

Randal Douc & Sylvain Le Corff

MAT 4506

Machine Learning

Contents

1	Principal component analysis	1
1.1	Principal Component Analysis as a singular value decomposition problem	2
1.2	Principal Component Analysis as an optimal projection	6
1.3	Interpretation of the Principal Component Analysis	7
2	Supervised classification	11
2.1	Bayes classifier	11
2.2	Parametric and semiparametric classifiers	12
2.3	Nonparametric Bayes classifier	23
2.4	Classification based on convexification	26
3	Stochastic gradient descent	31
3.1	Gentle introduction, minimization of a convex Lipschitz function on \mathbb{R}	31
3.2	Gradient descent on \mathbb{R}^d	34
3.3	Gradient descent projected on a bounded convex subset of \mathbb{R}^d	36
3.4	Stochastic gradient descent algorithm	38
3.5	Optimization methods for neural networks	40
4	Multivariate regression	43
4.1	Gaussian vectors	43
4.2	Full rank multivariate regression	44
4.3	Introduction to regularized multivariate regression	47
5	Technical results	51
5.1	Probabilistic inequalities	51
5.2	Matrix calculus	52
6	Exercices	55
6.1	Linear discriminant analysis	55
6.2	Plug-in classifier	57
6.3	Logistic Regression	58
6.4	K-means algorithm	61
6.5	Gaussian vectors	63
6.6	Regression: prediction of a new observation	63
6.7	Regression: linear estimators	63
6.8	Kernels	64
6.9	Penalized kernel regression	64
6.10	Expectation Maximization algorithm	65
	References	68

Chapter

1

Principal component analysis

Contents

1.1	Principal Component Analysis as a singular value decomposition problem	2
1.1.1	Refresher on matrices	2
1.1.2	Singular value decomposition	2
1.1.3	Application to Principal Component Analysis	3
1.2	Principal Component Analysis as an optimal projection	6
1.3	Interpretation of the Principal Component Analysis	7
1.3.1	Principal components	7
1.3.2	Projection of the variables	7
1.3.3	Explained variance and projection quality	9
1.3.4	Application to the USArrests dataset [James et al., 2013]	9

Keywords 1.1 *Principal components; singular value decomposition.*

Principal component analysis is a multivariate technique which aims at analyzing the statistical structure of high dimensional dependent observations by representing data using orthogonal variables called *principal components*. Its origin may be traced back to [Hotelling, 1933] who first introduced the principal components as a way to reduce the dimensionality of the data. The objective is to find a low-dimensional representation that captures the statistical properties of high-dimensional data. Reducing the dimensionality of the data is motivated by several practical reasons such as the following (in addition to better understand the observations which are difficult to plot and interpret).

- Compression, denoising, data completion, anomaly detection.
- Preprocessing before supervised learning (improve performances / regularization to reduce overfitting).
- Simplifying the description of massive datasets.
- Providing tools to analyze both observations and variables.

Principal component analysis relies on the implicit assumption that the data lies in a low-dimensional manifold of the original space. Let $(X_i)_{1 \leq i \leq n}$ be i.i.d. random variables in \mathbb{R}^d and consider the matrix $X \in \mathbb{R}^{n \times d}$ such that the i -th row of X is the observation X_i^T . In this chapter, it is assumed that the columns of X are centered. This means that for all $1 \leq k \leq d$, $\sum_{i=1}^n X_{i,k} = 0$. Let Σ_n be the empirical covariance matrix:

$$\Sigma_n = n^{-1} \sum_{i=1}^n X_i X_i^T.$$

Principal Component Analysis aims at reducing the dimensionality of the observations $(X_i)_{1 \leq i \leq n}$ using a *compression* matrix $W \in \mathbb{R}^{p \times d}$ with $p \leq d$ so that for each $1 \leq i \leq n$, WX_i is a low dimensional representation of X_i . The original observation may then be partially recovered using another matrix $U \in \mathbb{R}^{d \times p}$. Principal Component Analysis computes U and W using the least squares approach:

$$(U_*, W_*) \in \underset{(U, W) \in \mathbb{R}^{d \times p} \times \mathbb{R}^{p \times d}}{\operatorname{argmin}} \sum_{i=1}^n \|X_i - UW X_i\|^2,$$

The source codes in R and/or Python of this chapter may be found at:

<https://sylvainlc.github.io/project/teaching/>

1.1 Principal Component Analysis as a singular value decomposition problem

1.1.1 Refresher on matrices

Lemma 1.1 *Let A be a $n \times d$ matrix with real entries. Then, $\operatorname{range}(A) = \operatorname{range}(AA^T)$.*

PROOF. First note that for all $x \in \mathbb{R}^n$, $AA^T x = 0$ implies $\langle A^T x; A^T x \rangle = 0$ so that $A^T x = 0$. The converse is obvious. Therefore, $\operatorname{Ker}(AA^T) = \operatorname{Ker}(A^T)$. And, using that for any matrix B , $\operatorname{Ker}(B^T) = (\operatorname{range}(B))^\perp$, yields $\operatorname{range}(AA^T)^\perp = \operatorname{range}(A)^\perp$, which concludes the proof. ■

Lemma 1.2 *Let $\{U_k\}_{1 \leq k \leq r}$ be a family of r orthonormal vectors of \mathbb{R}^d . Then, $\sum_{k=1}^r U_k U_k^T$ is the matrix of the orthogonal projection onto*

$$\mathbf{H} = \left\{ \sum_{k=1}^r \alpha_k U_k; \alpha_1, \dots, \alpha_r \in \mathbb{R} \right\}.$$

Remark 1.3 *If A is a $n \times d$ matrix with real entries such that each column of A is in \mathbf{H} , then,*

$$\left(\sum_{k=1}^r U_k U_k^T \right) A = A.$$

PROOF. For all $X \in \mathbb{R}^d$, let $\pi_{\mathbf{H}}(X)$ be the orthogonal projection of X onto \mathbf{H} . Since $\{U_k\}_{1 \leq k \leq r}$ is an orthonormal basis of \mathbf{H} ,

$$\pi_{\mathbf{H}}(X) = \sum_{k=1}^r \langle X; U_k \rangle U_k = \left(\sum_{k=1}^r U_k U_k^T \right) X.$$

This implies in particular that for each $X \in \mathbf{H}$, $X = (\sum_{k=1}^r U_k U_k^T) X$. ■

1.1.2 Singular value decomposition

Proposition 1.4 *For all $\mathbb{R}^{n \times d}$ matrix A with rank r , there exist $\sigma_1 \geq \dots \geq \sigma_r > 0$ such that*

$$A = \sum_{k=1}^r \sigma_k u_k v_k^T,$$

where $\{u_1, \dots, u_r\} \in (\mathbb{R}^n)^r$ and $\{v_1, \dots, v_r\} \in (\mathbb{R}^d)^r$ are two orthonormal families. The vectors $\{\sigma_1, \dots, \sigma_r\}$ are called singular values of A and $\{u_1, \dots, u_r\}$ (resp. $\{v_1, \dots, v_r\}$) are the left-singular (resp. right-singular) vectors of A .

Remark 1.5 If U denotes the $\mathbb{R}^{n \times r}$ matrix with columns given by $\{u_1, \dots, u_r\}$ and V denotes the $\mathbb{R}^{p \times r}$ matrix with columns given by $\{v_1, \dots, v_r\}$, then the singular value decomposition of A may also be written as

$$A = UD_rV^T,$$

where $D_r = \text{diag}(\sigma_1, \dots, \sigma_r)$.

Remark 1.6 The singular value decomposition is closely related to the spectral theorem for symmetric semipositive definite matrices. In the framework of Proposition 1.8, $A^T A$ and AA^T are positive semidefinite such that

$$A^T A = VD_r^2V^T \quad \text{and} \quad AA^T = UD_r^2U^T.$$

PROOF. Since the matrix AA^T is positive semidefinite, its spectral decomposition is given by

$$AA^T = \sum_{k=1}^r \lambda_k u_k u_k^T,$$

where $\lambda_1 \geq \dots \geq \lambda_r > 0$ are the nonzero eigenvalues of AA^T and $\{u_1, \dots, u_r\}$ is an orthonormal family of \mathbb{R}^n . For all $1 \leq k \leq r$, define $v_k = \lambda_k^{-1/2} A^T u_k$ so that

$$\|v_k\|^2 = \lambda_k^{-1} \langle A^T u_k, A^T u_k \rangle = \lambda_k^{-1} u_k^T A A^T u_k = 1,$$

$$A^T A v_k = \lambda_k^{-1/2} A^T A A^T u_k = \lambda_k v_k.$$

On the other hand, for all $1 \leq k \neq j \leq r$, $\langle v_k, v_j \rangle = \lambda_k^{-1/2} \lambda_j^{-1/2} u_k^T A A^T u_j = \lambda_k^{-1/2} \lambda_j^{1/2} u_k^T u_j = 0$. Therefore, $\{v_1, \dots, v_r\}$ is an orthonormal family of eigenvectors of $A^T A$ associated with the eigenvalues $\lambda_1 \geq \dots \geq \lambda_r > 0$. Define, for all $1 \leq k \leq r$, $\sigma_k = \lambda_k^{1/2}$ which yields

$$\sum_{k=1}^r \sigma_k u_k v_k^T = \sum_{k=1}^r u_k u_k^T A = \left(\sum_{k=1}^r u_k u_k^T \right) A.$$

As $\{u_1, \dots, u_r\}$ is an orthonormal family, by Lemma 1.2 $U U^T = \sum_{k=1}^r u_k u_k^T$ is the orthogonal projection onto the range(AA^T). And, by Lemma 1.1, $\text{range}(AA^T) = \text{range}(A)$, which implies

$$\sum_{k=1}^r \sigma_k u_k v_k^T = \left(\sum_{k=1}^r u_k u_k^T \right) A = A.$$

■

1.1.3 Application to Principal Component Analysis

As mentioned in the introduction, Principal Component Analysis aims at solving the following optimization problem:

$$(U_*, W_*) \in \underset{(U, W) \in \mathbb{R}^{d \times p} \times \mathbb{R}^{p \times d}}{\text{argmin}} \sum_{i=1}^n \|X_i - U W X_i\|^2, \quad (1.1)$$

Lemma 1.7 Let $(U_*, W_*) \in \mathbb{R}^{d \times p} \times \mathbb{R}^{p \times d}$ be a solution to (1.1). Then, the columns of U_* are orthonormal and $W_* = U_*^T$.

PROOF. Let $(U, W) \in \mathbb{R}^{d \times p} \times \mathbb{R}^{p \times d}$ and V be a matrix whose columns are given by an orthonormal basis of the vector space E generated by the UW . Therefore, for all $x \in \mathbb{R}^d$,

$$VV^T x = \underset{z \in E}{\text{argmin}} \|x - z\|^2.$$

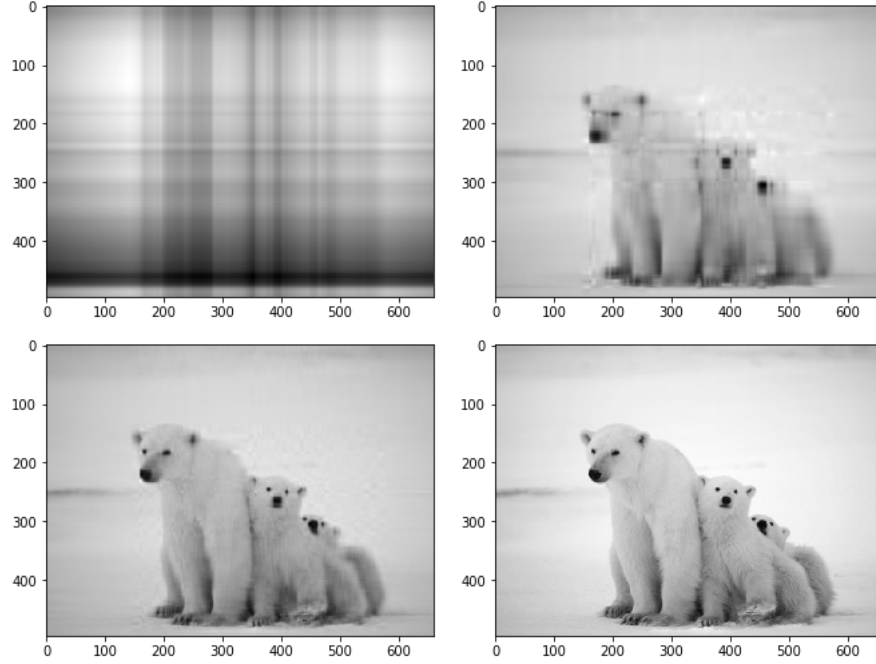


Fig. 1.1 Application of SVD to image reconstruction. The original image is a $\mathbb{R}^{495 \times 660}$ matrix of pixels (grey scale). Its SVD is obtained using the `np.linalg.svd` function in Python and the image is then reconstructed using only the largest (top left), the first ten (top right), the first 25 (bottom left) and the first 100 (bottom right) singular values.

As for all $1 \leq i \leq n$, $UW X_i \in E$,

$$\sum_{i=1}^n \|X_i - UW X_i\|^2 \geq \sum_{i=1}^n \|X_i - VV^T X_i\|^2,$$

which concludes the proof. ■

Let $U \in \mathbb{R}^{d \times p}$ be such that $U^T U = I_p$. Then,

$$\begin{aligned} \sum_{i=1}^n \|X_i - U U^T X_i\|^2 &= \sum_{i=1}^n \|X_i\|^2 + \sum_{i=1}^n \|U U^T X_i\|^2 - 2 \sum_{i=1}^n \langle X_i, U U^T X_i \rangle, \\ &= \sum_{i=1}^n \|X_i\|^2 + \sum_{i=1}^n X_i^T U U^T X_i - 2 \sum_{i=1}^n X_i^T U U^T X_i, \\ &= \sum_{i=1}^n \|X_i\|^2 - \sum_{i=1}^n X_i^T U U^T X_i, \\ &= \sum_{i=1}^n \|X_i\|^2 - \text{trace}(U^T X X^T U). \end{aligned}$$

Therefore, by Lemma 1.7, solving (1.1) boils down to computing

$$U_\star \in \underset{U \in \mathbb{R}^{d \times p}, U^T U = I_p}{\operatorname{argmax}} \{ \text{trace}(U^T \Sigma_n U) \}. \quad (1.2)$$

Proposition 1.8 *Let $\{\vartheta_1, \dots, \vartheta_d\}$ be orthonormal eigenvectors associated with the eigenvalues $\lambda_1 \geq \dots \geq \lambda_d$ of Σ_n . Then a solution to (1.1) is given by the matrix U_\star with columns $\{\vartheta_1, \dots, \vartheta_p\}$ and $W_\star = U_\star^T$.*


```

# write a function with input the path of an image "path_image" and an integer "k"
# and return in gray scale the reconstructed picture with the first k singular values
def svd_decomposition(path_image,k):
    img = Image.open(path_image)
    img_mat = np.array(list(img.getdata(band=0)), float)
    img_mat.shape = (img.size[1], img.size[0])
    img_mat = np.matrix(img_mat)

    # Perform Singular Value Decomposition
    U, sigma, V = np.linalg.svd(img_mat)
    print('Size left singular eigenvectors ' + str(np.shape(U)))
    print('Size right singular eigenvectors ' + str(np.shape(V)))
    print('Size eigenvalues matrix ' + str(np.shape(sigma)))

    # Image reconstruction
    reconstimg = np.matrix(U[:, :k]) * np.diag(sigma[:k]) * np.matrix(V[:k, :])

    fig = plt.figure(1)
    plt.plot(sigma[0:12]*100/np.sum(sigma))
    plt.title("Normalized values of the singular values (in %)")
    fig = plt.figure(2)
    plt.title("Image reconstruction with %g singular values"%k)
    plt.imshow(reconstimg, cmap='gray')
    plt.axis('off')

```

```

Size left singular eigenvectors (1415, 1415)
Size right singular eigenvectors (2122, 2122)
Size eigenvalues matrix (1415,)

```

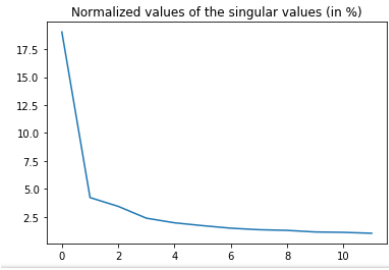


Fig. 1.2 Application of a singular value decomposition in Python

Remark 1.9 Note that $\sum_{i=1}^n X_i X_i^T = X^T X$ so that $\Sigma_n = (X/\sqrt{n})^T (X/\sqrt{n})$ and the eigenvalue decomposition of Σ_n allows to recover the singular values of X/\sqrt{n} and its right-singular vectors.

PROOF. By Lemma 1.7 and (1.2), the proof is equivalent to prove that U_* is a solution to (1.2). Let $\Sigma_n = V D_n V^T$ be the spectral decomposition of Σ_n where $D_n = \text{Diag}(\lambda_1, \dots, \lambda_d)$ and $V \in \mathbb{R}^{d \times d}$ is a matrix with columns $\{\vartheta_1, \dots, \vartheta_d\}$. For all $U \in \mathbb{R}^{d \times p}$ matrix with orthonormal columns define $B = V^T U$ so that, as $V \in \mathbb{R}^{d \times d}$ is an orthogonal matrix,

$$VB = VV^T U = U \quad \text{and} \quad U^T \Sigma_n U = B^T V^T V D_n V^T V B = B^T D_n B.$$

Therefore,

$$\text{Trace}(U^T \Sigma_n U) = \text{Trace}(B^T D_n B) = \sum_{i=1}^d \lambda_i \sum_{j=1}^p b_{i,j}^2. \quad (1.3)$$

On the other hand,

$$B^T B = U^T V V^T U = U^T U = I_p,$$

so that the columns of B are orthonormal and

$$\sum_{i=1}^d \sum_{j=1}^p b_{i,j}^2 = p.$$

By (1.3),

$$\text{Trace}(U^T \Sigma_n U) = \sum_{i=1}^d \alpha_i \lambda_i,$$

with, for all $1 \leq i \leq d$, $\alpha_i \in [0, 1]$ and $\sum_{i=1}^d \alpha_i = p$. As $\lambda_1 \geq \lambda_2 \geq \dots, \lambda_d$,

$$\text{Trace}(U^T \Sigma_n U) \leq \sum_{i=1}^p \lambda_i.$$

As the columns of U_* are $\{\vartheta_1, \dots, \vartheta_p\}$, for all $1 \leq i \leq d$ and $1 \leq j \leq p$, $b_{i,j} = \langle \vartheta_i; \vartheta_j \rangle = \delta_{i,j}$. Therefore, for all $1 \leq i \leq d$, $\sum_{j=1}^p b_{i,j}^2 = 1$ and by (1.3),

$$\text{Trace}(U_*^T \Sigma_n U_*) = \sum_{i=1}^p \lambda_i,$$

which completes the proof. \blacksquare

1.2 Principal Component Analysis as an optimal projection

For any dimension $1 \leq p \leq d$, let \mathcal{F}_d^p be the set of all vector subspaces of \mathbb{R}^d with dimension p . In this section, it is proved that Principal Component Analysis computes a linear span V_d such as

$$V_p \in \underset{V \in \mathcal{F}_d^p}{\operatorname{argmin}} \sum_{i=1}^n \|X_i - \pi_V(X_i)\|^2, \quad (1.4)$$

where π_V is the orthogonal projection onto the linear span V . Assume first that $p = 1$ and write $V_1 = \operatorname{span}\{v_1\}$ for $v_1 \in \mathbb{R}^d$ such that $\|v_1\| = 1$. Then,

$$\begin{aligned} \sum_{i=1}^n \|X_i - \pi_{V_1}(X_i)\|^2 &= \sum_{i=1}^n \|X_i - \langle X_i; v_1 \rangle v_1\|^2, \\ &= \sum_{i=1}^n (\|X_i\|^2 - 2\langle X_i; \langle X_i; v_1 \rangle v_1 \rangle + \|\langle X_i; v_1 \rangle v_1\|^2), \\ &= \sum_{i=1}^n (\|X_i\|^2 - \langle X_i; v_1 \rangle^2). \end{aligned}$$

Consequently, V_1 is a solution to (1.4) if and only if v_1 is solution to:

$$v_1 \in \underset{v \in \mathbb{R}^d; \|v\|=1}{\operatorname{argmax}} \sum_{i=1}^n \langle X_i, v \rangle^2.$$

For all $2 \leq p \leq d$, following the same steps, it can be proved that a solution to (1.4) is given by $V_p = \operatorname{span}\{v_1, \dots, v_p\}$ where

$$v_1 \in \underset{v \in \mathbb{R}^d; \|v\|=1}{\operatorname{argmax}} \sum_{i=1}^n \langle X_i, v \rangle^2 \text{ and for all } 2 \leq k \leq p, \quad v_k \in \underset{\substack{v \in \mathbb{R}^d; \|v\|=1; \\ v \perp v_1, \dots, v \perp v_{k-1}}}{\operatorname{argmax}} \sum_{i=1}^n \langle X_i, v \rangle^2. \quad (1.5)$$

It remains to prove that the vectors $\{v_1, \dots, v_p\}$ defined by (1.5) can be chosen as the orthonormal eigenvectors associated with the p largest eigenvalues of the empirical covariance matrix Σ_n . Note that for all $v \in \mathbb{R}^d$ such that $\|v\| = 1$,

$$\frac{1}{n} \sum_{i=1}^n \langle X_i, v \rangle^2 = \frac{1}{n} \sum_{i=1}^n (v^T X_i)(X_i^T v) = v^T \Sigma_n v.$$

As $(\vartheta_i)_{1 \leq i \leq d}$ are the orthonormal eigenvectors associated with the eigenvalues $\lambda_1 \geq \dots \geq \lambda_d \geq 0$ of Σ_n . Then,

$$\frac{1}{n} \sum_{i=1}^n \langle X_i, v \rangle^2 = v^T \left(\sum_{i=1}^d \lambda_i \vartheta_i \vartheta_i^T \right) v = \sum_{i=1}^d \lambda_i \langle v, \vartheta_i \rangle^2 \leq \lambda_1 \sum_{i=1}^d \langle v, \vartheta_i \rangle^2$$

and, as $(\vartheta_i)_{1 \leq i \leq d}$ is an orthonormal basis of \mathbb{R}^d , $\sum_{i=1}^d \langle v, \vartheta_i \rangle^2 = \|v\|^2 = 1$. Therefore,

$$\frac{1}{n} \sum_{i=1}^n \langle X_i, v \rangle^2 \leq \lambda_1 .$$

On the other hand, for all $2 \leq i \leq d$, $\langle \vartheta_1, \vartheta_i \rangle = 0$ and $\langle \vartheta_1, \vartheta_1 \rangle = 1$ so that $\sum_{i=1}^d \lambda_i \langle \vartheta_1, \vartheta_i \rangle^2 = \lambda_1$ which proves that ϑ_1 is solution to (1.5).

Assume now that $v \in \mathbb{R}^d$ is such that $\|v\| = 1$ and for all $1 \leq j \leq k-1$, $\langle v, \vartheta_j \rangle = 0$ and write

$$\frac{1}{n} \sum_{i=1}^n \langle X_i, v \rangle^2 = \sum_{i=1}^d \lambda_i \langle v, \vartheta_i \rangle^2 \leq \lambda_k \sum_{i=k}^d \langle v, \vartheta_i \rangle^2 \leq \lambda_k ,$$

since, as $(\vartheta_i)_{1 \leq i \leq d}$ is an orthonormal basis of \mathbb{R}^d , $\sum_{i=1}^d \langle v, \vartheta_i \rangle^2 = \sum_{i=k}^d \langle v, \vartheta_i \rangle^2 = \|v\|^2 = 1$. On the other hand, for all $1 \leq i \leq d$, $i \neq k$, $\langle \vartheta_k, \vartheta_i \rangle = 0$ and $\langle \vartheta_k, \vartheta_k \rangle = 1$ so that $\sum_{i=1}^d \lambda_i \langle \vartheta_k, \vartheta_i \rangle^2 = \lambda_k$ which proves that ϑ_k is solution to (1.5).

Therefore, $V_p = \text{span}\{\vartheta_1, \dots, \vartheta_p\}$ is a solution to (1.5) and, as $(\vartheta_i)_{1 \leq i \leq p}$ is an orthonormal family, the projection matrix onto V_p is given by $U_* U_*^T$ where U_* is a $\mathbb{R}^{d \times p}$ matrix with columns $\{\vartheta_1, \dots, \vartheta_p\}$.

1.3 Interpretation of the Principal Component Analysis

1.3.1 Principal components

The orthonormal eigenvectors associated with the eigenvalues of Σ_n allow to define the principal components as follows. Then, as $V_d = \text{span}\{\vartheta_1, \dots, \vartheta_d\}$, for all $1 \leq i \leq n$,

$$\pi_{V_d}(X_i) = \sum_{k=1}^d \langle X_i, \vartheta_k \rangle \vartheta_k = \sum_{k=1}^d (X_i^T \vartheta_k) \vartheta_k = \sum_{k=1}^d c_k(i) \vartheta_k ,$$

where for all $1 \leq k \leq d$, the k -th principal component is defined as $c_k = X \vartheta_k$. Therefore the k -th principal component is the vector whose components are the coordinate are the coordinates of each X_i , $1 \leq i \leq n$, relative to the basis $\{\vartheta_1, \dots, \vartheta_d\}$ of V_d .

