : we have

$$\nabla_{\sigma^2} l\left(\mathbf{w}, \sigma^2\right) = \frac{n}{2\sigma^2} - \frac{n}{2\sigma^4}\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \mathbf{w}^\top\mathbf{x}_i\right)^2$$

Setting $\nabla_{\sigma^2} l\left(\mathbf{w}, \sigma^2\right)$ to zero gives

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \mathbf{w}^\top\mathbf{x}_i\right)^2$$

## 3.3 Application: Trend Scanning for Labeling Financial Time Series:

### 3.3.1 Visualizing Linear Trends: Understanding Horizon Effects

This section will focus on the exploration of linear trends across different time horizons within financial time series data. We'll visually dissect how varying the observation span impacts the trend analysis, providing insights into the dynamic nature of financial markets and the critical role of temporal context in trend identification.

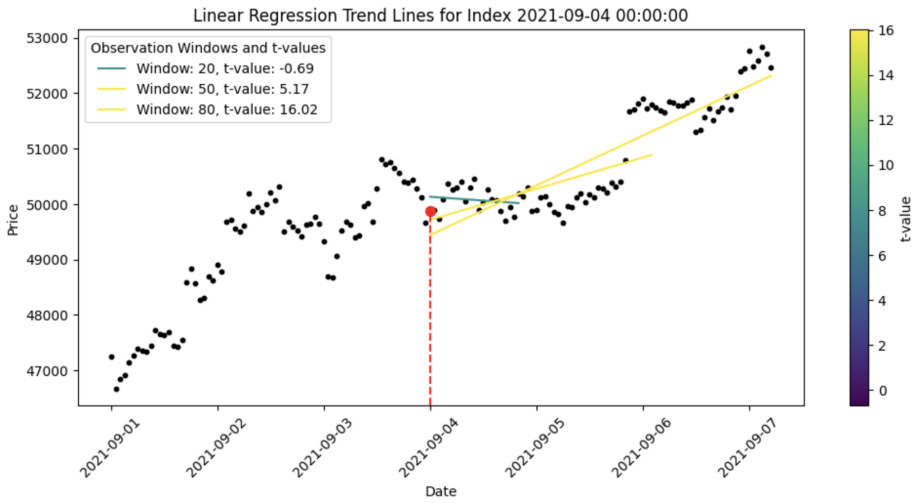


Figure 1: Visualization of the trend lines for the different horizons

### 3.3.2 Trend Scanning on crypto data

In this segment, we delve into the practical application of labeling trends over an entire dataset segment. This hands-on approach aims to equip you with the methodologies for systematically categorizing financial time series data based on observed trends, enhancing the predictive modeling and decision-making process.


Figure 2: Visualization of the trend lines for the different horizons

# 4 Logistic Regression

## 4.1 The Logistic Regression model

**Exercise:**

Let $X \in \mathbb{R}^p, Y \in \{0,1\}$. We assume that $Y$ follows a Bernoulli distribution with parameter $\theta$. The problem is to find $\theta$. Let's define the sigmoid function $\sigma$ defined on the real axis and taking values in $[0,1]$, such that

$$\forall z \in \mathbb{R}, \sigma(z) = \frac{1}{1+e^{-z}}$$

One can easily prove that

$$\begin{aligned} \forall z \in \mathbb{R}, \quad \sigma(-z) &= 1 - \sigma(z), \\ \forall z \in \mathbb{R}, \quad \sigma'(z) &= \sigma(z)(1-\sigma(z)) = \sigma(z)\sigma(-z). \end{aligned}$$

We now assume that, for a given observation $X = \mathbf{x}$, the output $Y \mid X = \mathbf{x}$ follows a Bernoulli law with parameter $\theta = \sigma\left(\mathbf{w}^\top \mathbf{x}\right)$, where $w$ is again a weighting vector.

Then, the conditional distribution is given by

$$p(Y = y \mid X = \mathbf{x}) = \theta^y (1-\theta)^{1-y} = \sigma\left(\mathbf{w}^\top \mathbf{x}\right)^y \sigma\left(-\mathbf{w}^\top \mathbf{x}\right)^{1-y}$$

Given a training set $\mathcal{D} = \{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_n, y_n)\}$ of iid random variables, find the optimal $\mathbf{w}$.