1.3.2 Projection of the variables

For all $1 \leq i \neq j \leq d$,

$$\langle c_i, c_j \rangle = \vartheta_i^T X^T X \vartheta_j = \vartheta_i^T (n \Sigma_n) \vartheta_j = n \lambda_j \vartheta_i^T \vartheta_j = 0 , \quad (1.6)$$

as $\{\vartheta_1, \dots, \vartheta_d\}$ is an orthonormal family. Let W_d be the vector subspace of \mathbb{R}^n generated by $\{c_1, \dots, c_d\}$. Since $(c_j)_{1 \leq j \leq d}$ form a orthogonal basis of W_d , for all $1 \leq j \leq d$,

$$\pi_{W_p}(X_{.,j}) = \sum_{\ell=1}^p \frac{\langle c_\ell, X_{.,j} \rangle}{\|c_\ell\|^2} c_\ell .$$

By (1.6), for all $1 \leq \ell \leq d$, $\|c_\ell\|^2 = n \lambda_\ell$ and

$$\langle c_\ell, X_{.,j} \rangle = \langle X \vartheta_\ell, X_{.,j} \rangle = X_{.,j}^T X \vartheta_\ell = (X^T X \vartheta_\ell)_j = (n \Sigma_n \vartheta_\ell)_j = n \lambda_\ell \vartheta_\ell(j) .$$

This yields, for all $1 \leq j \leq d$,

$$\pi_{W_p}(X_{.,j}) = \sum_{\ell=1}^p \vartheta_\ell(j) c_\ell .$$

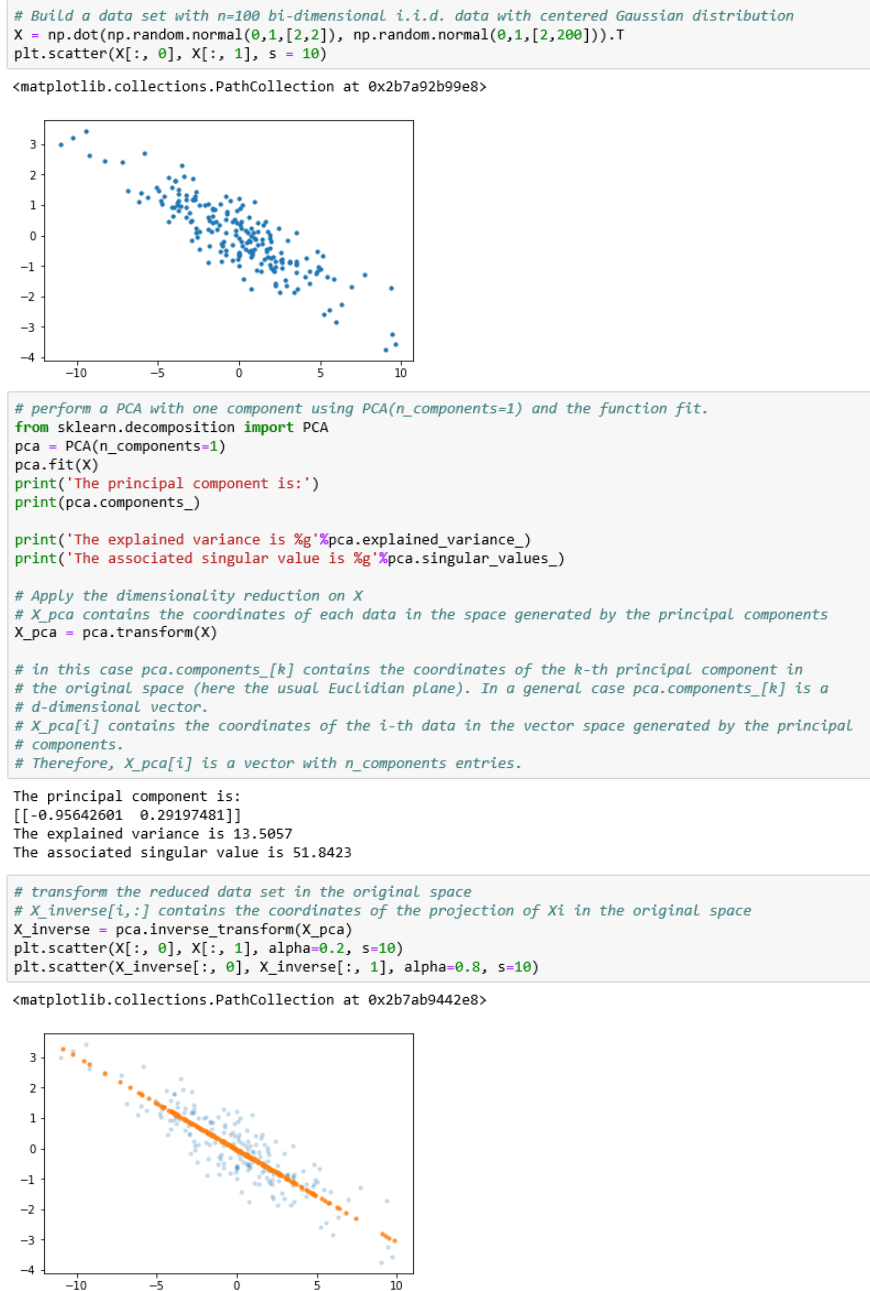


Fig. 1.3 Principal component analysis with one component in Python.

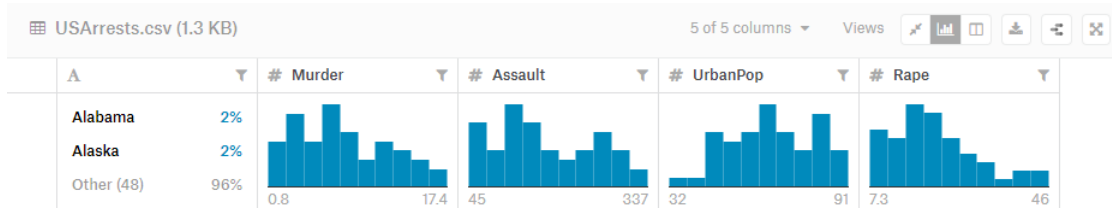
1.3.3 Explained variance and projection quality

The percentage of variance explained by the first p dimensions is:

$$\alpha_p = \frac{n^{-1} \sum_{i=1}^n \|\pi_{V_p}(X_i)\|^2}{n^{-1} \sum_{i=1}^n \|X_i\|^2} = \frac{n^{-1} \sum_{i=1}^n \|\pi_{V_p}(X_i)\|^2}{\text{trace}(\Sigma_n)} = \frac{\sum_{i=1}^p \lambda_i}{\sum_{i=1}^d \lambda_i}.$$

1.3.4 Application to the USArrests dataset [James et al., 2013]

Dataset containing arrests per 100.000 residents for assault, murder and rape for the 50 US states in 1973. Percent of the population living in urban areas are also given.



	PC1	PC2
Murder	0.5358995	-0.4181809
Assault	0.5831836	-0.1879856
UrbanPop	0.2781909	0.8728062
Rape	0.5434321	0.1673186

In this setting $n = 50$ (number states) and $d = 4$ (number of variables - Murder, Assault, Rape and UrbanPop).

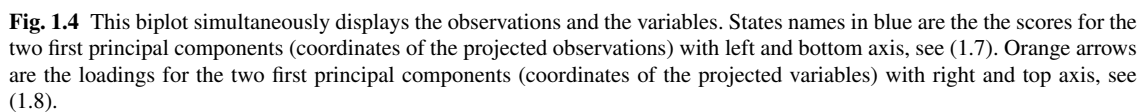
$$\begin{aligned}\vartheta_1 &= (0.53, 0.58, 0.28, 0.54), \\ \vartheta_2 &= (-0.42, -0.19, 0.87, 0.17).\end{aligned}$$

The coordinates of the projected observations (the states) in the plane generated by the two first eigenvectors are:

$$\pi_{V_2}(X_i) = (X_i^T \vartheta_1) \vartheta_1 + (X_i^T \vartheta_2) \vartheta_2. \quad (1.7)$$

The coordinates of the projected variables (Murder, Assault, Rape and UrbanPop) in the plane generated by the two first components are:

$$\pi_{W_2}(X_{.,j}) = \sum_{\ell=1}^2 \vartheta_\ell(j) c_\ell. \quad (1.8)$$



Chapter 2

Supervised classification

Contents

2.1	Bayes classifier	11
2.2	Parametric and semiparametric classifiers	12
2.2.1	Mixture of Gaussian distributions	12
2.2.2	Logistic regression	15
2.2.3	Feed Forward Neural Networks - Multi layer perceptron	16
2.2.4	Support vector machines	19
2.3	Nonparametric Bayes classifier	23
2.4	Classification based on convexification	26

Keywords 2.1 *Bayes classifier, empirical risk, oracle inequality, linear discriminant analysis, logistic regression, support vector machines.*

In a supervised learning framework, the problem is to learn whether an individual from a given state space \mathcal{X} belongs to some class. In this introduction, the focus is set on supervised learning with two classes so that an individual is associated with a label in $\{-1, 1\}$. The state space \mathcal{X} is usually a subset of \mathbb{R}^d an element of \mathcal{X} contains all the features the label prediction has to be based on. The prediction rule relies on a training data set $\{(X_i, Y_i)\}_{1 \leq i \leq n}$ where for all $1 \leq i \leq n$, $X_i \in \mathcal{X}$ is an individual which has been associated with the class $Y_i \in \{-1, 1\}$. Using these training examples, supervised learning aims at designing a *classifier* $h: \mathcal{X} \rightarrow \{-1, 1\}$ employed to predict the label of new individuals. The source codes in R and/or Python of this chapter may be found at:

<https://sylvainlc.github.io/project/teaching/>

2.1 Bayes classifier

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space. Assume that (X, Y) is a couple of random variables defined on $(\Omega, \mathcal{F}, \mathbb{P})$ and taking values in $\mathcal{X} \times \{-1, 1\}$ where \mathcal{X} is a given state space. One aim of supervised classification is to define a function $h: \mathcal{X} \rightarrow \{-1, 1\}$, called *classifier*, such that $h(X)$ is the best prediction of Y in a given context. For instance, the probability of misclassification of h is

$$L_{\text{miss}}(h) = \mathbb{P}(Y \neq h(X)) .$$

Note that $\mathbb{E}[Y|X]$ is a random variable measurable with respect to the σ -algebra $\sigma(X)$. Therefore, there exists a function $\eta: \mathcal{X} \rightarrow [-1, 1]$ so that $\mathbb{E}[Y|X] = \eta(X)$ almost surely.

Lemma 2.1 The classifier h_* , defined for all $x \in \mathcal{X}$, by

$$h_*(x) = \begin{cases} 1 & \text{if } \eta(x) > 0, \\ -1 & \text{otherwise,} \end{cases}$$

is such that

$$h_* = \arg \min_{h: \mathcal{X} \rightarrow \{-1, 1\}} L_{\text{miss}}(h).$$

PROOF. For all $u, v \in \{-1, 1\}$, $\mathbb{1}\{u \neq v\} = \mathbb{1}\{uv = -1\} = (1 - uv)/2$. Since Y and $h(X)$ take values in $\{-1, 1\}$, this implies

$$L_{\text{miss}}(h) = \mathbb{P}(Y \neq h(X)) = (1 - \mathbb{E}[Yh(X)]) / 2. \quad (2.1)$$

Now, using successively the tower property, the equality $|u| = u \times \text{sgn}(u)$, and the tower property again,

$$\mathbb{E}[Yh(X)] = \mathbb{E}[\mathbb{E}[Y|X]h(X)] \leq \mathbb{E}[|\mathbb{E}[Y|X]| |h(X)|] = \mathbb{E}[\mathbb{E}[Y|X] \underbrace{\text{sgn}(\mathbb{E}[Y|X])}_{h_*(X)}] = \mathbb{E}[Yh_*(X)]$$

Plugging this into (2.1) yields $L_{\text{miss}}(h) \geq L_{\text{miss}}(h_*)$, which concludes the proof. \blacksquare

Note that

$$\mathbb{E}[Y|X] = \mathbb{P}(Y = 1|X) - \mathbb{P}(Y = -1|X) = 2\mathbb{P}(Y = 1|X) - 1,$$

which motivates this alternative definition of h_* .

Definition 2.2. The classifier h_* is called the Bayes classifier. It may also be written as follows:

$$h_*(X) = \begin{cases} 1 & \text{if } \mathbb{P}(Y = 1|X) > 1/2 \text{ i.e. if } \mathbb{P}(Y = 1|X) > \mathbb{P}(Y = -1|X), \\ -1 & \text{otherwise.} \end{cases}$$

The Bayes classifier is the optimal choice to minimize the probability of misclassification L_{miss} . However, as the conditional distribution of Y given X is usually unknown, it cannot be computed explicitly. Supervised classification aims at designing an approximate classifier \hat{h}_n using independent observations $(X_i, Y_i)_{1 \leq i \leq n}$ with the same distribution as (X, Y) so that the error $L_{\text{miss}}(\hat{h}_n) - L_{\text{miss}}(h_*)$ may be controlled.

2.2 Parametric and semiparametric classifiers

2.2.1 Mixture of Gaussian distributions

In this first example, we consider a *parametric model*, that is, we assume that the joint distribution of (X, Y) belongs to a family of distributions parametrized by a vector θ with real components. For $k \in \{-1, 1\}$, write $\pi_k = \mathbb{P}(Y = k)$. Assume that $\mathcal{X} = \mathbb{R}^d$ and that, conditionally on the event $\{Y = k\}$, X has a Gaussian distribution with mean $\mu_k \in \mathbb{R}^d$ and covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$, whose density is denoted g_k . In this case, the parameter $\theta = (\pi_1, \mu_1, \mu_{-1}, \Sigma)$ belongs to the set $\Theta = [0, 1] \times \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^{d \times d}$. The parameter π_{-1} is not part of the components of θ since $\pi_{-1} = 1 - \pi_1$. In this case, the parameter $\theta = (\pi_1, \mu_1, \mu_{-1}, \Sigma)$ belongs to the set $\Theta = [0, 1] \times \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^{d \times d}$. The parameter π_{-1} is not part of the components of θ since $\pi_{-1} = 1 - \pi_1$. The explicit computation of $\mathbb{P}(Y = 1|X)$ writes

$$\mathbb{P}(Y = 1|X) = \frac{\pi_1 g_1(X)}{\pi_1 g_1(X) + \pi_{-1} g_{-1}(X)} = \frac{1}{1 + \frac{\pi_{-1} g_{-1}(X)}{\pi_1 g_1(X)}} = \sigma(\log(\pi_1/\pi_{-1}) + \log(g_1(X)/g_{-1}(X))),$$

where $\sigma : x \mapsto (1 + e^{-x})^{-1}$ is the sigmoid function. Then,

$$\mathbb{P}(Y = 1|X) = \sigma(X^T \omega + b), \quad (2.2)$$

where

$$\omega = \Sigma^{-1}(\mu_1 - \mu_{-1}), b = \log(\pi_1/\pi_{-1}) + \frac{1}{2}(\mu_1 + \mu_{-1})' \Sigma^{-1}(\mu_{-1} - \mu_1).$$

Since

$$\mathbb{E}[Y|X] = \mathbb{P}(Y = 1|X) - \mathbb{P}(Y = -1|X),$$

the Bayes classifier is such that for all $x \in \mathcal{X}$,

$$\begin{aligned} h_*(x) = 1 &\Leftrightarrow \mathbb{P}(Y = 1|X)|_{X=x} > \mathbb{P}(Y = -1|X)|_{X=x}, \\ &\Leftrightarrow \pi_1 \exp\left\{-\frac{1}{2}(x - \mu_1)^T \Sigma^{-1}(x - \mu_1)\right\} > \pi_{-1} \exp\left\{-\frac{1}{2}(x - \mu_{-1})^T \Sigma^{-1}(x - \mu_{-1})\right\}, \\ &\Leftrightarrow \log\left(\frac{\pi_1}{\pi_{-1}}\right) > -\frac{1}{2}(x - \mu_{-1})^T \Sigma^{-1}(x - \mu_{-1}) + \frac{1}{2}(x - \mu_1)^T \Sigma^{-1}(x - \mu_1), \\ &\Leftrightarrow \log\left(\frac{\pi_1}{\pi_{-1}}\right) > x^T \Sigma^{-1}(\mu_{-1} - \mu_1) + \frac{1}{2}(\mu_1 + \mu_{-1})^T \Sigma^{-1}(\mu_1 - \mu_{-1}). \end{aligned}$$

In this case, the Bayes classifier is given by

$$h_* : x \mapsto \begin{cases} 1 & \text{if } \left\langle \Sigma^{-1}(\mu_1 - \mu_{-1}); x - \frac{\mu_1 + \mu_{-1}}{2} \right\rangle + \log\left(\frac{\pi_1}{\pi_{-1}}\right) > 0, \\ -1 & \text{otherwise,} \end{cases}$$

When Σ and μ_1 and μ_{-1} are unknown, this classifier cannot be computed explicitly. We will approximate it using the observations. Assume that $(X_i, Y_i)_{1 \leq i \leq n}$ are independent observations with the same distribution as (X, Y) . The loglikelihood of these observations is given by

$$\begin{aligned} \log \mathbb{P}_\theta(X_{1:n}, Y_{1:n}) &= \sum_{i=1}^n \log \mathbb{P}_\theta(X_i, Y_i), \\ &= -\frac{nd}{2} \log(2\pi) + \sum_{i=1}^n \sum_{k \in \{-1, 1\}} \mathbb{1}_{Y_i=k} \left(\log \pi_k - \frac{\log \det \Sigma}{2} - \frac{1}{2}(X_i - \mu_k)^T \Sigma^{-1}(X_i - \mu_k) \right), \\ &= -\frac{nd}{2} \log(2\pi) - \frac{n}{2} \log \det \Sigma + \left(\sum_{i=1}^n \mathbb{1}_{Y_i=1} \right) \log \pi_1 + \left(\sum_{i=1}^n \mathbb{1}_{Y_i=-1} \right) \log(1 - \pi_1) \\ &\quad - \frac{1}{2} \sum_{i=1}^n \mathbb{1}_{Y_i=1} (X_i - \mu_1)^T \Sigma^{-1}(X_i - \mu_1) - \frac{1}{2} \sum_{i=1}^n \mathbb{1}_{Y_i=-1} (X_i - \mu_{-1})^T \Sigma^{-1}(X_i - \mu_{-1}). \end{aligned}$$

By Lemma 5.4, the gradient of $\log \mathbb{P}_\theta(X_{1:n}, Y_{1:n})$ with respect to θ is therefore given by

$$\begin{aligned} \frac{\partial \log \mathbb{P}_\theta(X_{1:n}, Y_{1:n})}{\partial \pi_1} &= \left(\sum_{i=1}^n \mathbb{1}_{Y_i=1} \right) \frac{1}{\pi_1} - \left(\sum_{i=1}^n \mathbb{1}_{Y_i=-1} \right) \frac{1}{1 - \pi_1}, \\ \frac{\partial \log \mathbb{P}_\theta(X_{1:n}, Y_{1:n})}{\partial \mu_1} &= \sum_{i=1}^n \mathbb{1}_{Y_i=1} (2\Sigma^{-1}X_i - 2\Sigma^{-1}\mu_1), \\ \frac{\partial \log \mathbb{P}_\theta(X_{1:n}, Y_{1:n})}{\partial \mu_{-1}} &= \sum_{i=1}^n \mathbb{1}_{Y_i=-1} (2\Sigma^{-1}X_i - 2\Sigma^{-1}\mu_{-1}), \\ \frac{\partial \log \mathbb{P}_\theta(X_{1:n}, Y_{1:n})}{\partial \Sigma^{-1}} &= \frac{n}{2} \Sigma - \frac{1}{2} \sum_{i=1}^n \mathbb{1}_{Y_i=1} (X_i - \mu_1)(X_i - \mu_1)^T - \frac{1}{2} \sum_{i=1}^n \mathbb{1}_{Y_i=-1} (X_i - \mu_{-1})(X_i - \mu_{-1})^T. \end{aligned}$$

The maximum likelihood estimator is defined as the only parameter $\hat{\theta}^n$ such that all these equations are set to 0. For $k \in \{-1, 1\}$, it is given by

$$\begin{aligned}\hat{\pi}_k^n &= \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{Y_i=k}, \\ \hat{\mu}_k^n &= \frac{1}{\sum_{i=1}^n \mathbb{1}_{Y_i=k}} \sum_{i=1}^n \mathbb{1}_{Y_i=k} X_i, \\ \hat{\Sigma}^n &= \frac{1}{n} \sum_{i=1}^n (X_i - \hat{\mu}_{Y_i}^n) (X_i - \hat{\mu}_{Y_i}^n)^T.\end{aligned}$$

Therefore, a natural surrogate for the bayes classifier is

$$\hat{h}_n : x \mapsto \begin{cases} 1 & \text{if } \langle \hat{\Omega}^n (\hat{\mu}_1^n - \hat{\mu}_{-1}^n); x - \frac{\hat{\mu}_1^n + \hat{\mu}_{-1}^n}{2} \rangle + \log \left(\frac{\hat{\pi}_1^n}{\hat{\pi}_{-1}^n} \right) > 0, \\ -1 & \text{otherwise,} \end{cases}$$

where $\hat{\Omega}^n = (\hat{\Sigma}^n)^{-1}$. From the asymptotic properties of the Maximum Likelihood Estimator as n goes to infinity, this classifier converges almost surely to the Bayes classifier as the number of observations n tends to infinity.

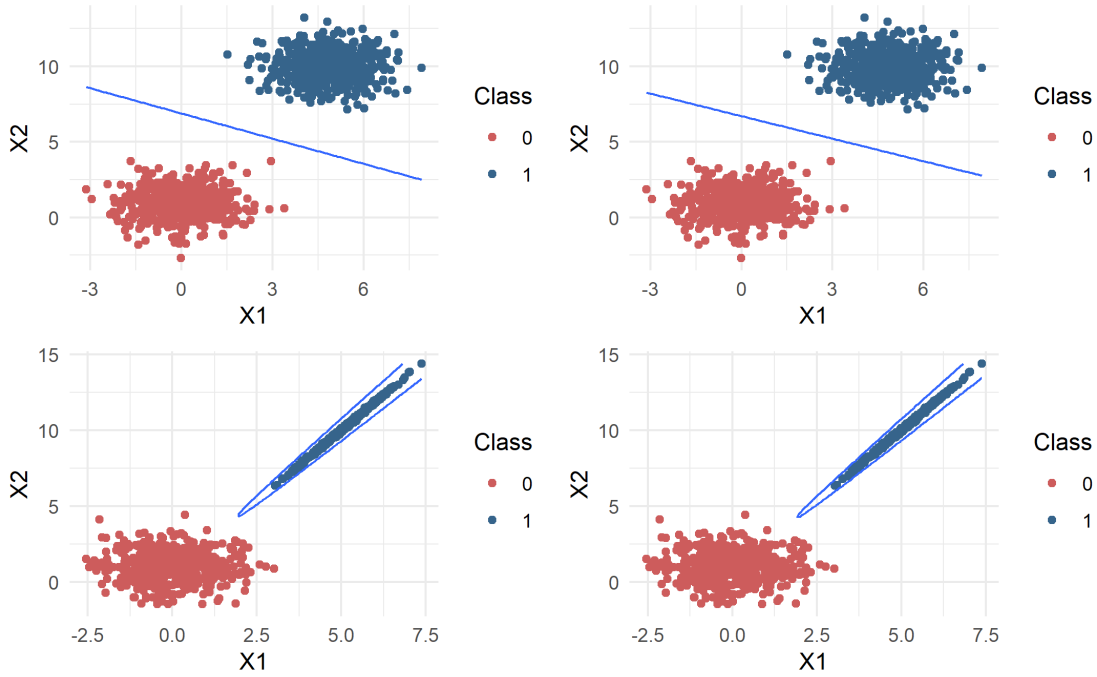


Fig. 2.1 (Top) linear discriminant analysis when both classes are parameterized by the same covariance matrix with the true parameters (left) and maximum likelihood estimates (right). (Bottom) quadratic classification frontier when classes are parameterized by difference covariance matrices with the true parameters (left) and maximum likelihood estimates (right).

2.2.2 Logistic regression

In some situations, it may be too restrictive to assume that the joint distribution of (X, Y) belongs to a parametric family. Instead, since the Bayes classifier defined in Lemma 2.1 only depends on the conditional distribution of Y given X , we only assume that this *conditional distribution* depends on a parameter. The model is said to be *semiparametric* instead of parametric. In the case where $\mathcal{X} = \mathbb{R}^d$, one of the most widely spread model for this conditional distribution is the *logistic regression* which is defined by

$$\mathbb{P}(Y = 1|X) = \sigma(\alpha + \beta^T X), \quad (2.3)$$

where $\alpha \in \mathbb{R}$, $\beta \in \mathbb{R}^d$ and σ is the sigmoid function. The parameter θ is thus $\theta = (\alpha, \beta) \in \mathbb{R} \times \mathbb{R}^d$. Note that for all $x \in \mathcal{X}$,

$$\begin{aligned} \sigma(\alpha + \beta^T x) &= \frac{1}{1 + e^{-\alpha - \langle \beta; x \rangle}}, \\ 1 - \sigma(\alpha + \beta^T x) &= \frac{1}{1 + e^{\alpha + \langle \beta; x \rangle}}, \\ \log \left(\frac{\sigma(\alpha + \beta^T x)}{1 - \sigma(\alpha + \beta^T x)} \right) &= \alpha + \langle \beta; x \rangle. \end{aligned}$$

The Bayes classifier is then given by

$$h_* : x \mapsto \begin{cases} 1 & \text{if } \alpha + \langle \beta; x \rangle > 0, \\ -1 & \text{otherwise.} \end{cases}$$

When α and β are unknown, this classifier cannot be computed explicitly and is approximated using the observations. Assume that $(X_i, Y_i)_{1 \leq i \leq n}$ are independent observations with the same distribution as (X, Y) . The conditional likelihood of the observations $Y_{1:n}$ given $X_{1:n}$ is:

$$\begin{aligned} \mathbb{P}_\theta(Y_{1:n}|X_{1:n}) &= \prod_{i=1}^n \mathbb{P}_\theta(Y_i|X_i), \\ &= \prod_{i=1}^n (\sigma_{\alpha, \beta})^{(1+Y_i)/2}(X_i) (1 - \sigma_{\alpha, \beta})^{(1-Y_i)/2}(X_i), \\ &= \prod_{i=1}^n \left(\frac{e^{\alpha + \langle \beta; X_i \rangle}}{1 + e^{\alpha + \langle \beta; X_i \rangle}} \right)^{(1+Y_i)/2} \left(\frac{1}{1 + e^{\alpha + \langle \beta; X_i \rangle}} \right)^{(1-Y_i)/2}. \end{aligned}$$

The associated conditional loglikelihood is therefore

$$\begin{aligned} \log \mathbb{P}_\theta(Y_{1:n}|X_{1:n}) &= \sum_{i=1}^n \left\{ \frac{1+Y_i}{2} \log \left(\frac{e^{\alpha + \langle \beta; X_i \rangle}}{1 + e^{\alpha + \langle \beta; X_i \rangle}} \right) + \frac{1-Y_i}{2} \log \left(\frac{1}{1 + e^{\alpha + \langle \beta; X_i \rangle}} \right) \right\}, \\ &= \sum_{i=1}^n \left\{ \frac{1+Y_i}{2} (\alpha + \langle \beta; X_i \rangle) - \log(1 + e^{\alpha + \langle \beta; X_i \rangle}) \right\}. \end{aligned}$$

This conditional loglikelihood function cannot be maximized explicitly yet numerous numerical optimization methods are available to maximize $(\alpha, \beta) \mapsto \log \mathbb{P}_\theta(Y_{1:n}|X_{1:n})$. If $(\hat{\alpha}_n, \hat{\beta}_n)$ is an approximate solution to the optimization problem:

$$(\hat{\alpha}_n, \hat{\beta}_n) \in \arg \max_{\theta = (\alpha, \beta) \in \mathbb{R} \times \mathbb{R}^d} \log \mathbb{P}_\theta(Y_{1:n}|X_{1:n}), \quad (2.4)$$

then the associated logistic regression classifier is given by

$$\hat{h}_n : x \mapsto \begin{cases} 1 & \text{if } \hat{\alpha}_n + \langle \hat{\beta}_n; x \rangle > 0, \\ -1 & \text{otherwise,} \end{cases}$$

Even though, the model is semiparametric (and not parametric), it can be shown that, specifically for logistic regression model, the approximated classifier almost surely tends to the Bayes classifier as the number of observations n tends to infinity.

2.2.3 Feed Forward Neural Networks - Multi layer perceptron

Following the logistic regression approach, the multi layer perceptron provides a parametric function to model $\mathbb{P}(Y = k|X)$ for each possible class k . The first mathematical model for a neuron was the Threshold Logic Unit (McCulloch and Pitts, 1943), with Boolean inputs and outputs. In this setting, the response associated with an input $x \in \{0, 1\}^d$ is defined as

$$f : x \mapsto \mathbb{1}_{\omega \sum_{j=1}^d x_j + b \geq 0}.$$

This construction allows to build any boolean function from elementary units

$$x \vee y = \mathbb{1}_{x+y-1/2 \geq 0}, \quad x \wedge y = \mathbb{1}_{x+y-3/2 \geq 0} \quad \text{and} \quad 1 - x = \mathbb{1}_{-x+1/2 \geq 0}$$

This elementary model can be extended to real valued inputs (Rosenblatt, 1957) with

$$f : x \mapsto \mathbb{1}_{\sum_{j=1}^d \omega_j x_j + b \geq 0}.$$

In this case, the nonlinear activation function is $\sigma : x \mapsto \mathbb{1}_{x \geq 0}$ and the output in $\{0, 1\}$ defined as:

$$f : x \mapsto \sigma(\omega^T x + b).$$

Linear discriminant analysis and logistic regression are other instances with the sigmoid activation function $\sigma : x \mapsto e^x / (1 + e^x)$ and the output $\sigma(\omega^T X + b)$ in $(0, 1)$ is $\mathbb{P}(Y = 1|X)$. The perceptron weakens the

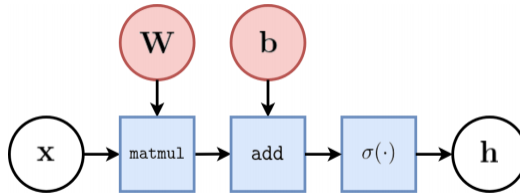


Fig. 2.2 Perceptron/LDA/Logistic regression. Schematic view of one layer.

modeling assumptions of LDA or logistic regression and composed in parallel q of these perceptron units to produce the output. Then, $x = h_0 \in \mathbb{R}^d$, $b_1 \in \mathbb{R}^q$, $\omega_1 \in \mathbb{R}^{d \times q}$ and

$$h_1 = \sigma(\omega_1^T x + b_1),$$

with σ the elementwise activation function. For all $1 \leq q$, the j -th component of h_1 is obtained by an application of the activation transform to an affine transform of x . The multi-layer perceptron, also known as the fully connected feed forward network, connects these units in series. For a given number L of layers,

$$h_1 = \sigma(\omega_1^T x + b_1), \quad h_2 = \sigma(\omega_2^T h_1 + b_2), \quad \dots, \quad h_L = \sigma(\omega_L^T h_{L-1} + b_L).$$

As there is no modelling assumptions anymore, virtually any activation function may be used. The rectified

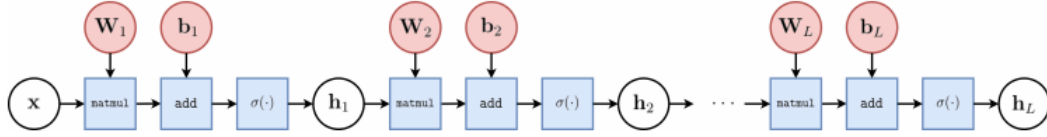


Fig. 2.3 Multiple layer perceptron.

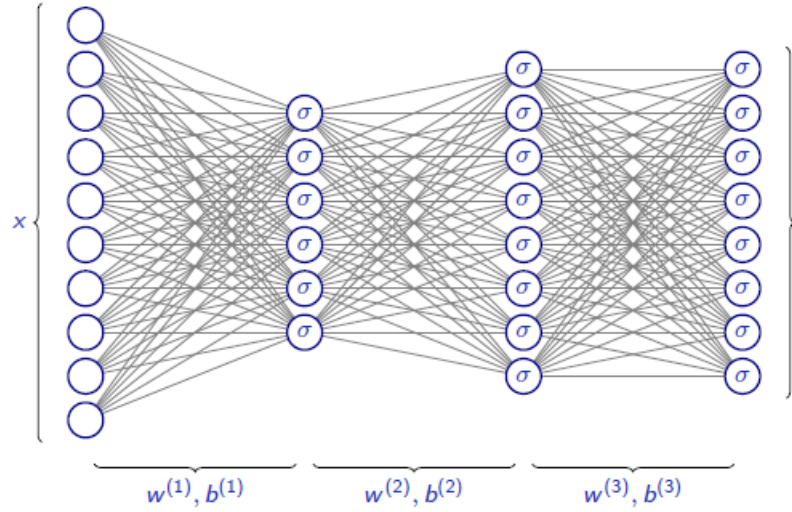


Fig. 2.4 Graphical model of the multiple layer perceptron.

linear unit (RELU) activation function $\sigma(x) = \max(0, x)$ and its extensions are the default recommendation in modern implementations (Jarrett et al., 2009; Nair and Hinton, 2010; Glorot et al., 2011a), (Maas et al., 2013), (He et al., 2015). One of the major motivations arise from the gradient based parameter optimization which is numerically more stable with this choice. Assume that the network contains L layers, then the output layer is of the form:

$$h_L = \sigma(\omega_L^T h_{L-1} + b_L) .$$

The choice of this last activation function greatly relies on the task the network is assumed to perform.

- **biclass classification.** The output h_L is the estimate of the probability that the class is 1 given the input X . The common choice in this case is the sigmoid function. Then, h_L contains $\mathbb{P}(Y = 1|X)$ and is enough to use as a plug-in Bayes classifier.
- **multiclass classification.** The output h_L is the estimate of the probability that the class is k for all $1 \leq k \leq M$, given the input x . The common choice in this case is the softmax function: for all $1 \leq i \leq M$

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^M e^{z_j}} .$$

Each component k of z_L contains $\mathbb{P}(Y = k|X)$.

These Feed Forward Neural Networks may be used to perform classification on the MNIST dataset. This dataset contains images representing handwritten digits. Each image is made of 28 x 28 pixels, and each pixel is represented by an integer (gray level). These arrays can be flattened into vectors in \mathbb{R}^{784} . Visualisations of this vector space are given here: <http://colah.github.io/posts/2014-10-Visualizing-MNIST/>. The labels in $\{0, \dots, 9\}$ are represented using one-hot-encoding and grayscale of each pixel in $\{0, \dots, 255\}$ are normalized to be in $(0, 1)$.

```
from keras.datasets import mnist
# Number of classes
num_classes = 10
# input image dimensions
img_rows, img_cols = 28, 28

# the data, shuffled and split between train and test sets
(x_train, y_train), (x_test, y_test) = mnist.load_data()
x_train = x_train.reshape(x_train.shape[0], img_rows, img_cols, 1)
x_test = x_test.reshape(x_test.shape[0], img_rows, img_cols, 1)
input_shape = (img_rows, img_cols, 1)

x_train = x_train.astype('float32')
x_test = x_test.astype('float32')

print('x_train shape:', x_train.shape)
print('x_test shape:', x_test.shape)
print('y_train shape:', y_train.shape)
print('y_test shape:', y_test.shape)
print(x_train.shape[0], 'train samples')
print(x_test.shape[0], 'test samples')
```

```
x_train shape: (60000, 28, 28, 1)
x_test shape: (10000, 28, 28, 1)
y_train shape: (60000,)
y_test shape: (10000,)
60000 train samples
10000 test samples
```

```
plt.figure(figsize=(8, 2))
for i in range(4):
    plt.subplot(1, 4, i+1)
    plt.imshow(x_train[i].reshape(28, 28),
               interpolation="none", cmap="gray_r")
    plt.title('Label = %d' % y_train[i], fontsize=14)
    plt.axis("off")
plt.tight_layout()
```

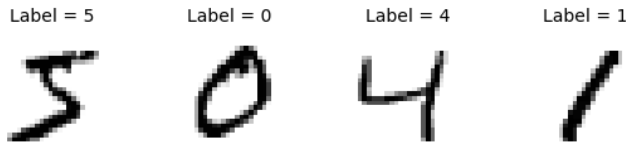


Fig. 2.5 MNIST dataset.

This model relies on more than 100,000 unknown parameters which should be estimated. As for the logistic regression and the discriminant analysis, a common choice is to minimize the negative loglikelihood of the data:

$$\theta \mapsto -\frac{1}{n} \sum_{i=1}^n \sum_{k=1}^{10} \mathbb{1}_{Y_i=k} \log \mathbb{P}_{\theta}(Y_i = k | \mathbf{X}_i) .$$

The negative loglikelihood is computed using $n = 60,000$ training samples and minimized using gradient descent algorithms. Then, the performance of the model is assessed using 10,000 new (test) samples: the accuracy is the frequency of labels which are well predicted by the model with the estimated parameters.

```

model_ffnn = Sequential()

model_ffnn.add(Flatten(input_shape=input_shape))

model_ffnn.add(Dense(128, activation='relu'))

model_ffnn.add(Dense(num_classes, activation='softmax'))

model_ffnn.compile(
    loss=keras.losses.categorical_crossentropy,
    optimizer=keras.optimizers.Adagrad(),
    metrics=['accuracy']
)

model_ffnn.summary()

```

Layer (type)	Output Shape	Param #
flatten_1 (Flatten)	(None, 784)	0
dense_1 (Dense)	(None, 128)	100480
dense_2 (Dense)	(None, 10)	1290
Total params: 101,770		
Trainable params: 101,770		
Non-trainable params: 0		

Fig. 2.6 Feed Forward Neural network. h_1 is obtained with the RELU activation function and is in \mathbb{R}^{128} . The last layer is $h_2 \in \mathbb{R}^{10}$ and is obtained with the softmax activation function so that each component k models $\mathbb{P}(Y = k|X)$. This neural network with one hidden layer relies on 101.770 parameters.

2.2.4 Support vector machines

Hard Support Vector Machines

Hard Support Vector Machines is a classification procedure which aims at building a linear classifier with the largest possible margin, i.e. the largest minimal distance between a point in the training set and the hyperplane. The objective is to find a hyperplane which correctly separates all training data points by maximizing the closest distance from this hyperplane of a point in the training data set. Let $H_{w,b}$ be the hyperplane of \mathbb{R}^d with orthogonal vector w and offset b :

$$H_{w,b} = \{x \in \mathbb{R}^d ; \langle w ; x \rangle + b = 0\} .$$

Following for instance the results obtained for linear discriminant analysis and logistic regression, a hyperplane $H_{w,b}$ may be used as a classifier by defining

$$h_{w,b} : x \mapsto \begin{cases} 1 & \text{if } \langle w ; x \rangle + b > 0 , \\ -1 & \text{otherwise} . \end{cases}$$

Hard Support Vector Machines are applied in the case where the training data points are linearly separable which means that there exists a linear classifier $h_{w,b}$ with $(w,b) \in \mathbb{R}^d \times \mathbb{R}$, $\|w\| = 1$, which classifies correctly all observed data, for all $1 \leq i \leq n$,

$$Y_i (\langle w ; X_i \rangle + b) > 0 .$$

The distance between any $x \in \mathbb{R}^d \setminus H_{w,b}$ and $H_{w,b}$ is given by

$$d(x, H_{w,b}) = |\langle w ; x \rangle + b| .$$

```

batch_size = 32
epochs = 8

# Run the train
history = model_ffnn.fit(x_train, y_train,
                        batch_size=batch_size,
                        epochs=epochs,
                        verbose=1,
                        validation_data=(x_test, y_test))
score = model_ffnn.evaluate(x_test, y_test, verbose=0)
print('Test loss:', score[0])
print('Test accuracy:', score[1])

plt.figure(figsize=(5, 4))
plt.plot(history.epoch, history.history['acc'], lw = 1, label='Training')
plt.plot(history.epoch, history.history['val_acc'], lw = 1, label='Testing')
plt.legend()
plt.title('Accuracy of softmax regression', fontsize=16)
plt.xlabel('Epoch', fontsize=14)
plt.ylabel('Accuracy', fontsize=14)
plt.tick_params(labelright=True)
plt.grid('True')
plt.tight_layout()

```

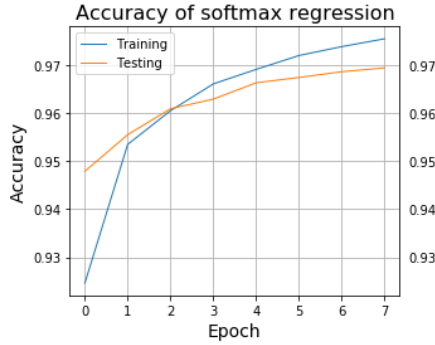


Fig. 2.7 Minimization of the negative likelihood using a gradient descent algorithm (here AdaGrad). The gradient is computed using batches of 32 observations and the whole data set is used 8 times.

Therefore, the hyperplane which correctly separates all training data sets with the largest margin is $H_{\hat{w}_n, \hat{b}_n}$ where

$$(\hat{w}_n, \hat{b}_n) \in \arg \max_{\substack{(w, b) \in \mathbb{R}^d \times \mathbb{R}^d; \|w\|=1, \\ \forall i \in \{1, \dots, n\}, Y_i(\langle w; X_i \rangle + b) > 0}} \left\{ \min_{1 \leq i \leq n} |\langle w; X_i \rangle + b| \right\}.$$

Proposition 2.3 *The hard Support Vector Machines procedure is equivalent to solving the following optimization problem:*

$$(\hat{w}_n, \hat{b}_n) \in \arg \max_{(w, b) \in \mathbb{R}^d \times \mathbb{R}; \|w\|=1} \left\{ \min_{1 \leq i \leq n} Y_i(\langle w; X_i \rangle + b) \right\}, \quad (2.5)$$

PROOF. Let (\hat{w}_n, \hat{b}_n) be a solution to (2.5). As the training data set is assumed to be linearly separable, there exist $(w, b) \in \mathbb{R}^d \times \mathbb{R}$ with $\|w\| = 1$ such that for all $1 \leq i \leq n$, $Y_i(\langle w; X_i \rangle + b) > 0$. By definition of (\hat{w}_n, \hat{b}_n) ,

$$\min_{1 \leq i \leq n} Y_i(\langle \hat{w}_n; X_i \rangle + \hat{b}_n) \geq \min_{1 \leq i \leq n} Y_i(\langle w; X_i \rangle + b) > 0$$

and then (\hat{w}_n, \hat{b}_n) satisfies the constraints of the hard Support Machines problem. On the other hand, for any $(w, b) \in \mathbb{R}^d \times \mathbb{R}$ satisfying these constraints, for all $1 \leq i \leq n$, Y_i and $\langle w; X_i \rangle + b$ have the same sign,

$$\min_{1 \leq i \leq n} Y_i(\langle w; X_i \rangle + b) = \min_{1 \leq i \leq n} |\langle w; X_i \rangle + b|,$$

which concludes the proof. \blacksquare

Proposition 2.4 Define $(\hat{w}_n, \hat{b}_n) = (w_\star / \|w_\star\|, b_\star / \|w_\star\|)$ where

$$(w_\star, b_\star) \in \arg \min_{\substack{(w,b) \in \mathbb{R}^d \times \mathbb{R} \\ \forall i \in \{1, \dots, n\}, Y_i(\langle w; X_i \rangle + b) \geq 1}} \|w\|^2. \quad (2.6)$$

Then, (\hat{w}_n, \hat{b}_n) is a solution to (2.5).

PROOF. Let (w_\star, b_\star) be a solution to (2.6) and define $(\hat{w}_n, \hat{b}_n) = (w_\star / \|w_\star\|, b_\star / \|w_\star\|)$. Therefore, $\|\hat{w}_n\| = 1$ and

$$\min_{1 \leq i \leq n} Y_i(\langle \hat{w}_n; X_i \rangle + \hat{b}_n) \geq \|w_\star\|^{-1}.$$

In addition, if $(w, b) \in \mathbb{R}^d \times \mathbb{R}$ is solution to (2.5),

$$\delta_\star = \min_{1 \leq i \leq n} Y_i(\langle w; X_i \rangle + b)$$

so that $(w/\delta_\star, b/\delta_\star)$ satisfies the constraints of (2.6) which yields $\|w_\star\| \leq \|w\|/\delta_\star \leq 1/\delta_\star$. Then, $\delta_\star \leq \|w_\star\|^{-1}$ which proves that (\hat{w}_n, \hat{b}_n) is a solution to (2.5). \blacksquare

The hard SVM problem displayed in Proposition 2.4 is a convex quadratic problem with linear constraints. When the training dataset is assumed to be linearly separable, this problem has a unique solution which may be approximated efficiently using one of the numerous numerical optimization algorithms such as stochastic gradient procedures. On the other hand, the dual problem associated with Proposition 2.4 offers an alternative to obtain a solution to hard SVM. The Lagrangian function associated with this problem is

$$\mathcal{L} : (w, b, \mu) \mapsto \|w\|^2 + \sum_{i=1}^n \mu_i (1 - Y_i(\langle w; X_i \rangle + b))$$

and the Lagrangian dual problem is

$$\hat{\mu} \in \arg \max_{\mu_i \geq 0} \left\{ \inf_{(w,b) \in \mathbb{R}^d \times \mathbb{R}} \mathcal{L}(w, b, \mu) \right\}.$$

As the function to be minimized is convex and continuously differentiable and the inequality constraints are affine, $(\hat{w}, \hat{b}) \in \mathbb{R}^{d+1}$ is solution to (2.4) if and only if there exists $\mu \in \mathbb{R}^d$ such that, for all $1 \leq i \leq n$, $\partial_{w_i} \mathcal{L}(\hat{w}, \hat{b}, \mu) = 0$, $\partial_b \mathcal{L}(\hat{w}, \hat{b}, \mu) = 0$, $Y_i(\langle \hat{w}; X_i \rangle + \hat{b}) \leq 1$, $\mu_i \geq 0$, and $\mu_i(Y_i(\langle \hat{w}; X_i \rangle + \hat{b}) - 1) = 0$. This yields

$$\hat{w} = 2^{-1} \sum_{i=1}^n \mu_i Y_i X_i \quad \text{and} \quad \sum_{i=1}^n \mu_i Y_i = 0.$$

For all $1 \leq i \leq n$, if $Y_i(\langle \hat{w}; X_i \rangle + \hat{b}) > 1$ and then $\mu_i = 0$ and the i -th observation is not involved in \hat{w} . Conversely, if $\mu_i > 0$, then $|\langle \hat{w}; X_i \rangle + \hat{b}| = 1$, the margin of all observations involved in \hat{w} is 1. These vectors are called the *support vectors*. Note that μ is solution to the dual problem

$$\mu \in \arg \max_{\substack{\mu_i \geq 0 \\ \sum_{i=1}^n \mu_i Y_i = 0}} \left\{ \sum_{i=1}^n \mu_i - \frac{1}{2} \sum_{1 \leq i, j \leq n} \mu_i \mu_j Y_i Y_j \langle X_i; X_j \rangle \right\}.$$

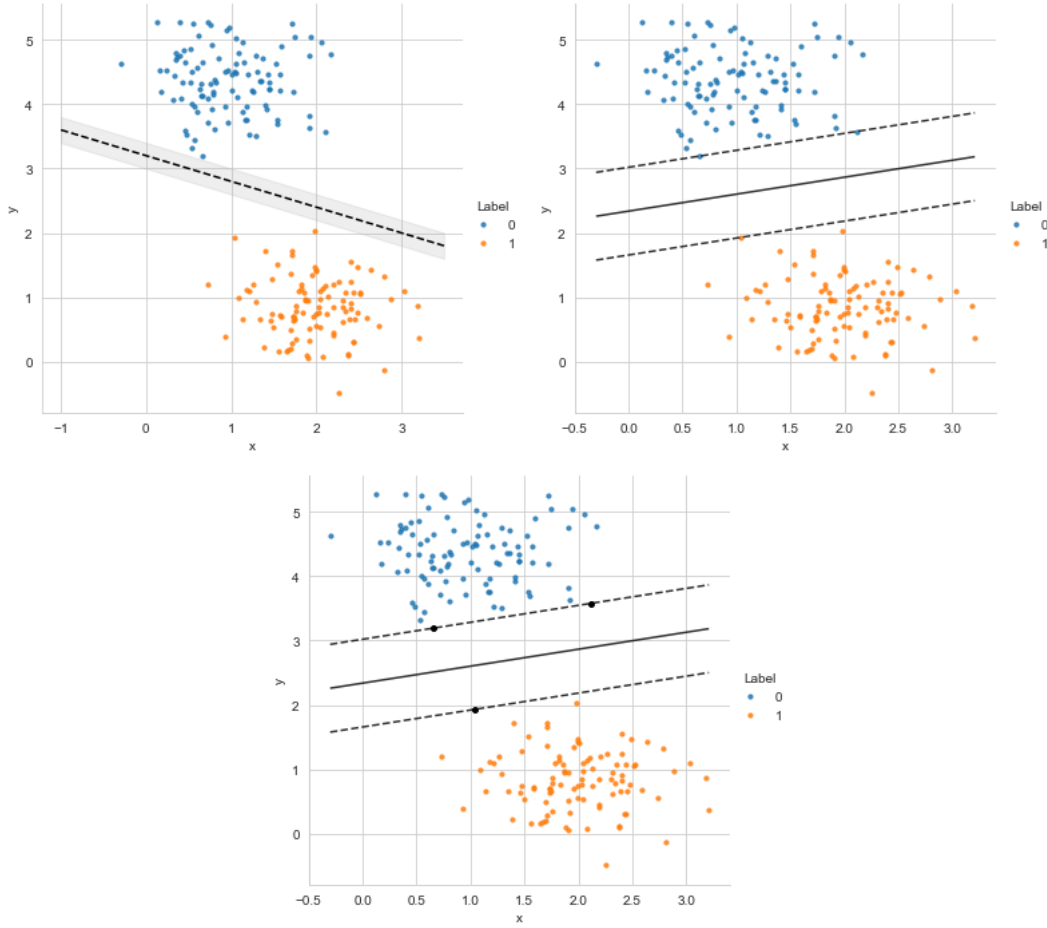


Fig. 2.8 (Top left) Linearly separable dataset, as shown by a hyperplane which separates correctly both classes and (top right) hard SVM classifier and its margin (solid line). (Bottom) hard SVM classifier with its three support vectors (black dots). These support vectors are those such that $\mu_i > 0$ and then $\langle \tilde{w} ; X_i \rangle + \hat{b} \in \{-1 ; 1\}$. They are on the contour lines with level 1 or -1 of the function $x \mapsto \langle \tilde{w} ; x \rangle + \hat{b}$ (dashed lines).