**Solution:**

we can compute the log-likelihood

$$l(\mathbf{w}) = \sum_{i=1}^{n} y_i \log \sigma\left(\mathbf{w}^\top \mathbf{x}_i\right) + (1 - y_i) \log \sigma\left(-\mathbf{w}^\top \mathbf{x}_i\right)$$

In order to minimize the log-likelihood, since $z \mapsto \log\left(1 + e^{-z}\right)$ is a convex function and $\mathbf{w} \mapsto \mathbf{w}^\top \mathbf{x}_i$ is linear, we calculate its gradient.

We write $\eta_i = \sigma\left(\theta^\top \mathbf{x}_i\right)$ :

$$\begin{aligned}\nabla_{\mathbf{w}} l(\mathbf{w}) &= \sum_{i=1}^{n} y_i \mathbf{x}_i \frac{\sigma\left(\mathbf{w}^\top \mathbf{x}_i\right) \sigma\left(-\mathbf{w}^\top \mathbf{x}_i\right)}{\sigma\left(\mathbf{w}^\top \mathbf{x}_i\right)} - (1 - y_i)\, \mathbf{x}_i \frac{\sigma\left(\mathbf{w}^\top \mathbf{x}_i\right) \sigma\left(-\mathbf{w}^\top \mathbf{x}_i\right)}{\sigma\left(-\mathbf{w}^\top \mathbf{x}_i\right)} \\ &= \sum_{i=1}^{n} \mathbf{x}_i \left(y_i - \eta_i\right)\end{aligned}$$

Thus, $\nabla_{\mathbf{w}} l(\mathbf{w}) = 0 \Longleftrightarrow \sum_{i=1}^{n} \mathbf{x}_i \left(y_i - \sigma\left(\theta^\top \mathbf{x}_i\right)\right) = 0$. This equation is nonlinear and we need an iterative optimization method to solve it.

For this purpose, we derive the Hessian matrix of $l$ :

$$\begin{aligned}Hl(\mathbf{w}) &= \sum_{i=1}^{n} \mathbf{x}_i \left(0 - \sigma'\left(\mathbf{w}^\top \mathbf{x}_i\right) \sigma'\left(-\mathbf{w}^\top \mathbf{x}_i\right) \mathbf{x}_i^\top\right) \\ &= \sum_{i=1}^{n} \left(-\eta_i \left(1 - \eta_i\right)\right) \mathbf{x}_i \mathbf{x}_i^\top = -\mathbf{X}^\top \operatorname{Diag}\left(\eta_i \left(1 - \eta_i\right)\right) \mathbf{X}\end{aligned}$$

where $\mathbf{X}$ is the design matrix defined previously.

In the following we discuss first- and second-order optimization methods and apply them to logistic regression.

- **First-order methods**

  Let $f : \mathbb{R}^p \to \mathbb{R}$ be the convex $C^1$ function that we want to minimize. A descent direction at point $\mathbf{x}$ is a vector $d$ such that $\langle \mathbf{d}, \nabla f(\mathbf{x}) \rangle < 0$. The minimization of $f$ can be done by applying a descent algorithm, which iteratively takes a step in a descent direction, leading to an iterative scheme of the form

  $$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \varepsilon^{(k)} \mathbf{d}^{(k)}$$

  where $\varepsilon^{(k)}$ is the stepsize. The direction $\mathbf{d}^{(k)}$ is often chosen as the opposite of the gradient of $f$ at point $\mathbf{x}^{(k)} : \mathbf{d}^{(k)} = -\nabla f\left(\mathbf{x}^{(k)}\right)$.

  There are several choices for $\varepsilon^{(k)}$ :

1. Constant step: $\varepsilon^{(k)} = \varepsilon$. But the scheme does not necessarily converge.
2. Decreasing step size: $\varepsilon^{(k)} \propto \frac{1}{k}$ with $\sum_k \varepsilon^{(k)} = \infty$ and $\sum_k \left(\varepsilon^{(k)}\right)^2 < \infty$. The scheme is guaranteed to converge.
3. One can determine $\varepsilon^{(k)}$ by doing a Line Search which tries to find $\min_\varepsilon f\left(\mathbf{x}^{(k)} + \varepsilon \mathbf{d}^{(k)}\right)$ :

   – either exactly but this is costly and rather useless in many situations;

   – or approximately (see the Armijo linesearch). This is a better method.