Soft Support Vector Machines

By Proposition 2.4, the hard Support Vector Machines can be reduced to a quadratic optimization problem with linear constraints, which may be solved in a reasonable computational time, when the training data sets is linearly separable. Restricting the problem to linearly separable training data sets is a somehow strong assumption. Soft Support Vector Machines algorithm introduces a relaxation of this constraint which can be applied with nonlinearly separable data sets. In this setting, the inequality constraints in the quadratic optimization problem (2.6) can be relaxed by introducing nonnegative variables $(\xi_i)_{1 \leq i \leq n}$ which quantify for each variable $1 \leq i \leq n$, the nonfeasibility of the constraint $Y_i(\langle w ; X_i \rangle + b) \geq 1$. Therefore, the soft Support Vector Machines algorithm simultaneously the margin of the linear classifier and the average value of these slack variables $(\xi_i)_{1 \leq i \leq n}$:

$$(w_*, b_*, \xi_*) \in \arg \min_{\substack{(w, b, \xi) \in \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}_+^n \\ \forall i \in \{1, \dots, n\}, Y_i(\langle w ; X_i \rangle + b) \geq 1 - \xi_i}} \left\{ \lambda \|w\|^2 + \frac{1}{n} \sum_{i=1}^n \xi_i \right\}, \quad (2.7)$$

where $\lambda > 0$.

Remark 2.5 In this case, for all $1 \leq i \leq n$, if $Y_i(\langle w; X_i \rangle + b) \geq 1$, then $\xi_{*,i} = 0$ and if $Y_i(\langle w; X_i \rangle + b) < 1$ then the optimal choice for $\xi_{*,i}$ is $\xi_{*,i} = 1 - Y_i(\langle w; X_i \rangle + b)$. Therefore, for all $1 \leq i \leq n$,

$$\xi_{*,i} = (1 - Y_i(\langle w; X_i \rangle + b))_+.$$

Conversely, if (w_*, b_*) is solution to

$$(w_*, b_*) \in \arg \min_{(w,b) \in \mathbb{R}^d \times \mathbb{R}} \left\{ \lambda \|w\|^2 + \frac{1}{n} \sum_{i=1}^n (1 - Y_i(\langle w; X_i \rangle + b))_+ \right\}, \quad (2.8)$$

then (w_*, b_*, ξ_*) is solution to (2.7) with for all $1 \leq i \leq n$, $\xi_{*,i} = (1 - Y_i(\langle w; X_i \rangle + b))_+$. Therefore, the linear classifier produced by the soft Support Vector Machines is solution to (2.8). This formulation of the Soft SVM problem motivates the introduction of kernel based SVM detailed at the end of this chapter.

The optimization problem (2.7) can be written

$$(\hat{w}, \hat{\xi}, \hat{b}) \in \arg \min_{\substack{h_i(\xi, w, b) \leq 0, \\ g_i(\xi, w, b) \leq 0}} f(\xi, w, b), \quad (2.9)$$

where $f : (\xi, w, b) \rightarrow n^{-1} \sum_{i=1}^n \xi_i + \lambda \|w\|^2$ and for all $1 \leq i \leq n$, $g_i : (\xi, w, b) \mapsto -\xi_i$ and $h_i : (\xi, w, b) \mapsto 1 - \xi_i - Y_i(\langle w; X_i \rangle + b)$. The Lagrangian function associated with this problem is

$$\mathcal{L} : (\xi, w, b, \mu, \gamma) \mapsto f(\xi, w, b) + \sum_{i=1}^n \mu_i h_i(\xi, w, b) + \sum_{i=1}^n \gamma_i g_i(\xi, w, b)$$

and the Lagrangian dual problem is

$$(\hat{\mu}, \hat{\gamma}) \in \arg \max_{\mu_i \geq 0, \gamma_i \geq 0} \left\{ \inf_{(\xi, w, b) \in \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}} \mathcal{L}(\xi, w, b, \mu, \gamma) \right\}.$$

As the function to be minimized is convex and continuously differentiable and the inequality constraints are affine, $(\hat{w}, \hat{\xi}, \hat{b}) \in \mathbb{R}^{2d+1}$ is solution to (2.16) if and only if there exists $(\mu, \gamma) \in \mathbb{R}^{2d}$ such that, for all $1 \leq i \leq n$, $\partial_{\xi_i} \mathcal{L}(\hat{\xi}, \hat{w}, \hat{b}, \mu, \gamma) = 0$, $\partial_{w_i} \mathcal{L}(\hat{\xi}, \hat{w}, \hat{b}, \mu, \gamma) = 0$ and $\partial_b \mathcal{L}(\hat{\xi}, \hat{w}, \hat{b}, \mu, \gamma) = 0$, $h_i(\hat{\xi}, \hat{w}, \hat{b}) \leq 0$, $g_i(\hat{\xi}, \hat{w}, \hat{b}) \leq 0$, $\mu_i \geq 0$, $\gamma_i \geq 0$, $\mu_i g_i(\hat{\xi}, \hat{w}, \hat{b}) = 0$ and $\gamma_i h_i(\hat{\xi}, \hat{w}, \hat{b}) = 0$. This yields

$$w = (2\lambda)^{-1} \sum_{i=1}^n \mu_i Y_i X_i, \quad \mu_i + \gamma_i = 1/n \quad \text{and} \quad \sum_{i=1}^n \mu_i Y_i = 0.$$

Then $w = (2\lambda)^{-1} \sum_{i=1}^n \mu_i Y_i X_i$ and μ is solution to

$$\mu \in \arg \max_{\substack{0 \leq \mu_i \leq 1/n \\ \sum_{i=1}^n \mu_i Y_i = 0}} \left\{ \sum_{i=1}^n \mu_i - \frac{1}{2\lambda} \sum_{1 \leq i, j \leq n} \mu_i \mu_j Y_i Y_j \langle X_i; X_j \rangle \right\}.$$

Therefore, the normal vector of the hyperplane is a linear combination of the $(X_i)_{1 \leq i \leq n}$ where μ is solution to a quadratic optimization problem which only involves $(X_i)_{1 \leq i \leq n}$ through $(\langle X_i; X_j \rangle)_{1 \leq i, j \leq n}$.

2.3 Nonparametric Bayes classifier

In the case of *nonparametric* models, it is not assumed anymore that the joint law of (X, Y) belongs to any parametric or semiparametric family of models. The assumption on the distribution of (X, Y) is relaxed but instead, we will make some restrictions on the set of classifiers on which the optimisation occurs.

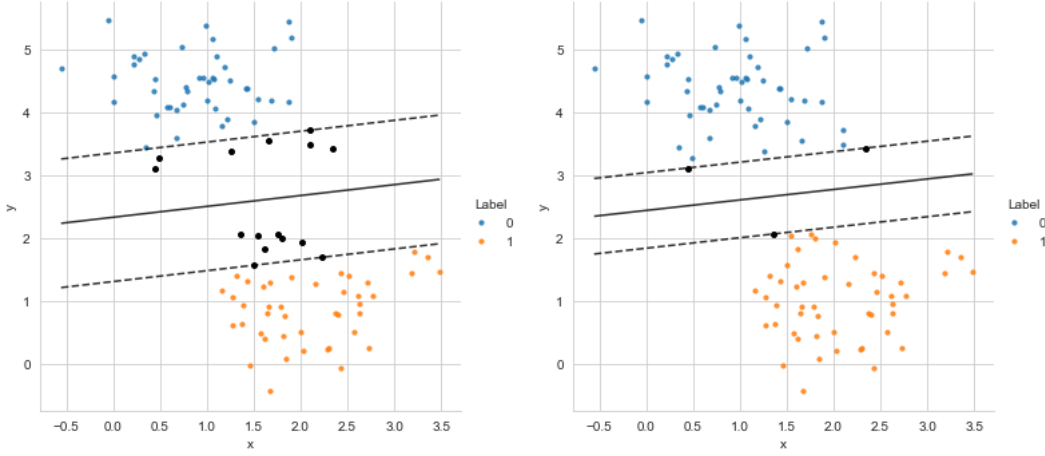


Fig. 2.9 (Left) Soft SVM classifier (solid line) and support vectors (black dots) for a chosen λ_0 . (Right) soft SVM classifier with its with $\lambda = 100\lambda_0$. The support vectors are those such that $\mu_i > 0$ and then $|\langle \hat{w}; X_i \rangle + \hat{b}| < 1$. They are inbetween the contour lines with level 1 or -1 of the function $x \mapsto \langle \hat{w}; x \rangle + \hat{b}$ (dashed lines).

More precisely, we consider that the optimization of classifiers holds on a specific set \mathcal{H} of classifiers (often called the *dictionary*), which may possibly not contain the Bayes classifier. Moreover, since in most cases, the classification risk L_{miss} cannot be computed nor minimized, it is instead estimated by the empirical classification risk defined as

$$\hat{L}_{\text{miss}}^n(h) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{Y_i \neq h(X_i)},$$

where $(X_i, Y_i)_{1 \leq i \leq n}$ are independent observations with the same distribution as (X, Y) . The classification problem then builds down to solving

$$\hat{h}_{\mathcal{H}}^n \in \arg \min_{h \in \mathcal{H}} \hat{L}_{\text{miss}}^n(h). \quad (2.10)$$

In this context several practical and theoretical challenges arise from the minimization of the empirical classification risk. The choice of \mathcal{H} is pivotal in designing an efficient classification procedure. Note that choosing \mathcal{H} as all possible classifiers is meaningless, in this case, $\hat{h}_{\mathcal{H}}^n$ is such that $\hat{h}_{\mathcal{H}}^n(X_i) = Y_i$ for all $1 \leq i \leq n$ and $\hat{h}_{\mathcal{H}}^n(x)$ is any element of $\{-1, 1\}$ for all $x \notin \{X_1, \dots, X_n\}$. Although $\hat{L}_{\text{miss}}^n(\hat{h}_{\mathcal{H}}^n) = 0$, is likely to be a poor approximation of $L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n)$. To understand this, the excess misclassification risk may be decomposed as follows

$$L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - L_{\text{miss}}(h_{\star}) = L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - \min_{h \in \mathcal{H}} L_{\text{miss}}(h) + \min_{h \in \mathcal{H}} L_{\text{miss}}(h) - L_{\text{miss}}(h_{\star}) \geq 0.$$

The first term of the decomposition $L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - \min_{h \in \mathcal{H}} L_{\text{miss}}(h)$ is a **stochastic error** which is likely to grow when the size of \mathcal{H} grows while $\min_{h \in \mathcal{H}} L_{\text{miss}}(h) - L_{\text{miss}}(h_{\star})$ is **deterministic** and likely to decrease as the size of \mathcal{H} grows.

Lemma 2.6 For all set \mathcal{H} of classifiers and all $n \geq 1$,

$$L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - \min_{h \in \mathcal{H}} L_{\text{miss}}(h) \leq 2 \sup_{h \in \mathcal{H}} |\hat{L}_{\text{miss}}^n(h) - L_{\text{miss}}(h)|. \quad (2.11)$$

PROOF. By definition of $\hat{h}_{\mathcal{H}}^n$, for any $h \in \mathcal{H}$,

$$\begin{aligned} L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - \min_{h \in \mathcal{H}} L_{\text{miss}}(h) &= L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - \hat{L}_{\text{miss}}^n(\hat{h}_{\mathcal{H}}^n) + \hat{L}_{\text{miss}}^n(\hat{h}_{\mathcal{H}}^n) - \min_{h \in \mathcal{H}} L_{\text{miss}}(h) , \\ &\leq L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - \hat{L}_{\text{miss}}^n(\hat{h}_{\mathcal{H}}^n) + \hat{L}_{\text{miss}}^n(h) - \min_{h \in \mathcal{H}} L_{\text{miss}}(h) . \end{aligned}$$

For all $\varepsilon > 0$ there exists $h_\varepsilon \in \mathcal{H}$ such that $L_{\text{miss}}(h_\varepsilon) < \min_{h \in \mathcal{H}} L_{\text{miss}}(h) + \varepsilon$ so that

$$\begin{aligned} L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - \min_{h \in \mathcal{H}} L_{\text{miss}}(h) &\leq L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) - \hat{L}_{\text{miss}}^n(\hat{h}_{\mathcal{H}}^n) + \hat{L}_{\text{miss}}^n(h_\varepsilon) - L_{\text{miss}}(h_\varepsilon) + \varepsilon , \\ &\leq 2 \sup_{h \in \mathcal{H}} \left| \hat{L}_{\text{miss}}^n(h) - L_{\text{miss}}(h) \right| + \varepsilon , \end{aligned}$$

which concludes the proof. ■

Oracle inequality when \mathcal{H} is finite

This section considers the simple case where the dictionary is finite, i.e., $\mathcal{H} = \{h_1, \dots, h_M\}$ where $M \geq 1$ and for all $1 \leq j \leq M$, $h_j : \mathcal{X} \rightarrow \{-1, 1\}$ is a given classifier.

Proposition 2.7 Assume that $\mathcal{H} = \{h_1, \dots, h_M\}$, then, for all $\delta > 0$,

$$\mathbb{P} \left(L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) \leq \min_{1 \leq j \leq M} L_{\text{miss}}(h_j) + \sqrt{\frac{2}{n} \log \left(\frac{2M}{\delta} \right)} \right) \geq 1 - \delta .$$

PROOF. By Lemma 2.6, for all $u > 0$,

$$\mathbb{P} \left(L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) > \min_{1 \leq j \leq M} L_{\text{miss}}(h_j) + u \right) \leq \mathbb{P} \left(\sup_{h \in \mathcal{H}} \left| \hat{L}_{\text{miss}}^n(h) - L_{\text{miss}}(h) \right| > \frac{u}{2} \right) \leq \sum_{j=1}^M \mathbb{P} \left(\left| \hat{L}_{\text{miss}}^n(h_j) - L_{\text{miss}}(h_j) \right| > \frac{u}{2} \right) .$$

By Hoeffding's inequality, see Theorem 5.1,

$$\mathbb{P} \left(L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) > \min_{1 \leq j \leq M} L_{\text{miss}}(h_j) + u \right) \leq 2M e^{-nu^2/2} ,$$

which concludes the proof by choosing

$$u = \sqrt{\frac{2}{n} \log \left(\frac{2M}{\delta} \right)} .$$
■

Proposition 2.8 Assume that $\mathcal{H} = \{h_1, \dots, h_M\}$, then,

$$\mathbb{E} \left[L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) \right] \leq \min_{1 \leq j \leq M} L_{\text{miss}}(h_j) + \sqrt{\frac{2 \log(2M)}{n}} .$$

PROOF. By Lemma 2.6,

$$\mathbb{E} \left[L_{\text{miss}}(\hat{h}_{\mathcal{H}}^n) \right] - \min_{1 \leq j \leq M} L_{\text{miss}}(h_j) \leq 2 \mathbb{E} \left[\sup_{h \in \mathcal{H}} \left| \hat{L}_{\text{miss}}^n(h) - L_{\text{miss}}(h) \right| \right] = \frac{2}{n} \mathbb{E} \left[\max_{1 \leq j \leq M} \left\{ n \left| \hat{L}_{\text{miss}}^n(h_j) - L_{\text{miss}}(h_j) \right| \right\} \right] .$$

Note that

$$n \left\{ \widehat{L}_{\text{miss}}^n(h_j) - L_{\text{miss}}(h_j) \right\} = \sum_{i=1}^n \left\{ \mathbb{1}_{Y_i \neq h_j(X_i)} - L_{\text{miss}}(h_j) \right\},$$

where the random variables $(\mathbb{1}_{Y_i \neq h_j(X_i)})_{1 \leq i \leq n}$ are independent Bernoulli random variables with mean $L_{\text{miss}}(h_j)$. By Lemma 5.2, for all $t > 0$,

$$\mathbb{E} \left[\exp \left\{ t \sum_{i=1}^n \left\{ \mathbb{1}_{Y_i \neq h_j(X_i)} - L_{\text{miss}}(h_j) \right\} \right\} \right] = \prod_{i=1}^n \mathbb{E} \left[\exp \left\{ t \left(\mathbb{1}_{Y_i \neq h_j(X_i)} - L_{\text{miss}}(h_j) \right) \right\} \right] \leq e^{nt^2/8}$$

and similarly

$$\mathbb{E} \left[\exp \left\{ -t \sum_{i=1}^n \left\{ \mathbb{1}_{Y_i \neq h_j(X_i)} - L_{\text{miss}}(h_j) \right\} \right\} \right] \leq e^{nt^2/8}.$$

Then, for all $t > 0$, by Jensen's inequality,

$$\begin{aligned} \exp \left\{ t \mathbb{E} \left[\max_{1 \leq j \leq M} \left\{ n \left| \widehat{L}_{\text{miss}}^n(h_j) - L_{\text{miss}}(h_j) \right| \right\} \right] \right\} &\leq \mathbb{E} \left[\exp \left\{ t \max_{1 \leq j \leq M} \left\{ n \left| \widehat{L}_{\text{miss}}^n(h_j) - L_{\text{miss}}(h_j) \right| \right\} \right\} \right] \\ &\leq 2Me^{nt^2/8}, \end{aligned}$$

which yields

$$\mathbb{E} \left[\max_{1 \leq j \leq M} \left\{ n \left| \widehat{L}_{\text{miss}}^n(h_j) - L_{\text{miss}}(h_j) \right| \right\} \right] \leq \frac{\log(2M)}{t} + \frac{nt}{8}.$$

Choosing $t = \sqrt{8 \log(2M)/n}$,

$$\mathbb{E} \left[\max_{1 \leq j \leq M} \left\{ n \left| \widehat{L}_{\text{miss}}^n(h_j) - L_{\text{miss}}(h_j) \right| \right\} \right] \leq \sqrt{n \log(2M)/2},$$

which concludes the proof. ■

2.4 Classification based on convexification

Nonparametric classification based on the empirical risk minimization may seem appealing since its statistical properties, such as oracle inequalities, can be obtained simply. However, these properties cannot be used to derive efficient practical classifiers due to the computational cost of the optimization problem defined by (2.10). One of the most popular approach to design efficient algorithm for classification follows from a convexification of the original problem (2.10). The target loss function $\widehat{L}_{\text{miss}}^n$ is replaced by a convex surrogate and its minimization is constrained to a convex set of classifiers.

For any convex function $f : \mathcal{X} \rightarrow \mathbb{R}$, it is possible to build a classifier h given by $h_f = \text{sign}(f)$. The associated empirical classification is then

$$\widehat{L}_{\text{miss}}^n(h_f) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{Y_i \neq h_f(X_i)} = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{Y_i f(X_i) < 0}.$$

Then, replacing the indicator function by any convex loss function ℓ yields a convex surrogate of $\widehat{L}_{\text{miss}}^n$:

$$\widehat{L}_{\text{miss}}^{n,\text{conv}}(f) = \frac{1}{n} \sum_{i=1}^n \ell(Y_i f(X_i)).$$

Assume then that \mathcal{F} is a convex set of functions from \mathcal{X} to \mathbb{R} . An approximate classifier can be efficiently computed by solving the following optimization problem.

$$\widehat{f}_{\mathcal{F}}^n \in \arg \min_{f \in \mathcal{F}} \widehat{L}_{\text{miss}}^{n,\text{conv}}(f). \quad (2.12)$$

In addition, when the smoothness of the function f is penalized, $\widehat{L}_{\text{miss}}^{n,\text{conv}}$ may be replaced by

$$\widehat{L}_{\text{miss}}^{n,\text{conv}}(f) = \frac{1}{n} \sum_{i=1}^n \ell(Y_i f(X_i)) + \lambda \|f\|^2, \quad (2.13)$$

where $\lambda > 0$ and $\|\cdot\|$ is a norm on the space \mathcal{F} . The soft Support Vector Machines algorithm defined by (2.8) fits this framework with the affine base function $f : x \mapsto \langle w ; x \rangle + b$ and ℓ chosen as the hinge loss $\ell : x \mapsto (1 - x)_+$ when the target function is penalized by its margin $\|w\|^2$.

A useful case in practice consists in choosing \mathcal{F} as a Reproducing Kernel Hilbert Space with positive definite reproducing kernel k on $\mathcal{X} \times \mathcal{X}$.

Definition 2.9. A function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is said to be a positive semi-definite kernel if and only if it is symmetric and if for all $n \geq 1$, $(x_1, \dots, x_n) \in \mathcal{X}^n$ and all $(a_1, \dots, a_n) \in \mathbb{R}^n$,

$$\sum_{1 \leq i, j \leq n} a_i a_j k(x_i, x_j) \geq 0.$$

Remark 2.10 The following functions, defined on $\mathbb{R}^d \times \mathbb{R}^d$, are positive semi-definite kernels:

$$k : (x, y) \mapsto x^T y \quad \text{and} \quad k : (x, y) \mapsto \exp(-\|x - y\|^2 / (2\sigma^2)), \quad \sigma > 0.$$

Definition 2.11. Let \mathcal{F} be a Hilbert space of functions $f : \mathcal{X} \rightarrow \mathbb{R}$. A symmetric function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is said to be a reproducing kernel of \mathcal{F} if and only if for all $x \in \mathcal{X}$, $k(x, \cdot) \in \mathcal{F}$ and for all $x \in \mathcal{X}$ and all $f \in \mathcal{F}$, $\langle f ; k(x, \cdot) \rangle = f(x)$. The space \mathcal{F} is said to be a reproducing kernel Hilbert space with kernel k .

A reproducing kernel associated with a reproducing kernel Hilbert space is positive semi-definite since for all $n \geq 1$, $(x_1, \dots, x_n) \in \mathcal{X}^n$ and all $(a_1, \dots, a_n) \in \mathbb{R}^n$,

$$\sum_{1 \leq i, j \leq n} a_i a_j k(x_i, x_j) = \sum_{1 \leq i, j \leq n} a_i a_j \langle k(x_i, \cdot) ; k(x_j, \cdot) \rangle = \left\| \sum_{1 \leq i \leq n} a_i \langle k(x_i, \cdot) \rangle \right\|^2 \geq 0.$$

Remark 2.12 The positive semi-definite kernel $k : (x, y) \mapsto x^T y$ defined on $\mathbb{R}^d \times \mathbb{R}^d$ is a reproducing kernel of the space

$$\mathcal{F} = \left\{ f : \mathbb{R}^d \rightarrow \mathbb{R} ; \exists \omega \in \mathbb{R}^d \forall x \in \mathbb{R}^d, f(x) = \omega^T x \right\},$$

equipped with the inner product defined, for all $(f, g) \in \mathcal{F} \times \mathcal{F}$, by

$$\langle f ; g \rangle = \omega_f^T \omega_g,$$

where $\omega_f, \omega_g \in \mathbb{R}^d$ and $f : x \mapsto \omega_f^T x$, $g : x \mapsto \omega_g^T x$.

Proposition 2.13 proves that the minimization of the penalized empirical loss amounts to solving a convex optimization problem on \mathbb{R}^n for which many efficient numerical solution exist.

Proposition 2.13 Let $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ be a positive definite kernel and \mathcal{F} the RKHS with kernel k . Then,

$$\widehat{f}_{\mathcal{F}}^n \in \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n \ell(Y_i f(X_i)) + \lambda \|f\|_{\mathcal{F}}^2,$$

where $\|f\|_{\mathcal{F}}^2 = \langle f; f \rangle$, is given by $\hat{f}_{\mathcal{F}}^n : x \mapsto \sum_{i=1}^n \hat{\alpha}_i k(X_i, x)$, where

$$\hat{\alpha} \in \arg \min_{\alpha \in \mathbb{R}^n} \left\{ \frac{1}{n} \sum_{i=1}^n \ell \left(\sum_{j=1}^n \alpha_j Y_j k(X_j, X_i) \right) + \lambda \sum_{1 \leq i, j \leq n} \alpha_i \alpha_j k(X_i, X_j) \right\}.$$

PROOF. Let V be the linear space spanned by $(k(X_i, \cdot))_{1 \leq i \leq n}$. For all $f \in \mathcal{F}$, f can be written $f = f_V + f_{V^\perp}$ with $f_V \in V$ and $f_{V^\perp} \in V^\perp$. Since \mathcal{F} is a RKHS with kernel k , for all $1 \leq i \leq n$,

$$f_{V^\perp}(X_i) = 0 \quad \text{and} \quad f(X_i) = \langle f; k(X_i, \cdot) \rangle = f_V(X_i).$$

Therefore,

$$\frac{1}{n} \sum_{i=1}^n \ell(Y_i f(X_i)) + \lambda \|f\|^2 = \frac{1}{n} \sum_{i=1}^n \ell(Y_i f_V(X_i)) + \lambda \|f_V\|^2 + \lambda \|f_{V^\perp}\|^2$$

and any minimizer of the target function is in V . There exist $(\alpha_1, \dots, \alpha_n) \in \mathbb{R}^n$ such that

$$\hat{f}_{\mathcal{F}}^n : x \mapsto \sum_{i=1}^n \alpha_i k(X_i, x),$$

which concludes the proof. \blacksquare

In the special case of the soft Support Vector Machines, the solution to the convex problem is $\hat{f}_{\mathcal{F}}^n : x \mapsto \sum_{i=1}^n \hat{\alpha}_i k(X_i, x)$ where

$$\hat{\alpha} \in \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n (1 - Y_i f(X_i))_+ + \lambda \alpha^T K \alpha,$$

Proposition 2.14 Let $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ be a positive definite kernel and \mathcal{F} the RKHS with kernel k . Then, a solution to

$$\hat{f}_{\mathcal{F}}^n \in \arg \min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^n (1 - Y_i f(X_i))_+ + \lambda \|f\|^2,$$

is given by $\hat{f}_{\mathcal{F}}^n : x \mapsto \sum_{i=1}^n \hat{\alpha}_i k(X_i, x)$, where, for all $1 \leq i \leq n$,

$$\begin{aligned} \hat{\alpha}_i &= 0 \quad \text{if} \quad Y_i \hat{f}_{\mathcal{F}}^n(X_i) > 1, \\ \hat{\alpha}_i &= Y_i / (2\lambda n) \quad \text{if} \quad Y_i \hat{f}_{\mathcal{F}}^n(X_i) < 1, \\ 0 &\leq Y_i \hat{\alpha}_i \leq 1 / (2\lambda n) \quad \text{if} \quad Y_i \hat{f}_{\mathcal{F}}^n(X_i) = 1. \end{aligned}$$

PROOF. By Proposition 2.13, $\hat{f}_{\mathcal{F}}^n : x \mapsto \sum_{i=1}^n \hat{\alpha}_i k(X_i, x)$ where

$$\hat{\alpha} \in \arg \min_{\alpha \in \mathbb{R}^n} \left\{ \frac{1}{n} \sum_{i=1}^n \left(1 - \sum_{j=1}^n \alpha_j Y_j k(X_j, X_i) \right)_+ + \lambda \sum_{1 \leq i, j \leq n} \alpha_i \alpha_j k(X_i, X_j) \right\}. \quad (2.14)$$

Any vector $\alpha \in \mathbb{R}^n$ solution to (2.14) coincides with a vector β solution to

$$(\hat{\beta}, \hat{\xi}) \in \arg \min_{\substack{(\beta, \xi) \in \mathbb{R}^{2n} \\ \xi_i \geq 0 \vee (1 - Y_i(K\beta)_i)}} \left\{ \frac{1}{n} \sum_{i=1}^n \xi_i + \lambda \beta^T K \beta \right\}, \quad (2.15)$$

where K is the $\mathbb{R}^{n \times n}$ matrix such that for all $1 \leq i, j \leq n$, $K_{i,j} = k(X_i, X_j)$. The optimization problem (2.15) is equivalent to

$$(\hat{\beta}, \hat{\xi}) \in \arg \min_{\substack{h_i(\xi, \beta) \leq 0, \\ g_i(\xi, \beta) \leq 0}} f(\xi, \beta), \quad (2.16)$$

where $f : (\xi, \beta) \rightarrow n^{-1} \sum_{i=1}^n \xi_i + \lambda \beta^T K \beta$ and for all $1 \leq i \leq n$, $g_i : (\xi, \beta) \mapsto -\xi_i$ and $h_i : (\xi, \beta) \mapsto 1 - \xi_i - Y_i(K\beta)_i$. As the function to be minimized and the inequality constraints are convex and continuously differentiable, $(\beta_*, \xi_*) \in \mathbb{R}^{2d}$ is solution to (2.15) if and only if it satisfies the Karush–Kuhn–Tucker conditions: there exists $(\mu, \gamma) \in \mathbb{R}^{2d}$ such that the following conditions hold.

i) **Stationarity**: for all $1 \leq i \leq n$,

$$\begin{aligned} \partial_{\xi_i} \left\{ f(\xi_*, \beta_*) + \sum_{i=1}^n \mu_i h_i(\xi_*, \beta_*) + \sum_{i=1}^n \gamma_i g_i(\xi_*, \beta_*) \right\} &= 0, \\ \partial_{\beta_i} \left\{ f(\xi_*, \beta_*) + \sum_{i=1}^n \mu_i h_i(\xi_*, \beta_*) + \sum_{i=1}^n \gamma_i g_i(\xi_*, \beta_*) \right\} &= 0, \end{aligned}$$

so that $2\lambda(K\beta_*)_i = (K(\mu \cdot Y))_i$ and $\mu_i + \gamma_i = 1/n$.

- ii) **Feasibility**: for all $1 \leq i \leq n$, $\xi_{*,i} \geq 0$, $\xi_{*,i} \geq 1 - Y_i(K\beta_*)_i$, $\mu_i \geq 0$ and $\gamma_i \geq 0$.
- iii) **Slackness**: for all $1 \leq i \leq n$, $\mu_i(\xi_{*,i} - 1 + Y_i(K\beta_*)_i) = 0$ and $\gamma_i \xi_{*,i} = 0$.

The stationarity condition holds if for all $1 \leq i \leq n$, $\beta_{*,i} = \mu_i Y_i / (2\lambda)$. If $Y_i(K\beta_*)_i > 1$, then $\mu_i = 0$ and $\beta_{*,i} = 0$. If $Y_i(K\beta_*)_i < 1$ then $\xi_{*,i} > 0$ and $\gamma_i = 0$. This yields $\mu_i = 1/n$ and then $\beta_{*,i} = Y_i / (2\lambda n)$. If $Y_i(K\beta_*)_i = 1$, $0 \leq \beta_{*,i} Y_i = \mu_i / (2\lambda) \leq 1 / (2\lambda n)$. ■

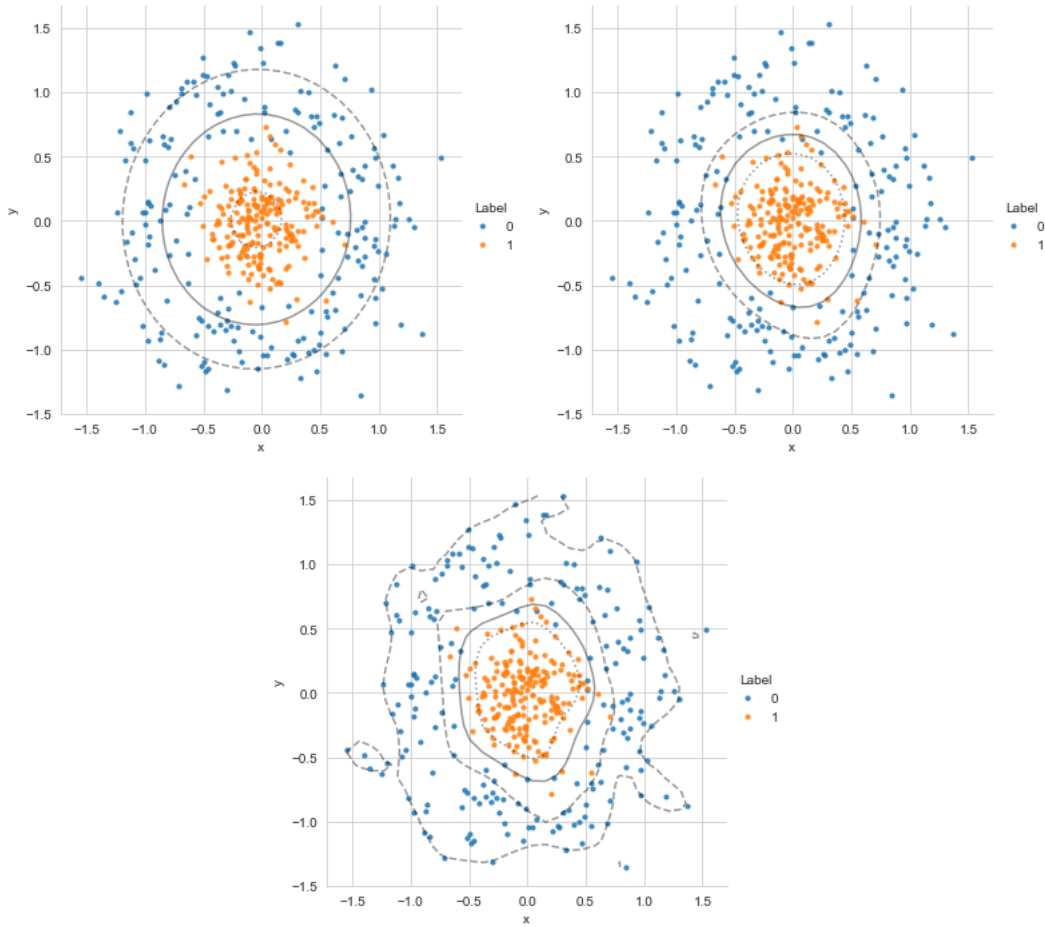


Fig. 2.10 (Top left) kernel based SVM classifier and its margin (dashed lines) γ_0 with kernel $k : (x, x') \mapsto e^{-\|x-x'\|^2/\gamma}$. (Top right) kernel based SVM classifier with $\gamma_1 > \gamma_0$ and (bottom) kernel based SVM classifier with $\gamma_2 > \gamma_1$.

Chapter 3

Stochastic gradient descent

Contents

3.1	Gentle introduction, minimization of a convex Lipschitz function on \mathbb{R}	31
3.2	Gradient descent on \mathbb{R}^d	34
3.3	Gradient descent projected on a bounded convex subset of \mathbb{R}^d	36
3.4	Stochastic gradient descent algorithm	38
3.5	Optimization methods for neural networks	40
3.5.1	AdaGrad	40
3.5.2	AdaDelta	40
3.5.3	RMSprop optimizer	40
3.5.4	ADAM: Adaptive moment estimation	40

Keywords 3.1

Supervised learning applications and nonparametric regression settings are usually based on the minimization of an objective function on \mathbb{R}^d , see for instance Proposition 2.13 for kernel based SVM algorithms, Exercise 6.9 for penalized kernel based regression or (2.4) for the maximum likelihood estimation of the parameters of a logistic regression application. In these cases, the optimization problem to be solved may be written

$$\hat{\theta}_n \in \arg \min_{\theta \in \mathbb{R}^d} \left\{ \frac{1}{n} \sum_{i=1}^n \ell(Y_i, \theta' \Phi(X_i)) + \lambda \omega(\theta) \right\}, \quad (3.1)$$

where $\{(X_i, Y_i)\}_{1 \leq i \leq n}$ are i.i.d., ℓ is a loss function, Φ is a known function, $\lambda > 0$ and ω is a penalization function. In this framework, the first term is the empirical loss function associated with the expected risk

$$R(\theta) = \mathbb{E}[\ell(Y, \theta' \Phi(X))],$$

which is usually not available explicitly. This chapter introduces widely used gradient descent algorithms to minimize (3.1). In the following sections, gradient descent algorithms are used to obtain a sequence $(x_k)_{k \geq 0}$ to approximate $x_* = \arg \min_{x \in \mathbb{R}} f(x)$ (resp. $x_* = \arg \min_{x \in \mathbb{R}^d} f(x)$) for a convex function $f : \mathbb{R} \rightarrow \mathbb{R}$ (resp. $f : \mathbb{R}^d \rightarrow \mathbb{R}$).

3.1 Gentle introduction, minimization of a convex Lipschitz function on \mathbb{R}

Definition 3.1. Let I be an interval of \mathbb{R} and $f : I \rightarrow \mathbb{R}$. The function f is said to be convex if and only if, for all $(x, y) \in I^2$ and all $\lambda \in [0, 1]$,

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y) .$$

If $-f$ is convex, f is concave. f is said to be **strictly convex** if the inequality is strict when $x \neq y$ and $\lambda \in (0, 1)$.

Proposition 3.2 *Let I be an interval of \mathbb{R} and $f : I \rightarrow \mathbb{R}$. Then, f is convex if and only if for all $x_0 \in I$, the function φ_{x_0} defined on $I \setminus \{x_0\}$ by*

$$\varphi_{x_0} : x \mapsto \frac{f(x) - f(x_0)}{x - x_0}$$

is nondecreasing.

PROOF. Assume that f is convex. Let $x_0 \in I$ and choose $a < b$ in I . Assume for instance that $a < x_0 < b$ (other cases are dealt with similarly). Write

$$x_0 = \frac{b - x_0}{b - a}a + \frac{x_0 - a}{b - a}b$$

so that, by convexity of f ,

$$f(x_0) \leq \frac{b - x_0}{b - a}f(a) + \frac{x_0 - a}{b - a}f(b) .$$

Then, using

$$f(x_0) = \frac{b - x_0}{b - a}f(x_0) + \frac{x_0 - a}{b - a}f(x_0) ,$$

yields

$$\frac{b - x_0}{b - a}(f(a) - f(x_0)) + \frac{x_0 - a}{b - a}(f(b) - f(x_0)) \geq 0$$

and

$$\frac{f(a) - f(x_0)}{x_0 - a} + \frac{f(b) - f(x_0)}{b - x_0} \geq 0 .$$

Therefore, $\varphi_{x_0}(a) \leq \varphi_{x_0}(b)$, which concludes the proof. Assume now that for all $x_0 \in I$, φ_{x_0} is nondecreasing. Let a and b in I be such that $a < b$, choose $\lambda \in (0, 1)$ and write $x_0 = \lambda a + (1 - \lambda)b$. Following the same steps as above yields $f(x_0) \leq \lambda f(a) + (1 - \lambda)f(b)$ and f is convex. ■

Theorem 3.3. *Let I be an interval of \mathbb{R} and $f : I \rightarrow \mathbb{R}$ a differentiable function on I . Then, f is convex if and only if f' is nondecreasing. Equivalently, f is convex if and only if, for all $(x, y) \in \mathbb{R}^2$,*

$$f(y) \geq f(x) + f'(x)(y - x) .$$

PROOF. Assume that f is convex. Then, for all $a < x < b$ in I , by Proposition 3.2,

$$\frac{f(x) - f(a)}{x - a} \leq \frac{f(b) - f(a)}{b - a} \leq \frac{f(b) - f(x)}{b - x} .$$

Taking the limit of the left hand term when x tends to a and of the right hand term when x tends to b yields $f'(a) \leq f'(b)$. f' is therefore nondecreasing. Assume now that f' is nondecreasing and let $a < x_0 < b$ be in I . By the mean value theorem, there exist $\alpha_1 \in (a, x_0)$ and $\alpha_2 \in (x_0, b)$ such that

$$f(x_0) - f(a) = f'(\alpha_1)(x_0 - a) \quad \text{et} \quad f(b) - f(x_0) = f'(\alpha_2)(b - x_0) .$$

As f' is nondecreasing,

$$\frac{f(x_0) - f(a)}{x_0 - a} \leq \frac{f(b) - f(x_0)}{b - x_0}$$

and f is convex by Proposition 3.2. ■

Corollary 3.4 *Let I be an interval of \mathbb{R} and $f : I \rightarrow \mathbb{R}$ be a twice differentiable function on I . Then, f is convex if and only if for all $x \in I$, $f''(x) \geq 0$.*

Consider a function $f : \mathbb{R} \rightarrow \mathbb{R}$ convex, differentiable and such that the derivative of f is Lipschitz: there exists $L \in \mathbb{R}$ such that for all $x \in \mathbb{R}$,

$$|f'(x) - f'(y)| \leq L|x - y|. \quad (3.2)$$

Note first that for all $(x, y) \in \mathbb{R}^2$,

$$f(x) + f'(x)(y - x) \leq f(y) \leq f(x) + f'(x)(y - x) + \frac{L}{2}(x - y)^2.$$

The first inequality is a direct consequence of Theorem 3.3. To prove the second inequality, define, for all $(x, y) \in \mathbb{R}^2$, the function $\varphi_{x,y}$ on $[0, 1]$ by

$$\varphi_{x,y} : h \mapsto f((1 - h)x + hy).$$

This function is differentiable and for all $h \in [0, 1]$,

$$\varphi'_{x,y}(h) = (y - x)f'((1 - h)x + hy).$$

Then,

$$\begin{aligned} f(y) - f(x) &= \varphi_{x,y}(1) - \varphi_{x,y}(0) = \int_0^1 (y - x)f'((1 - h)x + hy)dh, \\ &= \int_0^1 (y - x) \{f'((1 - h)x + hy) - f'(x)\} dh + (y - x)f'(x), \\ &\leq (y - x)f'(x) + (y - x)^2 L \int_0^1 h dh, \\ &\leq (y - x)f'(x) + \frac{L}{2}(y - x)^2. \end{aligned}$$

Define the sequence $(x_n)_{n \geq 0}$ by choosing $x_0 \in \mathbb{R}$ and setting, for all $n \in \mathbb{N}$,

$$x_{n+1} = x_n - \frac{1}{L}f'(x_n).$$

As for all $y \in \mathbb{R}$,

$$\begin{aligned} f(y) &\leq f(x_n) + (y - x_n)f'(x_n) + \frac{L}{2}(y - x_n)^2, \\ &\leq f(x_n) - \frac{1}{2L}|f'(x_n)|^2 + \frac{L}{2} \left| x_n - y - \frac{1}{L}f'(x_n) \right|^2, \end{aligned}$$

the following upper bound holds

$$f(x_{n+1}) \leq f(x_n) - \frac{1}{2L}|f'(x_n)|^2. \quad (3.3)$$

On the other hand, for all $n \in \mathbb{N}$, $f(x_n) + f'(x_n)(x_* - x_n) \leq f(x_*)$, then $f(x_n) - f(x_*) \leq f'(x_n)(x_n - x_*)$ and, for all $n \in \mathbb{N}$ such that $x_n \neq x_*$,

$$|f'(x_n)| \geq \frac{f(x_n) - f(x_*)}{|x_n - x_*|}.$$

Plugging this inequality in (3.3) yields, for all $n \in \mathbb{N}$ such that $x_n \neq x_*$,

$$f(x_{n+1}) - f(x_*) \leq f(x_n) - f(x_*) - \frac{1}{2L} \frac{(f(x_n) - f(x_*))^2}{|x_n - x_*|^2}.$$

Proposition 3.5 For all $n \in \mathbb{N}^*$,

$$f(x_n) - f(x_*) \leq \frac{2L}{n} |x_0 - x_*|^2.$$

PROOF. Note that for all $n \in \mathbb{N}$, by definition of x_{n+1} ,

$$\begin{aligned} |x_{n+1} - x_*|^2 &= |x_n - x_*|^2 - \frac{2}{L} f'(x_n)(x_n - x_*) + \frac{1}{L^2} (f'(x_n))^2, \\ &\leq |x_n - x_*|^2 - \frac{2}{L} (f(x_n) - f(x_*)) + \frac{2}{L} (f(x_n) - f(x_{n+1})), \\ &\leq |x_n - x_*|^2, \end{aligned}$$

by (3.3) and by (3.2), $f'(x_n)(x_n - x_*) \geq f(x_n) - f(x_*)$. The sequence $(|x_n - x_*|)_{n \in \mathbb{N}}$ is then nonincreasing and for all $n \in \mathbb{N}$ such that $x_n \neq x_*$,

$$f(x_{n+1}) - f(x_*) \leq f(x_n) - f(x_*) - \frac{1}{2L} \frac{(f(x_n) - f(x_*))^2}{|x_0 - x_*|^2}.$$

Using that for all x , $1 - x \leq (1 + x)^{-1}$,

$$\begin{aligned} f(x_{n+1}) - f(x_*) &\leq \{f(x_n) - f(x_*)\} \left(1 - \frac{f(x_n) - f(x_*)}{2L|x_0 - x_*|^2}\right), \\ &\leq \{f(x_n) - f(x_*)\} \left(1 + \frac{f(x_n) - f(x_*)}{2L|x_0 - x_*|^2}\right)^{-1}. \end{aligned}$$

Writing, for all $n \in \mathbb{N}$, $\alpha_n = (f(x_{n+1}) - f(x_*))^{-1}$,

$$\alpha_n \geq \alpha_{n-1} + \frac{1}{2L|x_0 - x_*|^2} \geq \dots \geq \alpha_0 + \frac{n}{2L|x_0 - x_*|^2} \geq \frac{n}{2L|x_0 - x_*|^2}.$$

Then, for all $n \in \mathbb{N}$,

$$f(x_{n+1}) - f(x_*) \leq \frac{2L|x_0 - x_*|^2}{n}.$$

■

3.2 Gradient descent on \mathbb{R}^d

In the following, f is assumed to be a convex differentiable function on \mathbb{R}^d .

Proposition 3.6 (L-smooth function) Assume that $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is a convex differentiable function such that ∇f is L-Lipschitz: for all $(x, y) \in \mathbb{R}^d \times \mathbb{R}^d$,

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|.$$

Then, for all $(x, y) \in \mathbb{R}^d \times \mathbb{R}^d$,

i)

$$f(x) + \langle \nabla f(x), y - x \rangle \leq f(y) \leq f(x) + \langle \nabla f(x), y - x \rangle + \frac{L}{2} \|y - x\|^2 .$$

ii)

$$f(y) - f(x) \leq \langle \nabla f(y), y - x \rangle - \frac{1}{2L} \|\nabla f(y) - \nabla f(x)\|^2 .$$

iii)

$$\langle \nabla f(y) - \nabla f(x), y - x \rangle \geq \frac{1}{L} \|\nabla f(y) - \nabla f(x)\|^2 .$$

PROOF.

i) For all $(x, h) \in \mathbb{R}^d \times \mathbb{R}^d$ the function $\varphi_{x,h}$ defined on $[0, 1]$ by

$$\varphi_{x,h} : t \mapsto f(x + th)$$

is differentiable and convex. Then, $\varphi'_{x,h}(0) \leq \varphi_{x,h}(1) - \varphi_{x,h}(0)$. As for all $t \in [0, 1]$, $\varphi'_{x,h}(t) = \langle \nabla f(x + th), h \rangle$,

$$f(y) - f(x) \geq \langle \nabla f(x), y - x \rangle . \quad (3.4)$$

The other inequality follows exactly the same steps as the proof for convex functions defined on \mathbb{R} .ii) For all x, y, z in \mathbb{R}^d , by convexity and L -smoothness,

$$f(y) - f(x) \leq \langle \nabla f(y), y - z \rangle + \langle \nabla f(x), z - x \rangle + \frac{L}{2} \|z - x\|^2 .$$

Then, choosing $z = x - (\nabla f(x) - \nabla f(y))/L$ yields

$$\begin{aligned} f(y) - f(x) &\leq \langle \nabla f(y), y - z \rangle + \langle \nabla f(x), z - x \rangle + \frac{L}{2} \|z - x\|^2 , \\ &\leq \langle \nabla f(y), y - x \rangle + L^{-1} \langle \nabla f(y), \nabla f(x) - \nabla f(y) \rangle - L^{-1} \langle \nabla f(x), \nabla f(x) - \nabla f(y) \rangle + \frac{1}{2L} \|\nabla f(x) - \nabla f(y)\|^2 , \\ &\leq \langle \nabla f(y), y - x \rangle - \frac{1}{2L} \|\nabla f(x) - \nabla f(y)\|^2 , \end{aligned}$$

which concludes the proof.

iii) By ii), for all x, y in \mathbb{R}^d ,

$$f(y) - f(x) \leq \langle \nabla f(y), y - x \rangle - \frac{1}{2L} \|\nabla f(x) - \nabla f(y)\|^2$$

and

$$f(x) - f(y) \leq \langle \nabla f(x), x - y \rangle - \frac{1}{2L} \|\nabla f(x) - \nabla f(y)\|^2 .$$

The proof is completed by summing these two inequalities. ■Gradient descent algorithm on \mathbb{R}^d follows the same steps as in the scalar case.Choosing $x_0 \in \mathbb{R}^d$ and $\eta > 0$, define, for all $n \in \mathbb{N}$,

$$x_{n+1} = x_n - \eta \nabla f(x_n) .$$

Proposition 3.7 Let f be a convex differentiable function on \mathbb{R}^d such that ∇f is L -Lipschitz. Assume that $\eta = 1/L$, then for all $n \in \mathbb{N}^*$,

$$f(x_n) - f(x_*) \leq \frac{2L \|x_0 - x_*\|^2}{n + 4} .$$

PROOF. First, for all $n \geq 1$, by Proposition 3.6, if $\eta \leq 2/L$,

$$\begin{aligned} \|x_n - x_*\|^2 &= \|x_{n-1} - x_* - \eta \nabla f(x_{n-1})\|^2 \leq \|x_{n-1} - x_*\|^2 + \eta^2 \|\nabla f(x_{n-1})\|^2 - 2\eta \langle x_{n-1} - x_*, \nabla f(x_{n-1}) \rangle, \\ &\leq \|x_{n-1} - x_*\|^2 + \eta^2 \|\nabla f(x_{n-1})\|^2 - 2\frac{\eta}{L} \|\nabla f(x_{n-1})\|^2, \\ &\leq \|x_{n-1} - x_*\|^2, \\ &\leq \|x_0 - x_*\|^2. \end{aligned}$$

On the other hand, by L -smoothness,

$$f(x_n) - f(x_{n-1}) \leq \langle \nabla f(x_{n-1}), -\eta \nabla f(x_{n-1}) \rangle + \frac{L\eta^2}{2} \|\nabla f(x_{n-1})\|^2 \leq -\frac{1}{2L} \|\nabla f(x_{n-1})\|^2.$$

Then, by convexity and the Cauchy-Schwarz inequality,

$$f(x_{n-1}) - f(x_*) \leq \langle \nabla f(x_{n-1}), x_{n-1} - x_* \rangle \leq \|\nabla f(x_{n-1})\| \cdot \|x_{n-1} - x_*\|.$$

This yields,

$$\begin{aligned} f(x_n) - f(x_*) &\leq f(x_{n-1}) - f(x_*) - \frac{1}{2L} \|\nabla f(x_{n-1})\|^2 \leq f(x_{n-1}) - f(x_*) - \frac{(f(x_{n-1}) - f(x_*))^2}{2L\|x_{n-1} - x_*\|^2}, \\ &\leq f(x_{n-1}) - f(x_*) - \frac{(f(x_{n-1}) - f(x_*))^2}{2L\|x_0 - x_*\|^2}. \end{aligned}$$

Hence,

$$\frac{1}{f(x_{n-1}) - f(x_*)} \leq \frac{1}{f(x_n) - f(x_*)} - \frac{f(x_{n-1}) - f(x_*)}{2L\|x_0 - x_*\|^2 (f(x_n) - f(x_*))} \leq \frac{1}{f(x_n) - f(x_*)} - \frac{1}{2L\|x_0 - x_*\|^2}$$

so that

$$\frac{1}{f(x_{n-1}) - f(x_*)} \geq \frac{n}{2L\|x_0 - x_*\|^2} + \frac{1}{f(x_0) - f(x_*)}$$

and

$$f(x_{n-1}) - f(x_*) \leq \frac{f(x_0) - f(x_*)}{1 + \frac{n(f(x_0) - f(x_*))}{2L\|x_0 - x_*\|^2}}.$$

The proof is completed by $f(x_0) - f(x_*) \leq L\|x_0 - x_*\|^2/2$. ■

3.3 Gradient descent projected on a bounded convex subset of \mathbb{R}^d

Let \mathcal{C} be a convex set of \mathbb{R}^d and $\Pi_{\mathcal{C}}$ the orthogonal projection on \mathcal{C} : for all $x \in \mathbb{R}^d$,

$$\Pi_{\mathcal{C}}(x) = \arg \min_{u \in \mathcal{C}} \|u - x\|^2.$$

\mathcal{C} is assumed to be bounded, so that there exists $r > 0$ such that $\mathcal{C} \subset \mathcal{B}(0, r)$. In the following, f is assumed to be a convex differentiable function on \mathbb{R}^d .