- **Second-order methods**

  This time, let $f : \mathbb{R}^p \to \mathbb{R}$ be the $C^2$ function that we want to minimize. We write the second-order Taylor-expansion of $f$ :

  $f(\mathbf{x}) = f(\mathbf{x}^t) + (\mathbf{x} - \mathbf{x}^t)^\top \nabla f(\mathbf{x}^t) + \frac{1}{2}(\mathbf{x} - \mathbf{x}^t)^\top Hf(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t) + o\left(\|\mathbf{x} - \mathbf{x}^t\|^2\right) \stackrel{\text{def}}{=} g_t(\mathbf{x}) + \left(\|\mathbf{x} - \mathbf{x}^t\|^2\right)$

  A local optimum $\mathbf{x}^*$ is then reached when

  $$\begin{cases} \nabla f(\mathbf{x}^*) = 0 \\ H(f(\mathbf{x}^*) \succeq 0 \end{cases}$$

  In order to solve such a problem, we are going to use Newton's method. If $f$ is a convex function, then $\nabla g_t(\mathbf{x}) = \nabla f(\mathbf{x}^t) + Hf(\mathbf{x}^t)(\mathbf{x} - \mathbf{x}^t)$ and we only need to find $\mathbf{x}^*$ so that $\nabla g_t(\mathbf{x}) = 0$, ie. we set $\mathbf{x}^{t+1} = \mathbf{x}^t - \left[Hf(\mathbf{x}^t]^{-1} \nabla f(\mathbf{x}^t)\right.$. If the Hessian matrix is not invertible, we can regularize the problem and minimize $g_t(\mathbf{x}) + \lambda \|\mathbf{x} - \mathbf{x}^t\|^2$ instead.

  In general the previous update, called the Pure Newton step does not lead to a convergent algorithm even if the function is convex!

  In general it is necessary to use the so-called Damped Newton method, to obtain a convergent algorithm which consists in doing the following iterations:

  $$\mathbf{x}^{t+1} = \mathbf{x}^t - \varepsilon^t \left(Hf(\mathbf{x}^t)\right)^{-1} \nabla f(\mathbf{x}^t),$$

  where $\varepsilon^t$ is set with the Armijo Line Search

  This method may be computationally costly in high dimension because of the inverse of the hessian matrix that needs to be computed at each iteration. For some functions, however, the pure Newton's method does converge. This is the case for logistic regression.

In the context of non-convex optimization, the situation is more complicated because the Hessian can have negative eigenvalues. In that case, so-called trust region methods are typically used.

- **Application to logistic regression**

  We will write the form that Newton's algorithm takes for logistic regression. We had :

$$
\begin{aligned}
l(\mathbf{w}) &= \sum_{i=1}^{n} y_i \log \sigma\left(\mathbf{w}^\top \mathbf{x}_i\right) + (1-y_i) \log \sigma\left(-\mathbf{w}^\top \mathbf{x}_i\right) \\
\nabla_{\mathbf{w}} l(\mathbf{w}) &= \sum_{i=1}^{n} \mathbf{x}_i \left(y_i - \eta_i\right) = \mathbf{X}^\top (\mathbf{y} - \boldsymbol{\eta}) \\
Hl(\mathbf{w}) &= -\mathbf{X}^\top \operatorname{Diag}\left(\eta_i \left(1-\eta_i\right)\right) \mathbf{X}
\end{aligned}
$$

  The second-order Taylor expansion of the loss function leads to

$$
l(\mathbf{w}) = l\left(\mathbf{w}^t\right) + \left(\mathbf{w} - \mathbf{w}^t\right)^\top \nabla l\left(\mathbf{w}^t\right) + \frac{1}{2}\left(\mathbf{w} - \mathbf{w}^t\right)^\top Hl\left(\mathbf{w}^t\right)\left(\mathbf{w} - \mathbf{w}^t\right)
$$

  Let us set $\mathbf{h} = \mathbf{w} - \mathbf{w}^t$.

  The minimization problem becomes:

$$
\begin{aligned}
&\min_{\mathbf{h}} \left\{ \mathbf{h}^\top \mathbf{X}^\top (\mathbf{y} - \boldsymbol{\eta}) - \frac{1}{2} \mathbf{h}^\top \mathbf{X}^\top \operatorname{Diag}(\eta(1-\eta)) \mathbf{X}\mathbf{h} \right\} \\
&\iff \min_{\mathbf{h}} \mathbf{h}^\top \nabla_{\mathbf{w}} l(\mathbf{w}) + \frac{1}{2} \mathbf{h}^\top Hl(\mathbf{w}) \mathbf{h}.
\end{aligned}
$$

  This leads, according to the previous part, to set:

$$
\mathbf{w}^{t+1} = \mathbf{w}^t + Hl\left(\mathbf{w}^t\right)^{-1} \nabla_{\mathbf{w}} l(\mathbf{w})
$$

  The minimization problem above can be seen as some weighted linear regression over $h$ of some function of the form $\sum_i \frac{\left(\tilde{y}_i - \tilde{\mathbf{x}}_i{}^\top \mathbf{h}\right)^2}{\sigma_i^2}$, where $\tilde{y}_i = y_i - \eta_i$ and $\sigma_i^2 = \left[\eta_i \left(1-\eta_i\right)\right]^{-1}$.