Choosing $x_0 \in \mathcal{C}$ and $\eta > 0$, define, for all $n \in \mathbb{N}$,

$$y_{n+1} = x_n - \eta \nabla f(x_n) \quad \text{and} \quad x_{n+1} = \Pi_{\mathcal{C}}(y_{n+1}).$$

Note that for all $c \in \mathcal{C}$ and all $x \in \mathbb{R}^d$, $\|x - c\| \geq \|x - \Pi_{\mathcal{C}}(x)\|$ and for all $\lambda \in [0, 1]$, since $\lambda c + (1 - \lambda)\Pi_{\mathcal{C}}(x) \in \mathcal{C}$,

$$\|x - (\lambda c + (1 - \lambda)\Pi_{\mathcal{C}}(x))\|^2 \geq \|x - \Pi_{\mathcal{C}}(x)\|^2 \quad (3.5)$$

and

$$\lambda \|c - \Pi_{\mathcal{C}}(x)\|^2 \geq 2 \langle x - \Pi_{\mathcal{C}}(x), c - \Pi_{\mathcal{C}}(x) \rangle . \quad (3.6)$$

Note that

$$\|x - \Pi_{\mathcal{C}}(x) - \lambda (c - \Pi_{\mathcal{C}}(x))\|^2 = \|x - \Pi_{\mathcal{C}}(x)\|^2 + \lambda^2 \|c - \Pi_{\mathcal{C}}(x)\|^2 - 2\lambda \langle x - \Pi_{\mathcal{C}}(x), (c - \Pi_{\mathcal{C}}(x)) \rangle .$$

By (3.6), $\langle x - \Pi_{\mathcal{C}}(x), c - \Pi_{\mathcal{C}}(x) \rangle \leq 0$ and

$$\begin{aligned} \|x - \Pi_{\mathcal{C}}(x)\|^2 + \|c - \Pi_{\mathcal{C}}(x)\|^2 &= \|x - \Pi_{\mathcal{C}}(x) + \Pi_{\mathcal{C}}(x) - c\|^2 - 2 \langle \Pi_{\mathcal{C}}(x) - x, c - \Pi_{\mathcal{C}}(x) \rangle , \\ &= \|x - c\|^2 + 2 \langle c - \Pi_{\mathcal{C}}(x), x - \Pi_{\mathcal{C}}(x) \rangle , \\ &\leq \|x - c\|^2 . \end{aligned} \quad (3.7)$$

Proposition 3.8 *Let f be a convex differentiable function on \mathbb{R}^d such that ∇f is L -Lipschitz. Then, for all $n \in \mathbb{N}^*$, choosing $\eta = r/(L\sqrt{n})$ yields*

$$f\left(\frac{1}{n} \sum_{k=1}^n x_k\right) - f(x_*) \leq \frac{rL}{\sqrt{n}} .$$

PROOF. By Proposition 3.6, for all $k \in \mathbb{N}$,

$$f(x_*) - f(x_k) \geq \langle \nabla f(x_k), x_* - x_k \rangle .$$

As $\nabla f(x_k) = \eta^{-1}(x_k - y_{k+1})$ and

$$\begin{aligned} f(x_k) - f(x_*) &\leq \eta^{-1} \langle x_k - y_{k+1}, x_k - x_* \rangle \leq \eta^{-1} \langle x_k - y_{k+1}, x_n - y_{k+1} \rangle + \eta^{-1} \langle x_k - y_{k+1}, y_{k+1} - x_* \rangle , \\ &\leq \eta \|\nabla f(x_k)\|^2 + \frac{1}{2\eta} (\|x_k - x_*\|^2 - \|x_k - y_{k+1}\|^2 - \|y_{k+1} - x_*\|^2) , \\ &\leq \frac{\eta}{2} \|\nabla f(x_k)\|^2 + \frac{1}{2\eta} (\|x_k - x_*\|^2 - \|y_{k+1} - x_*\|^2) . \end{aligned}$$

This yields

$$\begin{aligned} \frac{1}{n} \sum_{k=1}^n f(x_k) - f(x_*) &\leq \frac{\eta L^2}{2} + \frac{1}{2\eta n} \sum_{k=1}^n (\|x_k - x_*\|^2 - \|y_{k+1} - x_*\|^2) , \\ &\leq \frac{\eta L^2}{2} + \frac{1}{2\eta n} \sum_{k=2}^n (\|x_k - x_*\|^2 - \|y_k - x_*\|^2) + \frac{1}{2\eta n} (\|x_1 - x_*\|^2 - \|y_{n+1} - x_*\|^2) . \end{aligned}$$

Since $\pi_{\mathcal{C}}(y_{k+1}) = x_{k+1}$, by Proposition 3.6,

$$\|\pi_{\mathcal{C}}(y_{k+1}) - y_{k+1}\|^2 + \|\pi_{\mathcal{C}}(y_{k+1}) - x_*\|^2 \leq \|y_{k+1} - x_*\|^2$$

and

$$\|x_{k+1} - x_*\|^2 - \|y_{k+1} - x_*\|^2 \leq -\|\pi_{\mathcal{C}}(y_{k+1}) - y_{k+1}\|^2 \leq 0 .$$

Then,

$$\frac{1}{n} \sum_{k=1}^n f(x_k) - f(x_*) \leq \frac{\eta L^2}{2} + \frac{\|x_1 - x_*\|^2}{2\eta n} ,$$

which concludes the proof by convexity of f choosing $\eta = r/(L\sqrt{n})$. ■

Definition 3.9. Assume that $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is a convex differentiable function. f is said to be α -strongly convex if and only if for all $(x, y) \in \mathbb{R}^d \times \mathbb{R}^d$,

$$f(y) \leq f(x) + \langle \nabla f(y), y - x \rangle - \frac{\alpha}{2} \|y - x\|^2 .$$

Proposition 3.10 *Let f be a convex differentiable function on \mathbb{R}^d such that ∇f is L -Lipschitz. Assume that f is α -strongly convex. Then, for all $n \in \mathbb{N}^*$, choosing $\eta = 1/L$ yields*

$$\|x_{n+1} - x_*\|^2 \leq e^{-\alpha n/L} \|x_1 - x_*\|^2.$$

PROOF. For all $n \geq 0$,

$$\|x_{n+1} - x_*\|^2 = \|x_n - \eta \nabla f(x_n) - x_*\|^2 = \|x_n - x_*\|^2 + \eta^2 \|\nabla f(x_n)\|^2 - 2\eta \langle x_n - x_*, \nabla f(x_n) \rangle$$

By strong convexity,

$$f(x_{n+1}) - f(x_*) \leq \langle \nabla f(x_n), x_n - x_* \rangle - \frac{1}{2L} \|\nabla f(x_n)\|^2 - \frac{\alpha}{2} \|x_n - x_*\|^2,$$

which yields

$$-2\eta \langle x_n - x_*, \nabla f(x_n) \rangle \leq -\eta^2 \|\nabla f(x_n)\|^2 - \alpha \eta \|x_n - x_*\|^2.$$

Therefore,

$$\|x_{n+1} - x_*\|^2 \leq (1 - \alpha \eta) \|x_n - x_*\|^2 \leq (1 - \alpha \eta)^n \|x_1 - x_*\|^2,$$

which concludes the proof. ■

Proposition 3.11 *Let f be a convex differentiable function on \mathbb{R}^d such that ∇f is L -Lipschitz. Assume that f is α -strongly convex. Then, for all $n \in \mathbb{N}^*$, choosing $\eta = 2/(\alpha + L)$ yields*

$$f(x_{n+1}) - f(x_*) \leq \frac{L}{2} e^{-4n/(L/\alpha + 1)} \|x_1 - x_*\|^2.$$

PROOF. Note first that the function $x \mapsto f(x) - \alpha \|x\|^2/2$ is convex as f is α -strongly convex. On the other hand,

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \frac{1}{L - \alpha} \|\nabla f(x) - \nabla f(y)\|^2.$$

As $\nabla f(x_*) = 0$,

$$f(x_n) - f(x_*) \leq \frac{L}{2} \|x_n - x_*\|^2.$$

For all $n \geq 0$,

$$\|x_{n+1} - x_*\|^2 = \|x_n - \eta \nabla f(x_n) - x_*\|^2 = \|x_n - x_*\|^2 + \eta^2 \|\nabla f(x_n)\|^2 - 2\eta \langle x_n - x_*, \nabla f(x_n) \rangle$$

Therefore,

$$\|x_{n+1} - x_*\|^2 \leq \|x_n - \eta \nabla f(x_n) - x_*\|^2 = \|x_n - x_*\|^2 + \eta^2 \|\nabla f(x_n)\|^2 - 2\eta \langle x_n - x_*, \nabla f(x_n) \rangle$$

and

$$\|x_{n+1} - x_*\|^2 \leq (1 - \alpha \eta) \|x_n - x_*\|^2 \leq (1 - \alpha \eta)^n \|x_1 - x_*\|^2,$$

which concludes the proof. ■

3.4 Stochastic gradient descent algorithm

The previous algorithm allows to obtain a sequence $(x_k)_{k \geq 0}$ to approximate $x_* = \arg \min_{x \in \mathbb{R}^d} f(x)$. At each time step $k \geq 0$, this gradient descent algorithm requires to compute $\nabla f(x_k)$ which may be computationally prohibitive when f is a function based on all the available data (as in Proposition 2.13, Exercise 6.9 or

(2.4)). Stochastic gradient descent algorithms offer an appealing alternative by computing an expectedly less costly unbiased estimate of this gradient.

Let $x_0 \in \mathbb{R}^d$ and $\eta > 0$, define, for all $k \in \mathbb{N}$,

$$x_{k+1} = x_k - \eta v_k,$$

where v_k is an unbiased estimate of $\nabla f(x_k)$. The $(v_k)_{k \geq 0}$ are assumed to be independent and such that

$$\text{Trace}(\mathbb{V}[v_k]) = \sigma_k^2.$$

Proposition 3.12 *Let f be a convex differentiable function defined on \mathbb{R}^d such that $\|\nabla f\|$ is L -Lipschitz. Assume that there exists σ^2 such that for all $k \geq 0$, $\sigma_k^2 \leq \sigma^2$ and that $\eta L \leq 1$. Then, for all $n \geq 1$,*

$$\mathbb{E} \left[f \left(\frac{1}{n} \sum_{k=1}^n x_k \right) \right] - f(x_*) \leq \frac{1}{2\eta n} \|x_0 - x_*\|^2 + \eta \sigma^2.$$

Therefore, choosing

$$\eta_n = \frac{\|x_0 - x_*\|}{\sigma \sqrt{2n}}$$

yields, for n sufficiently large,

$$\mathbb{E} \left[f \left(\frac{1}{n} \sum_{k=1}^n x_k \right) \right] - f(x_*) \leq \frac{\sqrt{2}\sigma \|x_0 - x_*\|}{\sqrt{n}}.$$

PROOF. By Proposition 3.6,

$$f(x_k) + \langle \nabla f(x_k), x_{k+1} - x_k \rangle \leq f(x_{k+1}) \leq f(x_k) + \langle \nabla f(x_k), x_{k+1} - x_k \rangle + \frac{L}{2} \|x_{k+1} - x_k\|^2$$

which yields

$$f(x_{k+1}) \leq f(x_k) - \eta \langle \nabla f(x_k), v_k \rangle + \frac{L\eta^2}{2} \|v_k\|^2.$$

Therefore, if \mathcal{F}_k denotes the σ -algebra generated by (v_0, \dots, v_{k-1}) ,

$$\mathbb{E}[f(x_{k+1}) | \mathcal{F}_k] \leq f(x_k) - \eta \|\nabla f(x_k)\|^2 + \frac{L\eta^2}{2} (\|\nabla f(x_k)\|^2 + \sigma_k^2)$$

and

$$\mathbb{E}[f(x_{k+1}) | \mathcal{F}_k] \leq f(x_k) - \eta \left(1 - \frac{L\eta}{2} \right) \|\nabla f(x_k)\|^2 + \frac{L\eta^2}{2} \sigma_k^2.$$

Using the fact that $L\eta \leq 1$,

$$\mathbb{E}[f(x_{k+1}) | \mathcal{F}_k] \leq f(x_k) - \frac{\eta}{2} \|\nabla f(x_k)\|^2 + \frac{\eta}{2} \sigma_k^2.$$

and, combined with Proposition 3.6,

$$\mathbb{E}[f(x_{k+1}) | \mathcal{F}_k] \leq f(x_*) + \langle \nabla f(x_k), x_k - x_* \rangle - \frac{\eta}{2} \|\nabla f(x_k)\|^2 + \frac{\eta}{2} \sigma_k^2.$$

Then,

$$\mathbb{E}[f(x_{k+1}) | \mathcal{F}_k] \leq f(x_*) + \mathbb{E} \left[\langle v_k, x_k - x_* \rangle - \frac{\eta}{2} \|v_k\|^2 | \mathcal{F}_k \right] + \eta \sigma_k^2$$

which yields

$$\begin{aligned}\mathbb{E}[f(x_{k+1})|\mathcal{F}_k] &\leq f(x_*) + \frac{1}{2\eta} \mathbb{E}[\|x_k - x_*\|^2 - \|x_k - x_* - \eta v_k\|^2 | \mathcal{F}_k] + \eta \sigma_k^2, \\ &\leq f(x_*) + \frac{1}{2\eta} \mathbb{E}[\|x_k - x_*\|^2 - \|x_{k+1} - x_*\|^2 | \mathcal{F}_k] + \eta \sigma_k^2\end{aligned}$$

and

$$\frac{1}{n} \sum_{k=0}^{n-1} \mathbb{E}[f(x_{k+1})] - f(x_*) \leq \frac{1}{2\eta n} \mathbb{E}[\|x_0 - x_*\|^2 - \|x_n - x_* - \eta v_n\|^2] + \eta \sigma^2.$$

By convexity,

$$\mathbb{E} \left[f \left(\frac{1}{n} \sum_{k=0}^{n-1} x_{k+1} \right) \right] - f(x_*) \leq \frac{1}{2\eta n} \|x_0 - x_*\|^2 + \eta \sigma^2,$$

which concludes the proof. ■

3.5 Optimization methods for neural networks

3.5.1 AdaGrad

The ADAPtive GRADient algorithm introduced by (Duchi et al. 2011) starts from $w^{(0)}$ and uses a learning rate $\eta > 0$ and a momentum α and defines, for all $k \geq 0$ and all $j \in \{1, \dots, d\}$,

$$w_j^{(k+1)} \leftarrow w_j^{(k)} - \frac{\eta}{\sqrt{\sum_{\tau=1}^k (\nabla f(w^{(\tau)}))_j^2}} (\nabla f(w^{(k)}))_j.$$

The rationale of this method is that different rates are used for all coordinates which is crucial for neural networks in which gradient at different layers can be of different order of magnitude. It is proved in (Ward et al., 2018) that AdaGrad achieves the same convergence rate as gradient descent with optimal fixed stepsize up to a log factor. The adaptive step size grows with the inverse of the gradient magnitudes, so that large gradients have small learning rates and small gradients have large learning rates.

3.5.2 AdaDelta

Introduced in (Zeiler, 2012), was introduced to reduce the sensitivity to initial conditions of AdaGrad. Indeed, if the initial gradients are large, the learning rates of AdaGrad will be low for all updates which can be overcome by increasing η , but making the AdaGrad method highly sensitive to the choice of η .

3.5.3 RMSprop optimizer

Unpublished method, from the course of Geoff Hinton.

3.5.4 ADAM: Adaptive moment estimation

Introduced in (Kingma et al., 2014) and considered as the state of the art to optimize neural networks, the ADAM procedure update the parameter estimate as follows. Starting from $m_0 = 0$ and $v_0 = 0$ and choosing $\beta_1, \beta_2, \eta, \epsilon \in (0, 1)$, compute first and second moment estimate

$$m_k = \beta_1 m_{k-1} + (1 - \beta_1) \nabla f(w^{(k)}) \quad \text{and} \quad v_k = \beta_2 v_{k-1} + (1 - \beta_2) (\nabla f(w^{(k)}))^2,$$

then, compute the correction terms

$$\hat{m}_k = \frac{m_k}{1 - \beta_1^k} \quad \hat{v}_k = \frac{v_k}{1 - \beta_2^k},$$

and update the parameter estimate with

$$w^{(k+1)} = w^{(k)} - \frac{\eta}{\sqrt{\hat{v}_k} + \epsilon} \hat{m}_k.$$

First convergence results can be found in (Kingma et al., 2014) and examples where ADAM algorithm does not converge to the optimum are given in (Reddi et al., 2018). Recent analysis by (Barakat et al., 2018).

Chapter 4

Multivariate regression

Contents

4.1	Gaussian vectors	43
4.2	Full rank multivariate regression	44
4.2.1	Least squares estimator	44
4.2.2	Confidence intervals and tests	46
4.3	Introduction to regularized multivariate regression	47
4.3.1	Ridge regression	47
4.3.2	Lasso regression	48
4.3.3	Nonparametric regression	49

Keywords 4.1

4.1 Gaussian vectors

Definition 4.1. A random variable $X \in \mathbb{R}^n$ is a Gaussian vector if and only if, for all $a \in \mathbb{R}^n$, the random variable $\langle a; X \rangle$ is a Gaussian random variable.

For all random variable $X \in \mathbb{R}^n$, $X \sim \mathcal{N}(\mu, \Sigma)$ means that X is a Gaussian vector with mean $\mathbb{E}[X] = \mu \in \mathbb{R}^n$ and covariance matrix $\mathbb{V}[X] = \Sigma \in \mathbb{R}^{n \times n}$. The characteristic function of X is given (see exercises), for all $t \in \mathbb{R}^n$, by

$$\mathbb{E}[e^{i\langle t; X \rangle}] = e^{i\langle t; \mu \rangle - t' \Sigma t / 2}.$$

Therefore, the law of a Gaussian vector is uniquely defined by its mean vector and its covariance matrix. If the covariance matrix Σ is nonsingular, then the law of X has a probability density with respect to the Lebesgue measure on \mathbb{R}^n given by :

$$x \mapsto \det(2\pi\Sigma)^{-1/2} \exp \left\{ -(x - \mu)' \Sigma^{-1} (x - \mu) / 2 \right\},$$

where $\mu = \mathbb{E}[X]$.

Proposition 4.2 Let $X \in \mathbb{R}^n$ be a Gaussian vector. Let $\{i_1, \dots, i_d\}$ be a subset of $\{1, \dots, n\}$, $d \geq 1$. If for all $1 \leq k \neq j \leq d$, $\text{Cov}(X_{i_k}, X_{i_j}) = 0$, then $(X_{i_1}, \dots, X_{i_d})$ are independent.

PROOF. The random vector $(X_{i_1}, \dots, X_{i_d})'$ is a Gaussian vector with mean $(\mathbb{E}[X_{i_1}], \dots, \mathbb{E}[X_{i_d}])'$ and diagonal covariance matrix $\text{diag}(\mathbb{V}[X_{i_1}], \dots, \mathbb{V}[X_{i_d}])$. Consider $(\xi_{i_1}, \dots, \xi_{i_d})$ i.i.d. random variables with distribution $\mathcal{N}(0, 1)$ and define, for all $1 \leq j \leq d$,

$$Z_{i_j} = \mathbb{E}[\xi_{i_j}] + \sqrt{\mathbb{V}[X_{i_j}]} \xi_{i_j}.$$

Then, the random vector $(Z_{i_1}, \dots, Z_{i_d})'$ is a Gaussian vector with the same mean and the same covariance matrix as $(X_{i_1}, \dots, X_{i_d})'$. The two vectors have therefore the same characteristic function and the same law and $(X_{i_1}, \dots, X_{i_d})$ are independent as $(\xi_{i_1}, \dots, \xi_{i_d})$ are independent. ■

Theorem 4.3 (Cochran). *Let $X \sim \mathcal{N}(0, I_n)$ be a Gaussian vector in \mathbb{R}^n , F be a vector subspace of \mathbb{R}^n and F^\perp its orthogonal. Denote by $\pi_F(X)$ (resp. $\pi_{F^\perp}(X)$) the orthogonal projection of X on F (resp. on F^\perp). Then, $\pi_F(X)$ and $\pi_{F^\perp}(X)$ are independent, $\|\pi_F(X)\|^2 \sim \chi^2(p)$ and $\|\pi_{F^\perp}(X)\|^2 \sim \chi^2(n-p)$, where p is the dimension of F .*

PROOF. Let (u_1, \dots, u_n) be an orthonormal basis of \mathbb{R}^n where (u_1, \dots, u_p) is an orthonormal basis of F and (u_{p+1}, \dots, u_n) and orthonormal basis of F^\perp . Consider the matrix $U \in \mathbb{R}^{n \times n}$ such that for all $1 \leq i \leq n$, the i -th column of U is u_i and $U_{(p)}$ (reps. $U_{(n-p)}^\perp$) the matrix made of the first p (resp. last $n-p$) columns of U . Note that

$$\pi_F(X) = \sum_{i=1}^p \langle X; u_i \rangle u_i,$$

which can be written $\pi_F(X) = U_{(p)} U_{(p)}' X$. Similarly, $\pi_{F^\perp}(X) = U_{(n-p)}^\perp (U_{(n-p)}^\perp)' X$. Therefore,

$$\begin{pmatrix} \pi_F(X) \\ \pi_{F^\perp}(X) \end{pmatrix} = \begin{pmatrix} U_{(p)} U_{(p)}' \\ U_{(n-p)}^\perp (U_{(n-p)}^\perp)' \end{pmatrix} X$$

is a centered Gaussian vector with covariance matrix given by

$$\begin{pmatrix} U_{(p)} U_{(p)}' & 0 \\ 0 & U_{(n-p)}^\perp (U_{(n-p)}^\perp)' \end{pmatrix}.$$

By Proposition 4.2, $\pi_F(X)$ and $\pi_{F^\perp}(X)$ are independent. On the other hand,

$$\|\pi_F(X)\|^2 = \sum_{i=1}^p \langle X; u_i \rangle^2 \quad \text{and} \quad \|\pi_{F^\perp}(X)\|^2 = \sum_{i=p+1}^n \langle X; u_i \rangle^2.$$

The random vector $(\langle X; u_i \rangle)_{1 \leq i \leq n}$ is given by $U'X$: it is a Gaussian random vector with mean 0 and covariance matrix I_n . The random variables $(\langle X; u_i \rangle)_{1 \leq i \leq n}$ are therefore i.i.d. with distribution $\mathcal{N}(0, 1)$, which concludes the proof. ■

4.2 Full rank multivariate regression

4.2.1 Least squares estimator

It is assumed that for all $1 \leq i \leq n$, $Y_i = X_i' \beta_\star + \varepsilon_i$ where the $(\varepsilon_i)_{1 \leq i \leq n}$ are i.i.d. random variables in \mathbb{R}^d , $X_i \in \mathbb{R}^d$ and β_\star is an unknown vector in \mathbb{R}^d . Let $Y \in \mathbb{R}^d$ (resp. $\varepsilon \in \mathbb{R}^d$) be the random vector such that for all $1 \leq i \leq n$, the i -th component of Y (resp. ε) is Y_i (resp. ε_i) and $X \in \mathbb{R}^{n \times d}$ the matrix with line i equal to X_i' . The model is then written

$$Y = X \beta_\star + \varepsilon.$$

In this section, it is assumed that $\mathbb{E}[\varepsilon] = 0$ and $\mathbb{E}[\varepsilon \varepsilon'] = \sigma_\star^2 I_n$ and that the matrix X has full rank, i.e. the columns of X are linearly independent. The least squares estimate of β_\star is defined as a solution to

$$\hat{\beta}_n \in \arg \min_{\beta \in \mathbb{R}^d} \|Y - X\beta\|^2.$$

Proposition 4.4 *If the matrix X has full rank, then, $\hat{\beta}_n = (X'X)^{-1}X'Y$. This estimator is unbiased and satisfies $\mathbb{V}[\hat{\beta}_n] = \sigma_*^2(X'X)^{-1}$.*

PROOF. For all $\beta \in \mathbb{R}^d$,

$$\|Y - X\beta\|^2 = \|Y\|^2 + \beta'X'X\beta + 2Y'X\beta.$$

The function $\ell : \beta \mapsto \|Y\|^2 + \beta'X'X\beta + 2Y'X\beta$ is convex and for all $\beta \in \mathbb{R}^d$,

$$\nabla \ell(\beta) = 2X'X\beta + 2X'Y.$$

As the matrix X has full rank, $X'X$ is nonsingular and $\nabla \ell(\beta) = 0$ has a unique solution given by

$$\hat{\beta}_n = (X'X)^{-1}X'Y.$$

First, note that $\hat{\beta}_n$ is unbiased as

$$\mathbb{E}[\hat{\beta}_n] = (X'X)^{-1}X'\mathbb{E}[Y] = (X'X)^{-1}X'X\beta_* = \beta_*.$$

In addition,

$$\mathbb{V}[\hat{\beta}_n] = (X'X)^{-1}X'\mathbb{V}[Y]X(X'X)^{-1} = \sigma_*^2(X'X)^{-1}X'X(X'X)^{-1} = \sigma_*^2(X'X)^{-1}.$$

■

Proposition 4.5 *In the case where $\varepsilon \sim \mathcal{N}(0, \sigma_*^2 I_n)$, the random variable*

$$\hat{\sigma}_n^2 = \frac{\|Y - X\hat{\beta}_n\|^2}{n-d}$$

is an unbiased estimator of σ_ . In addition, $(n-d)\hat{\sigma}_n^2/\sigma_*^2 \sim \chi^2(n-d)$, $\hat{\beta}_n \sim \mathcal{N}(\beta_*, \sigma_*^2(X'X)^{-1})$ and $\hat{\beta}_n$ and $\hat{\sigma}_n^2$ are independent.*

PROOF. By definition of $\hat{\beta}_n$,

$$\hat{\sigma}_n^2 = \frac{\|Y - X\hat{\beta}_n\|^2}{n-d} = \frac{\|Y - X(X'X)^{-1}X'Y\|^2}{n-d} = \frac{\|(I_n - X(X'X)^{-1}X')Y\|^2}{n-d}$$

The columns of X are linearly independent so that the matrix of the orthogonal projection on $\text{Range}(X)$ is $X(X'X)^{-1}X'$ and therefore $(I_n - X(X'X)^{-1}X')$ is the matrix of the orthogonal projection on $\text{Range}(X)^\perp$. Then,

$$(I_n - X(X'X)^{-1}X')Y = (I_n - X(X'X)^{-1}X')(X\beta_* + \varepsilon) = (I_n - X(X'X)^{-1}X')\varepsilon.$$

By Theorem 4.3, $\|\sigma_*^{-1}(I_n - X(X'X)^{-1}X')\varepsilon\|^2$ has a χ^2 distribution with $n-d$ degrees of freedom which yields

$$\mathbb{E}[\|(I_n - X(X'X)^{-1}X')Y\|^2] = \sigma_*^2(n-d)$$

and $\mathbb{E}[\hat{\sigma}_n^2] = \sigma_*^2$. By Proposition 4.5, $\mathbb{E}[\hat{\beta}_n] = \beta_*$ and $\mathbb{V}[\hat{\beta}_n] = \sigma_*^2(X'X)^{-1}$ and $\hat{\beta}_n$ is a Gaussian vector as an affine transformation of a Gaussian vector. Note that $(n-d)\hat{\sigma}_n^2 = \|(I_n - X(X'X)^{-1}X')\varepsilon\|^2$ and $\hat{\beta}_n = (X'X)^{-1}X'X\beta_* + (X'X)^{-1}X'\varepsilon$ and that $(X'X)^{-1}X'\varepsilon$ and $(I_n - X(X'X)^{-1}X')\varepsilon$ are not correlated as

$$\mathbb{E}[(I_n - X(X'X)^{-1}X')\varepsilon \varepsilon' X(X'X)^{-1}] = \sigma_*^2 \mathbb{E}[(I_n - X(X'X)^{-1}X')X(X'X)^{-1}] = 0.$$

The independence follows from Proposition 4.2.

■

4.2.2 Confidence intervals and tests

Student's t-statistics

Proposition 4.6 For all $1 \leq j \leq n$,

$$\frac{\hat{\beta}_{n,j} - \beta_{*,j}}{\hat{\sigma}_n \sqrt{(X'X)^{-1}_{j,j}}} \sim \mathcal{S}(n-d),$$

where $\mathcal{S}(n-d)$ is the Student's t-distribution with $n-p$ degrees of freedom, i.e. the law of $X/\sqrt{Y/(n-d)}$ where $X \sim \mathcal{N}(0,1)$ is independent of $Y \sim \chi^2(n-d)$.

PROOF. By definition, for all $1 \leq j \leq d$,

$$\frac{\hat{\beta}_{n,j} - \beta_{*,j}}{\hat{\sigma}_n \sqrt{(X'X)^{-1}_{j,j}}} = \frac{\sigma_*^{-1}(\hat{\beta}_{n,j} - \beta_{*,j})}{\sigma_*^{-1} \hat{\sigma}_n \sqrt{(X'X)^{-1}_{j,j}}} = \frac{e'_j(\sigma_*^{-1}(\hat{\beta}_n - \beta_*))}{\sigma_*^{-1} \hat{\sigma}_n \sqrt{(X'X)^{-1}_{j,j}}}.$$

Note that $\sigma_*^{-1}(\hat{\beta}_n - \beta_*) \sim \mathcal{N}(0, (X'X)^{-1})$ so that $e'_j(\sigma_*^{-1}(\hat{\beta}_n - \beta_*)) \sim \mathcal{N}(0, e'_j(X'X)^{-1}e_j)$ and

$$\frac{e'_j(\sigma_*^{-1}(\hat{\beta}_n - \beta_*))}{\sqrt{(X'X)^{-1}_{j,j}}} \sim \mathcal{N}(0,1).$$

In addition,

$$\sigma_*^{-1} \hat{\sigma}_n = \sqrt{\sigma_*^{-2} \hat{\sigma}_n^2} = \sqrt{\|\sigma_*^{-1}(I_n - X(X'X)^{-1}X')\varepsilon\|^2/(n-d)},$$

where $\sigma_*^{-2} \hat{\sigma}_n^2 = \|\sigma_*^{-1}(I_n - X(X'X)^{-1}X')\varepsilon\|^2 \sim \chi^2(n-d)$. The proof is concluded by noting that $\hat{\beta}_n$ and $\hat{\sigma}_n^2$ are independent. ■

By Proposition 4.6, for $\alpha \in (0,1)$, if $s_{1-\alpha/2}^{n-d}$ denotes the quantile of order $1-\alpha/2$ of the law $\mathcal{S}(n-d)$, then

$$\mathbb{P}\left(\left|\frac{\hat{\beta}_{n,j} - \beta_{*,j}}{\hat{\sigma}_n \sqrt{(X'X)^{-1}_{j,j}}}\right| \leq s_{1-\alpha/2}^{n-d}\right) = 1 - \alpha.$$

Therefore,

$$I_{n,j}^{n-p}(\beta_*) = \left[\hat{\beta}_{n,j} - \hat{\sigma}_n \sqrt{(X'X)^{-1}_{j,j}}; \hat{\beta}_{n,j} + \hat{\sigma}_n \sqrt{(X'X)^{-1}_{j,j}}\right]$$

is a confidence interval for $\beta_{*,j}$ with confidence level $1-\alpha$. The result of Proposition 4.6 may also be used to perform the test

$$H_0 : \beta_{*,j} = 0 \quad \text{vs} \quad H_1 : \beta_{*,j} \neq 0.$$

Under H_0 , the random variable $T_{n,j}$ defined by

$$T_{n,j} = \frac{\hat{\beta}_{n,j}}{\hat{\sigma}_n \sqrt{(X'X)^{-1}_{j,j}}}$$

does not depend on β_* neither on σ_* and is distributed as a Student $\mathcal{S}(n-d)$ random variable. A statistical test with statistical significance $1-\alpha$ to decide whether $\beta_* \neq 0$ is $T_{n,j} < s_{1-\alpha/2}^{n-d}$.

Fisher statistics

Proposition 4.7 Let L be a $\mathbb{R}^{q \times d}$ matrix with rank $q \leq d$. Then,

$$\frac{(\hat{\beta}_n - \beta_*)' L' (L(X'X)^{-1} L')^{-1} L (\hat{\beta}_n - \beta_*)}{q \hat{\sigma}_n^2} \sim \mathcal{F}(q, n-d),$$

where $\mathcal{F}(q, n-d)$ is the Fisher distribution with q and $n-d$ degrees of freedom, i.e. the law of $(X/q)/(Y/(n-p))$ where $X \sim \chi^2(q)$ is independent of $Y \sim \chi^2(n-d)$.

PROOF. Note that $\text{rank}(L(X'X)^{-1} L') = \text{rank}(LL') = q$. The matrix $L(X'X)^{-1} L'$ is therefore positive definite. There exists a diagonal matrix $D \in \mathbb{R}^{q \times q}$ with positive diagonal terms and an orthogonal matrix $Q \in \mathbb{R}^{q \times q}$ such that $L(X'X)^{-1} L' = QDQ^{-1}$. The matrix $(L(X'X)^{-1} L')^{-1/2}$ may be defined as $(L(X'X)^{-1} L')^{-1/2} = QD^{-1/2}Q^{-1}$. It is then enough to note that $(L(X'X)^{-1} L')^{-1/2} L (\hat{\beta}_n - \beta_*) / \sigma_* \sim \mathcal{N}(0, I_q)$. Therefore,

$$\sigma_*^{-2} \|(L(X'X)^{-1} L')^{-1/2} L (\hat{\beta}_n - \beta_*)\|^2 = (\hat{\beta}_n - \beta_*)' L' (L(X'X)^{-1} L')^{-1} L (\hat{\beta}_n - \beta_*) / \sigma_*^2 \sim \chi^2(q).$$

On the other hand, by Proposition 4.5,

$$(n-d) \sigma_*^{-2} \hat{\sigma}_n^2 \sim \chi^2(n-d).$$

The proof is concluded by noting that $\hat{\beta}_n$ and $\hat{\sigma}_n^2$ are independent. ■

By Proposition 4.7, for $\alpha \in (0, 1)$, if $f_{1-\alpha}^{q, n-d}$ denotes the quantile of order $1 - \alpha$ of the law $\mathcal{F}(q, n-p)$, then

$$\mathbb{P} \left(\beta_* \in \left\{ \beta \in \mathbb{R}^d ; (\hat{\beta}_n - \beta)' L' (L(X'X)^{-1} L')^{-1} L (\hat{\beta}_n - \beta) \leq q \hat{\sigma}_n^2 f_{1-\alpha}^{q, n-d} \right\} \right) = 1 - \alpha.$$

Therefore,

$$I_n^{q, n-d}(\beta_*) = \left\{ \beta \in \mathbb{R}^d ; (\hat{\beta}_n - \beta)' L' (L(X'X)^{-1} L')^{-1} L (\hat{\beta}_n - \beta) \leq q \hat{\sigma}_n^2 f_{1-\alpha}^{q, n-d} \right\}$$

is a confidence region for β_* with confidence level $1 - \alpha$. The result of Proposition 4.7 may also be used to perform the test

$$H_0 : L\beta_* = \bar{\beta} \quad \text{vs} \quad H_1 : L\beta_* \neq \bar{\beta},$$

for a given $\bar{\beta} \in \mathbb{R}^d$.

4.3 Introduction to regularized multivariate regression

4.3.1 Ridge regression

In the case where $X'X$ is singular (resp. has eigenvalues close to zero), the least squares estimate cannot be computed (resp. is not robust). A common approach to control the estimator variance is to solve the surrogate Ridge regression problem

$$\hat{\beta}_n^{\text{ridge}} \in \arg \min_{\beta \in \mathbb{R}^d} \|Y - X\beta\|^2 + \lambda \|\beta\|^2,$$

where $\lambda > 0$. The matrix $X'X + \lambda I_n$ is definite positive for all $\lambda > 0$ as for all $u \in \mathbb{R}^d$,

$$u'(X'X + \lambda I_n)u = \|Xu\|^2 + \lambda \|u\|^2,$$

which is positive for all $u \neq 0$. This remark allows to obtain the following result.

Proposition 4.8 *The unique solution to the Ridge regression problem is given by*

$$\hat{\beta}_n^{\text{ridge}} = (X'X + \lambda I_n)^{-1} X'Y .$$

This estimator is biased and satisfies

$$\begin{aligned} \mathbb{E}[\hat{\beta}_n] - \beta_* &= -\lambda (X'X + \lambda I_n)^{-1} \beta_* , \\ \mathbb{V}[\hat{\beta}_n] &= \sigma_*^2 (X'X + \lambda I_n)^{-2} X'X . \end{aligned}$$

PROOF. ■

The mean square error of the estimator is then given by

$$\mathbb{E} \left[\left\| \hat{\beta}_n - \beta_* \right\|^2 \right] = \text{Trace} \left(\mathbb{V}[\hat{\beta}_n] \right) + \left\| \mathbb{E}[\hat{\beta}_n] - \beta_* \right\|^2 .$$

Let $(\vartheta_1, \dots, \vartheta_d)$ be an orthonormal basis of \mathbb{R}^d of eigenvectors of $X'X$ associated with the eigenvalues $(\gamma_1, \dots, \gamma_d) \in \mathbb{R}^d$. Then,

$$\mathbb{E} \left[\left\| \hat{\beta}_n - \beta_* \right\|^2 \right] = \sigma_*^2 \sum_{j=1}^d \frac{\gamma_j}{(\gamma_j + \lambda)^2} + \lambda^2 \sum_{j=1}^d \frac{\langle \beta_* ; \vartheta_j \rangle^2}{(\gamma_j + \lambda)^2} .$$

The mean square error is therefore a sum of two contributions, a bias related term which increases with λ and a variance related term which decreases with λ . In practice, the value of λ is chosen using cross-validation.

4.3.2 Lasso regression

The Least Absolute Shrinkage and Selection Operator (Lasso) regression is a L_1 based regularized regression which aims at fostering sparsity. The objective is to solve the following minimization problem,

$$\hat{\beta}_n^{\text{lasso}} \in \arg \min_{\beta \in \mathbb{R}^d} \|Y - X\beta\|^2 + \lambda \|\beta\|_1 ,$$

where $\lambda > 0$ and

$$\|\beta\|_1 = \sum_{j=1}^d |\beta_j| .$$

The function $\beta \mapsto \|Y - X\beta\|^2 + \lambda \|\beta\|_1$ is convex but not differentiable and the solution to this problem may not be unique. For all $\beta \in \mathbb{R}^d$,

$$\partial_\beta \|Y - X\beta\|^2 = -2X'(Y - X\beta) .$$

Then, for all $1 \leq j \leq d$, $(\partial_\beta \|Y - X\beta\|^2)_j = -2\mathbf{X}'_j(Y - X\beta)$, where \mathbf{X}_j is the j -th column of the matrix X . Define, for all $1 \leq j \leq d$,

$$v_j = \mathbf{X}'_j \left(Y - \sum_{\substack{i=1 \\ i \neq j}}^n \beta_i \mathbf{X}_i \right) ,$$

then,

$$(\partial_{\beta} \|Y - X\beta\|^2)_j = -2(v_j - \beta_j) .$$

Consequently, for all $\beta_j \neq 0$,

$$\partial_j(\|Y - X\beta\|^2 + \lambda \|\beta\|_1) = 2(\beta_j - v_j + \lambda \text{sign}(\beta_j)/2) .$$

For all $1 \leq j \leq d$, $\beta_j \mapsto \|Y - X\beta\|^2 + \lambda \|\beta\|_1$ is convex and grows to infinity when $|\beta_j| \rightarrow \infty$ and admits thus a minimum at some $\beta_j^* \in \mathbb{R}$.

- If $\beta_j^* \neq 0$, then

$$\beta_j^* = v_j \left(1 - \frac{\lambda \text{sign}(\beta_j^*)}{2v_j} \right) ,$$

which yields, as $\text{sign}(\beta_j^*) = \text{sign}(v_j)$,

$$\beta_j^* = v_j \left(1 - \frac{\lambda}{2|v_j|} \right)$$

and

$$1 - \frac{\lambda}{2|v_j|} \geq 0 .$$

- If $1 - \lambda/(2|v_j|) < 0$, there is no solution to $\partial_j(\|Y - X\beta\|^2 + \lambda \|\beta\|_1) = 0$ for $\beta_j \neq 0$. Since $\beta_j \mapsto \|Y - X\beta\|^2 + \lambda \|\beta\|_1$ admits a minimum, $\beta_j^* = 0$.

Therefore,

$$\beta_j^* = v_j \left(1 - \frac{\lambda}{2|v_j|} \right)_+ .$$

An algorithm to approximatively solve the Lasso regression problem proceeds as follows.

4.3.3 Nonparametric regression

In a nonparametric regression framework, it is not assumed that the observations depend linearly on the covariates and a more general model is introduced. For all $1 \leq i \leq n$, the observation model is given by

$$Y_i = f^*(X_i) + \xi_i ,$$

where for all $1 \leq i \leq n$, $X_i \in \mathcal{X}$, and the $(\xi_i)_{1 \leq i \leq n}$ are i.i.d. centered Gaussian random variables with variance σ^2 . The function f^* is unknown and has to be estimated using the observations $(X_i, Y_i)_{1 \leq i \leq n}$. A simple approach consists in defining an estimator of f^* as a linear combination of $M \geq 1$ known functions $(\varphi_1, \dots, \varphi_M)$ defined on \mathcal{X} . Define \mathcal{F}_φ as

$$\mathcal{F}_\varphi = \left\{ \sum_{j=1}^M \alpha_j \varphi_j ; (\alpha_1, \dots, \alpha_M) \in \mathbb{R}^M \right\} .$$

Then, the least squares estimator of f^* on \mathcal{F}_φ is defined as

$$\hat{f}_n^\varphi \in \arg \min_{f \in \mathcal{F}_\varphi} \sum_{i=1}^n (Y_i - f(X_i))^2 .$$

Let Ψ be the $\mathbb{R}^{M \times n}$ matrix such as, for all $1 \leq i \leq n$ and $1 \leq j \leq M$, $\Psi_{i,j} = \varphi_j(X_i)$. Then, for all $f \in \mathcal{F}_\varphi$, there exists $\alpha = (\alpha_1, \dots, \alpha_M) \in \mathbb{R}^M$ such that,

$$\sum_{i=1}^n (Y_i - f(X_i))^2 = \|Y - \Psi\alpha\|^2.$$

Then, following the same steps as in Section 4.2, in the case where $\Psi'\Psi$ is nonsingular, the least squares estimate is

$$\hat{f}_n^\varphi : x \mapsto \sum_{j=1}^M \hat{\alpha}_{n,j} \varphi_j, \quad (4.1)$$

where

$$\hat{\alpha}_n = (\Psi'\Psi)^{-1} \Psi'Y.$$

Introducing the function $\varphi : x \mapsto (\varphi_1(x), \dots, \varphi_M(x))'$ yields the linear estimator

$$\hat{f}_n^\varphi : x \mapsto \sum_{i=1}^n w_i(x) Y_i,$$

where, for all $1 \leq i \leq n$,

$$w_i(x) = (\varphi(x)'(\Psi'\Psi)^{-1}\Psi')_i.$$

Proposition 4.9 *Let $W = (w_i(X_j))_{1 \leq i, j \leq n}$ and $\bar{f}^* = (f^*(X_1), \dots, f^*(X_n))'$. Then,*

$$\frac{1}{n} \mathbb{E} \left[\sum_{i=1}^n (\hat{f}_n^\varphi(X_i) - f^*(X_i))^2 \right] = \frac{1}{n} \sum_{i=1}^n ((W\bar{f}^*)_i - f^*(X_i))^2 + \frac{\sigma^2}{n} \text{Trace}(W'W),$$

where \hat{f}_n^φ is defined by (4.1).

PROOF. See the exercises. ■

Chapter 5

Technical results

Contents

5.1	Probabilistic inequalities	51
5.2	Matrix calculus	52

5.1 Probabilistic inequalities

Theorem 5.1 (Hoeffding's inequality). *Let $(X_i)_{1 \leq i \leq n}$ be n independent random variables such that for all $1 \leq i \leq n$, $\mathbb{P}(a_i \leq X_i \leq b_i) = 1$ where a_i, b_i are real numbers such that $a_i < b_i$. Then, for all $t > 0$,*

$$\mathbb{P} \left(\left| \sum_{i=1}^n X_i - \sum_{i=1}^n \mathbb{E}[X_i] \right| > t \right) \leq 2 \exp \left(\frac{-2t^2}{\sum_{i=1}^n (b_i - a_i)^2} \right).$$

PROOF. Without loss of generality, assume that $\mathbb{E}[X_i] = 0$ for all $1 \leq i \leq n$. It is enough to prove that, for all $t > 0$,

$$\mathbb{P} \left(\sum_{i=1}^n X_i > t \right) \leq \exp \left(\frac{-2t^2}{\sum_{i=1}^n (b_i - a_i)^2} \right). \quad (5.1)$$

Equation (5.1) implies Hoeffding's inequality by noting that $\mathbb{P}(|\sum_{i=1}^n X_i| > t) \leq \mathbb{P}(\sum_{i=1}^n X_i > t) + \mathbb{P}(-\sum_{i=1}^n X_i > t)$ and by applying (5.1) to $(X_i)_{1 \leq i \leq n}$ and $(-X_i)_{1 \leq i \leq n}$. Write, for any $s, t > 0$,

$$\mathbb{P} \left(\sum_{i=1}^n X_i > t \right) = \mathbb{P} \left(e^{s \sum_{i=1}^n X_i} > e^{st} \right) < e^{-st} \mathbb{E} \left[e^{s \sum_{i=1}^n X_i} \right] = e^{-st} \prod_{i=1}^n \mathbb{E} \left[e^{s X_i} \right]$$

To bound the right hand side of this inequality, set, for all $1 \leq i \leq n$, $\phi_i : s \mapsto \log(\mathbb{E}[e^{s X_i}])$. Since X_i is almost surely bounded, ϕ_i is differentiable and for all $s > 0$, $\phi'_i(s) = \mathbb{E}[X_i e^{s X_i}] / \mathbb{E}[e^{s X_i}]$. Then, differentiating again,

$$\phi''_i(s) = \log''(\mathbb{E}[e^{s X_i}]) = \frac{\mathbb{E}[X_i^2 e^{s X_i}]}{\mathbb{E}[e^{s X_i}]} - \left(\frac{\mathbb{E}[X_i e^{s X_i}]}{\mathbb{E}[e^{s X_i}]} \right)^2 = \tilde{\mathbb{E}}_i[X^2] - (\tilde{\mathbb{E}}_i[X])^2 = \tilde{\mathbb{E}}_i[(X - \tilde{\mathbb{E}}_i[X])^2],$$

where

$$\tilde{\mathbb{E}}_i[Z] = \frac{\mathbb{E}[Z e^{s X_i}]}{\mathbb{E}[e^{s X_i}]}.$$

Then,

$$\phi_i''(s) = \inf_{x \in [a_i, b_i]} \tilde{\mathbb{E}}_i[(X - x)^2] \leq \tilde{\mathbb{E}}_i \left[\left(X - \frac{a_i + b_i}{2} \right)^2 \right] \leq \left(\frac{b_i - a_i}{2} \right)^2.$$

Finally, using Taylor's expansion,

$$\phi_i(s) \leq \phi_i(0) + \phi_i'(0)s + \frac{s^2}{2} \sup_{\alpha \in [0,1]} \phi_i''(\alpha s) \leq \frac{s^2(b_i - a_i)^2}{8}. \quad (5.2)$$

This implies

$$\mathbb{P} \left(\sum_{i=1}^n X_i > t \right) \leq e^{-st} e^{s^2 \sum_{i=1}^n \frac{(b_i - a_i)^2}{8}}.$$

Choosing $s = 4t / (\sum_{i=1}^n (b_i - a_i)^2)$ minimizes the right hand side and yields (5.1). ■

Lemma 5.2 *Let X be a Bernoulli random variable. Then, for all $t > 0$,*

$$\Psi(t) = \mathbb{E} \left[e^{t(X - \mathbb{E}[X])} \right] \leq e^{t^2/8}.$$

PROOF. Let $p \in (0, 1)$ be such that $p = \mathbb{P}(X = 1)$ (cases $p = 0$ and $p = 1$ are straightforward). For all $t > 0$,

$$\varphi(t) = \log \Psi(t) = \log(1 - p + pe^t) - pt.$$

The proof then follows from proof of the Hoeffding inequality, i.e. (5.2) with $b_i = 1 - p$ and $a_i = -p$. ■

5.2 Matrix calculus

Let M_d^+ the space of real-valued $d \times d$ symmetric positive matrices.