  Thus, this method is often refered as the iterative reweighted least squares algorithm (IRLS).

## 4.2 Application: Feature Selection on dummy data

### 4.2.1 The problem

Consider a binary random classification problem composed of forty features, where five are informative, thirty are redundant, and five are noise.

- **Informative features** (marked with the $I$ prefix) are those used to generate labels.
- **Redundant features** (marked with the $R$ prefix) are those that are formed by adding Gaussian noise to a randomly chosen informative feature.
- **Noise features** (marked with the $N$ prefix) are those that are not used to generate labels.

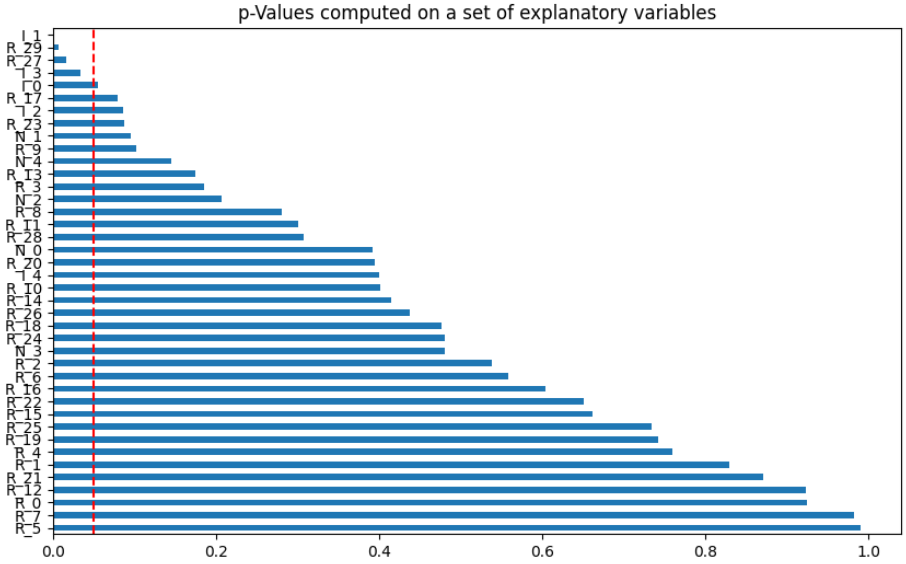


Figure 3: p-values from a logit regression

Figure 3 presents the p-values from a logit regression on selected features, where horizontal bars indicate the p-values, and a vertical dashed line denotes the 5% significance threshold. Remarkably, only $I_1, R_{29}, R_{27}, I_3$ out of thirty-five non-noise features are statistically significant at this level.

Interestingly, noise features are often misinterpreted as important, with fourteen non-noise features ranked among the least important, challenging the reliability of these p-values due to several caveats:

1. **Assumption Sensitivity:** P-values depend heavily on underlying assumptions. Inaccuracies here can lead to false positives (low p-value with

a true coefficient of zero) or false negatives (high p-value with a nonzero true coefficient).

2. **Multicollinearity Issues:** In the presence of multicollinearity among explanatory variables, p-values become unreliable. Traditional regression struggles to differentiate among correlated variables, distorting the interpretation of their significance.

3. **Irrelevant Probability Estimation:** P-values estimate the probability of observing data as extreme as or more than the estimated coefficient $\hat{\beta}$ under the null hypothesis $H_0$. This contrasts with the often more relevant probability of $H_0$ being true given $\hat{\beta}$, which requires Bayesian approaches and additional assumptions.

4. **In-Sample Bias:** P-values assess significance using the same sample for both coefficient estimation and significance testing, potentially overfitting and misrepresenting out-of-sample explanatory power. This in-sample testing can lead to p-hacking and false discoveries.

In essence, p-values necessitate multiple assumptions (#1), produce estimates sensitive to multicollinearity (#2), calculate a probability of limited practical interest (#3), and may lack out-of-sample generalizability (#4).

The theoretical transparency offered by classical methods in attributing significance is compromised in practice by these issues. It suggests a potential benefit from integrating modern computational techniques to address these challenges.

# References

[1] Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.