Lemma 5.3 *The function $\Sigma \mapsto \log \det \Sigma$ is concave on M_d^+ .*

PROOF. Let $\Sigma, \Gamma \in M_d^+$ and $\lambda \in [0, 1]$. Since $\Sigma^{-1/2} \Gamma \Sigma^{-1/2} \in M_d^+$, it is diagonalisable in some orthonormal basis and write μ_1, \dots, μ_d the (possibly repeated) entries of the diagonal. Note in particular that $\det(\Sigma^{-1/2} \Gamma \Sigma^{-1/2}) = \prod_{i=1}^d \mu_i$. Then,

$$\begin{aligned} \log \det((1 - \lambda)\Sigma + \lambda\Gamma) &= \log \det \left[\Sigma^{1/2} \left((1 - \lambda)I + \lambda \Sigma^{-1/2} \Gamma \Sigma^{-1/2} \right) \Sigma^{1/2} \right] \\ &= \log \det \Sigma + \log \det \left((1 - \lambda)I + \lambda \Sigma^{-1/2} \Gamma \Sigma^{-1/2} \right) \\ &= \log \det \Sigma + \sum_{i=1}^d \log(1 - \lambda + \lambda \mu_i) \\ &\geq \log \det \Sigma + \sum_{i=1}^d \underbrace{(1 - \lambda) \log(1) + \lambda \log(\mu_i)}_{=0} := D \end{aligned}$$

where the last inequality follows from the concavity of the log. Now, rewrite the rhs D as:

$$\begin{aligned} D &= (1 - \lambda) \log \det \Sigma + \lambda \left(\log \det \Sigma^{1/2} + \log \det \Sigma^{-1/2} \Gamma \Sigma^{-1/2} + \log \det \Sigma^{1/2} \right) \\ &= (1 - \lambda) \log \det \Sigma + \lambda \log \det \Gamma \end{aligned}$$

This finishes the proof. ■

Lemma 5.4 *Let Σ be a symmetric and invertible matrix in $\mathbb{R}^{d \times d}$.*

(i) The derivative of the real valued function $\Sigma \mapsto \log \det(\Sigma)$ defined on $\mathbb{R}^{d \times d}$ is given by:

$$\partial_{\Sigma} \{\log \det(\Sigma)\} = \Sigma^{-1},$$

where, for all real valued function f defined on $\mathbb{R}^{d \times d}$, $\partial_{\Sigma} f(\Sigma)$ denotes the $\mathbb{R}^{d \times d}$ matrix such that for all $1 \leq i, j \leq d$, $\{\partial_{\Sigma} f(\Sigma)\}_{i,j}$ is the partial derivative of f with respect to $\Sigma_{i,j}$.

(ii) The derivative of the real valued function $x \mapsto x' \Sigma x$ defined on \mathbb{R}^d is given by:

$$\partial_x \{x' \Sigma x\} = 2 \Sigma x.$$

PROOF.

(i) Recall that for all $i \in \{1, \dots, d\}$ we have $\det(\Sigma) = \sum_{k=1}^d \Sigma_{i,k} \Delta_{i,k}$ where $\Delta_{i,j}$ is the (i, j) -cofactor associated to Σ . For any fixed i, j , the component $\Sigma_{i,j}$ does not appear anywhere in the decomposition $\sum_{k=1}^d \Sigma_{i,k} \Delta_{i,k}$, except for the term $k = j$. This implies

$$\frac{\partial \log \det(\Sigma)}{\partial \Sigma_{i,j}} = \frac{1}{\det \Sigma} \frac{\partial \det(\Sigma)}{\partial \Sigma_{i,j}} = \frac{\Delta_{i,j}}{\det \Sigma}$$

Recalling the identity $\Sigma [\Delta_{j,i}]_{1 \leq i, j \leq d} = (\det \Sigma) I_d$ so that $\Sigma^{-1} = \frac{[\Delta_{j,i}]_{1 \leq i, j \leq d}^T}{\det \Sigma}$, we finally get

$$\left[\frac{\partial \log \det(\Sigma)}{\partial \Sigma_{i,j}} \right]_{1 \leq i, j \leq d} = (\Sigma^{-1})^T = \Sigma^{-1}$$

where the last equality follows from the fact that Σ is symmetric.

(ii) Define $\varphi(x) = x' \Sigma x$. Then, by straightforward algebra, $\varphi(x+h) = \varphi(x) + 2h' \Sigma x + \varphi(h) = \varphi(x) + 2h' \Sigma x + o(\|h\|)$, which concludes the proof. ■

Chapter

6

Exercices

Contents

6.1	Linear discriminant analysis	55
6.2	Plug-in classifier	57
6.3	Logistic Regression	58
6.4	K-means algorithm	61
6.5	Gaussian vectors	63
6.6	Regression: prediction of a new observation	63
6.7	Regression: linear estimators	63
6.8	Kernels	64
6.9	Penalized kernel regression	64
6.10	Expectation Maximization algorithm	65
	References	68

6.1 Linear discriminant analysis

Linear discriminant analysis assumes that the random variables $(X, Y) \in \mathbb{R}^p \times \{0, 1\}$ has the following distribution. For all $A \in \mathcal{B}(\mathbb{R}^p)$ and all $y \in \{0, 1\}$,

$$\mathbb{P}(X \in A; Y = y) = \pi_y \int_A g_y(x) dx ,$$

where π_0 and π_1 are positive real numbers such that $\pi_0 + \pi_1 = 1$ and g_0 (resp. g_1) is the probability density of a Gaussian random variable with mean $\mu_0 \in \mathbb{R}^d$ (resp. μ_1) and positive definite covariance matrix $\Sigma_0 \in \mathbb{R}^{d \times d}$ (resp. Σ_1). The Bayes classifier $h_* : \mathbb{R}^p \rightarrow \{0, 1\}$ is defined by

$$h_* : x \mapsto \mathbb{1}_{\{\pi_1 g_1(x) > \pi_0 g_0(x)\}} .$$

1. Give the distribution of the random variable X and prove that

$$\mathbb{P}(h_*(X) \neq Y) = \min_{h: \mathbb{R}^p \rightarrow \{0, 1\}} \{\mathbb{P}(h(X) \neq Y)\} .$$

For all $A \in \mathcal{B}(\mathbb{R}^p)$,

$$\begin{aligned} \mathbb{P}(X \in A) &= \mathbb{P}(Y = 0)\mathbb{P}(X \in A|Y = 0) + \mathbb{P}(Y = 1)\mathbb{P}(X \in A|Y = 1) , \\ &= \pi_0 \int_A g_0(x) dx + \pi_1 \int_A g_1(x) dx . \end{aligned}$$

The probability density of the random variable X is given, for all $x \in \mathbb{R}^d$, by

$$g(x) = \pi_0 g_0(x) + \pi_1 g_1(x) .$$

Then, note that for all $x \in \mathbb{R}^d$,

$$\eta(x) = \mathbb{P}(Y = 1 | X) |_{X=x} = \frac{\mathbb{P}(X | Y = 1) |_{X=x} \mathbb{P}(Y = 1)}{g(x)} = \frac{\pi_1 g_1(x)}{\pi_0 g_0(x) + \pi_1 g_1(x)} ,$$

and the condition $\eta(x) \leq 1/2$ can be rewritten as

$$\frac{\pi_1 g_1(x)}{\pi_0 g_0(x) + \pi_1 g_1(x)} \leq 1/2 ,$$

that is $\pi_1 g_1(x) \leq \pi_0 g_0(x)$.

2. Assume that $\mu_0 \neq \mu_1$. Prove that when $\Sigma_0 = \Sigma_1 = \Sigma$, for all $x \in \mathbb{R}^p$,

$$h_*(x) = 1 \Leftrightarrow (\mu_1 - \mu_0)^T \Sigma^{-1} \left(x - \frac{\mu_1 + \mu_0}{2} \right) > \log(\pi_0/\pi_1) .$$

Provide a geometrical interpretation.

For all $x \in \mathbb{R}^d$,

$$\begin{aligned} \pi_1 g_1(x) > \pi_0 g_0(x) &\Leftrightarrow \log(\pi_1 g_1(x)) > \log(\pi_0 g_0(x)) , \\ &\Leftrightarrow -\frac{1}{2}(x - \mu_1)^T \Sigma^{-1} (x - \mu_1) + \frac{1}{2}(x - \mu_0)^T \Sigma^{-1} (x - \mu_0) > \log(\pi_0/\pi_1) , \\ &\Leftrightarrow -\frac{1}{2} \left(-\mu_1^T \Sigma^{-1} x + \mu_1^T \Sigma^{-1} \mu_1 - x^T \Sigma^{-1} \mu_1 + \mu_0^T \Sigma^{-1} x - \mu_0^T \Sigma^{-1} \mu_0 + x^T \Sigma^{-1} \mu_0 \right) > \log(\pi_0/\pi_1) , \\ &\Leftrightarrow x^T \Sigma^{-1} \mu_1 - x^T \Sigma^{-1} \mu_0 - \frac{1}{2} \mu_1^T \Sigma^{-1} \mu_1 + \frac{1}{2} \mu_0^T \Sigma^{-1} \mu_0 > \log(\pi_0/\pi_1) , \\ &\Leftrightarrow (\mu_1 - \mu_0)^T \Sigma^{-1} \left(x - \frac{\mu_1 + \mu_0}{2} \right) > \log(\pi_0/\pi_1) . \end{aligned}$$

Therefore, all $x \in \mathbb{R}^d$ is classified according to its position with respect to an affine hyperplane orthogonal to $\Sigma^{-1}(\mu_1 - \mu_0)$.

3. Prove that when $\pi_1 = \pi_0$,

$$\mathbb{P}(h_*(X) = 1 | Y = 0) = \Phi(-d(\mu_1, \mu_0)/2) ,$$

where Φ is the cumulative distribution function of a standard Gaussian random variable and

$$d(\mu_1, \mu_0)^2 = (\mu_1 - \mu_0)^T \Sigma^{-1} (\mu_1 - \mu_0) .$$

Let Z_0 be a Gaussian random variable with mean μ_0 and variance Σ . Note that

$$\mathbb{P}(h_*(X) = 1 | Y = 0) = \mathbb{P} \left(\underbrace{(\mu_1 - \mu_0)^T \Sigma^{-1} (Z_0 - \frac{\mu_1 + \mu_0}{2})}_Z > 0 \right) ,$$

where, using $\delta = d(\mu_1, \mu_0)$,

$$\mathbb{E}[Z] = (\mu_1 - \mu_0)^T \Sigma^{-1} \left(\frac{\mu_0 - \mu_1}{2} \right) = -\frac{\delta^2}{2}$$

and

$$\mathbb{V}[Z] = \mathbb{V}[(\mu_1 - \mu_0)^T \Sigma^{-1} X] = ((\mu_1 - \mu_0)^T \Sigma^{-1}) \Sigma (\Sigma^{-1}(\mu_1 - \mu_0)) = \delta^2.$$

Hence,

$$\mathbb{P}(h_*(X) = 1 | Y = 0) = \mathbb{P}\left(-\frac{\delta^2}{2} + \delta \varepsilon > 0\right) = \mathbb{P}\left(\varepsilon > \frac{\delta}{2}\right) = \Phi\left(-\frac{\delta}{2}\right).$$

4. When $\Sigma_1 \neq \Sigma_0$, what is the nature of the frontier between $\{x; h_*(x) = 1\}$ and $\{x; h_*(x) = 0\}$?

In this case, for all $x \in \mathbb{R}^d$,

$$\begin{aligned} \pi_1 g_1(x) > \pi_0 g_0(x) &\Leftrightarrow \log(\pi_1 g_1(x)) > \log(\pi_0 g_0(x)), \\ &\Leftrightarrow -\frac{1}{2}(x - \mu_1)^T \Sigma_1^{-1} (x - \mu_1) + \frac{1}{2}(x - \mu_0)^T \Sigma_0^{-1} (x - \mu_0) > \log(\pi_0/\pi_1), \\ &\Leftrightarrow \frac{1}{2}x' \Sigma_0^{-1} x - \frac{1}{2}x' \Sigma_1^{-1} x + x^T \Sigma_1^{-1} \mu_1 - x^T \Sigma_0^{-1} \mu_0 - \frac{1}{2}\mu_1^T \Sigma_1^{-1} \mu_1 + \frac{1}{2}\mu_0^T \Sigma_0^{-1} \mu_0 > \log(\pi_0/\pi_1). \end{aligned}$$

As the quadratic term does not vanish anymore, the frontier between $\{x; h_*(x) = 1\}$ and $\{x; h_*(x) = 0\}$ is a quadric.

6.2 Plug-in classifier

Let (X, Y) be random variables defined on the same probability space $(\Omega, \mathcal{F}, \mathbb{P})$. For any classifier $h : \mathcal{X} \rightarrow \{-1, 1\}$, define its classification error by

$$R(h) = \mathbb{P}(Y \neq h(X)).$$

The best classifier in terms of the classification error R is the Bayes classifier

$$h_*(x) = \text{sign}(\eta(x) - 1/2),$$

where

$$\eta : x \mapsto \mathbb{P}(Y = 1 | X=x).$$

Given n independent couples $\{(X_i, Y_i)\}_{1 \leq i \leq n}$ with the same distribution as (X, Y) , an empirical surrogate for h_* is obtained from a possibly nonparametric estimator $\hat{\eta}_n$ of η :

$$\hat{h}_n : x \mapsto \text{sign}(\hat{\eta}_n(x) - 1/2).$$

1. Prove that for any classifier $h : \mathcal{X} \rightarrow \{-1, 1\}$,

$$\mathbb{P}(Y \neq h(X) | X) = (2\eta(X) - 1) \mathbb{1}_{h(X)=-1} + 1 - \eta(X)$$

and

$$R(h) - R(h_*) = 2\mathbb{E} \left[\left| \eta(X) - \frac{1}{2} \right| \mathbb{1}_{h(X) \neq h_*(X)} \right].$$

For all $x \in \mathcal{X}$,

$$\begin{aligned} \mathbb{P}(Y \neq h(X) | X) &= \mathbb{P}(Y = -1, h(X) = 1 | X) + \mathbb{P}(Y = 1, h(X) = -1 | X), \\ &= \mathbb{1}_{h(X)=1} \mathbb{P}(Y = -1 | X) + \mathbb{1}_{h(X)=-1} \mathbb{P}(Y = 1 | X), \\ &= \mathbb{1}_{h(X)=-1} (2\eta(X) - 1) + 1 - \eta(X). \end{aligned}$$

On the other hand,

$$\begin{aligned}
R(h) - R(h_*) &= \mathbb{E} [\mathbb{E} [\mathbb{P}(Y \neq h(X)|X) - \mathbb{P}(Y \neq h_*(X)|X)|X]] , \\
&= \mathbb{E} [(2\eta(X) - 1) (\mathbb{1}_{h(X)=-1} - \mathbb{1}_{h_*(X)=-1})|X] , \\
&= \mathbb{E} [\mathbb{1}_{h_*(X) \neq h(X)} ((2\eta(X) - 1) \mathbb{1}_{h_*(X)=1} - (2\eta(X) - 1) \mathbb{1}_{h_*(X)=-1})|X] , \\
&= 2\mathbb{E} [\eta(X) - 1/2 | \mathbb{1}_{h_*(X) \neq h(X)}] .
\end{aligned}$$

2. Prove that

$$|\eta(x) - 1/2| \mathbb{1}_{\hat{h}_n(x) \neq h_*(x)} \leq |\eta(x) - \hat{\eta}_n(x)| \mathbb{1}_{\hat{h}_n(x) \neq h_*(x)} ,$$

where

$$\hat{h}_n : x \mapsto \text{sign}(\hat{\eta}_n(x) - 1/2) .$$

Deduce that

$$R(\hat{h}_n) - R(h_*) \leq 2 \|\hat{\eta}_n - \eta\|_{L^2(\mathbb{P}_X)} ,$$

where \mathbb{P}_X is the distribution of X .

Note that

$$\{x \in \mathcal{X} ; \hat{h}_n(x) \neq h_*(x)\} = \{x \in \mathcal{X} ; \eta(x) \geq 1/2, \hat{\eta}_n(x) \leq 1/2\} \cup \{x \in \mathcal{X} ; \eta(x) \leq 1/2, \hat{\eta}_n(x) \geq 1/2\} .$$

For all $x \in \{x \in \mathcal{X} ; \eta(x) \geq 1/2, \hat{\eta}_n(x) \leq 1/2\}$,

$$|\eta(x) - \hat{\eta}_n(x)| = \eta(x) - \hat{\eta}_n(x) \geq \eta(x) - 1/2$$

On the other hand, for all $x \in \{x \in \mathcal{X} ; \eta(x) \leq 1/2, \hat{\eta}_n(x) \geq 1/2\}$,

$$|\eta(x) - \hat{\eta}_n(x)| = \hat{\eta}_n(x) - \eta(x) \geq 1/2 - \eta(x) .$$

Therefore, for all $x \in \mathcal{X}$,

$$|\eta(x) - 1/2| \mathbb{1}_{\hat{h}_n(x) \neq h_*(x)} \leq |\eta(x) - \hat{\eta}_n(x)| \mathbb{1}_{\hat{h}_n(x) \neq h_*(x)} .$$

By the first question and Cauchy-Schwarz inequality,

$$R(\hat{h}_n) - R(h_*) = 2\mathbb{E} [\eta(X) - 1/2 | \mathbb{1}_{h_*(X) \neq \hat{h}_n(X)}] \leq 2\mathbb{E} [|\eta(X) - \hat{\eta}_n(X)| \mathbb{1}_{\hat{h}_n(X) \neq h_*(X)}] \leq 2 \|\eta - \hat{\eta}_n\|_{L^2(\mathbb{P}_X)} .$$

6.3 Logistic Regression

The *logistic model* assumes that the random variables $(X, Y) \in \mathbb{R}^p \times \{0, 1\}$ are such that

$$\mathbb{P}(Y = 1|X) = \frac{\exp(\langle \beta^*, X \rangle)}{1 + \exp(\langle \beta^*, X \rangle)} , \quad (6.1)$$

with $\beta^* \in \mathbb{R}^d$. In this case, for all $x \in \mathbb{R}^d$, $\mathbb{P}(Y = 1|X)_{X=x} > 1/2$ if and only if $\langle \beta^*, x \rangle > 0$, so the frontier between $\{x ; h_*(x) = 1\}$ and $\{x ; h_*(x) = 0\}$ is an hyperplane, with orthogonal direction β^* . The unknown parameter β^* may be estimated by maximizing the conditional likelihood of Y given X

$$\hat{\beta}_n \in \operatorname{argmax}_{\beta \in \mathbb{R}^d} \prod_{i=1}^n \left[\left(\frac{\exp(\langle \beta, x_i \rangle)}{1 + \exp(\langle \beta, x_i \rangle)} \right)^{Y_i} \left(\frac{1}{1 + \exp(\langle \beta, x_i \rangle)} \right)^{1-Y_i} \right] ,$$

to define the empirical classifier

$$\hat{h}_n : x \mapsto \mathbb{1}_{\langle \hat{\beta}_n, x \rangle > 0} .$$

1. Compute the gradient and the Hessian H_n of

$$\ell_n : \beta \mapsto - \sum_{i=1}^n [Y_i \langle x_i, \beta \rangle - \log(1 + \exp(\langle x_i, \beta \rangle))] .$$

What can be said about the function ℓ_n when for all $\beta \in \mathbb{R}^d$, $H_n(\beta)$ is nonsingular? This assumption is supposed to hold in the following questions.

Since for all $u \in \mathbb{R}^d$, $\nabla_\beta \langle u, \beta \rangle = u$,

$$\nabla \ell_n(\beta) = - \sum_{i=1}^n Y_i x_i + \sum_{i=1}^n \frac{\exp(\langle x_i, \beta \rangle)}{1 + \exp(\langle x_i, \beta \rangle)} x_i .$$

On the other hand, for all $1 \leq i \leq n$ and all $1 \leq j \leq d$,

$$\partial_j \left(\frac{\exp(\langle x_i, \beta \rangle)}{1 + \exp(\langle x_i, \beta \rangle)} x_i \right) = \frac{\exp(\langle x_i, \beta \rangle)}{(1 + \exp(\langle x_i, \beta \rangle))^2} x_{ij} x_i ,$$

where x_{ij} is the j th component of x_i . Then

$$(H_n(\beta))_{\ell j} = \sum_{i=1}^n \frac{\exp(\langle x_i, \beta \rangle)}{(1 + \exp(\langle x_i, \beta \rangle))^2} x_{ij} x_{i\ell} ,$$

that is,

$$H_n(\beta) = \sum_{i=1}^n \frac{\exp(\langle x_i, \beta \rangle)}{(1 + \exp(\langle x_i, \beta \rangle))^2} x_i x_i' .$$

$H_n(\beta)$ is a semi positive definite matrix, which implies that $\ell_n(\beta)$ is convex. If we assume that H_n is nonsingular, ℓ_n is strictly convex.

2. Prove that there exists $\tilde{\beta}_n \in \mathbb{R}^d$ such that $\|\tilde{\beta}_n - \beta^*\| \leq \|\hat{\beta}_n - \beta^*\|$ and

$$\hat{\beta}_n - \beta^* = -H_n(\tilde{\beta}_n)^{-1} \nabla \ell_n(\beta^*) .$$

Using a Taylor expansion between β^* and $\hat{\beta}_n$, there exists $\tilde{\beta}_n \in B(\beta^*, \|\hat{\beta}_n - \beta^*\|)$ such that

$$\nabla \ell_n(\hat{\beta}_n) = \nabla \ell_n(\beta^*) + H_n(\tilde{\beta}_n)(\hat{\beta}_n - \beta^*) .$$

By definition, $\ell_n(\hat{\beta}_n) = 0$. Therefore,

$$\hat{\beta}_n - \beta^* = -H_n(\tilde{\beta}_n)^{-1} \nabla \ell_n(\beta^*) ,$$

where $H_n(\tilde{\beta}_n)^{-1}$ exists since $H_n(\tilde{\beta})$ is assumed to be non-singular for all β .

In the following it is assumed that the $(x_i)_{1 \leq i \leq n}$ are uniformly bounded, $\hat{\beta}_n \rightarrow \beta^*$ a.s. and that there exists a continuous and nonsingular function H such that $n^{-1} H_n(\beta)$ converges to $H(\beta)$, uniformly in a ball around β^* .

3. Define for all $1 \leq i \leq n$, $p_i(\beta) = e^{\langle x_i, \beta \rangle} / (1 + e^{\langle x_i, \beta \rangle})$. Check that

$$\begin{aligned} \mathbb{E} \left[e^{-n^{-1/2} \langle t, \nabla \ell_n(\beta^*) \rangle} \right] &= \prod_{i=1}^n \left(1 - p_i(\beta^*) + p_i(\beta^*) e^{\langle t, x_i \rangle / \sqrt{n}} \right) e^{-p_i(\beta^*) \langle t, x_i \rangle / \sqrt{n}} , \\ &= \exp \left(\frac{1}{2} t^T (n^{-1} H_n(\beta^*)) t + O(n^{-1/2}) \right) . \end{aligned}$$

For all $t \in \mathbb{R}^d$,

$$\begin{aligned}\mathbb{E} \left[\exp \left(-\frac{1}{\sqrt{n}} \langle t, \nabla \ell_n(\beta^*) \rangle \right) \right] &= \prod_{i=1}^n \mathbb{E} \left[\exp \left(\frac{1}{\sqrt{n}} (Y_i - p_i(\beta^*)) \langle x_i, t \rangle \right) \right], \\ &= \prod_{i=1}^n \left[\left(1 - p_i(\beta^*) + p_i(\beta^*) \exp \left(\frac{1}{\sqrt{n}} \langle x_i, t \rangle \right) \right) \exp \left(-\frac{p_i(\beta^*)}{\sqrt{n}} \langle x_i, t \rangle \right) \right].\end{aligned}$$

Note that

$$\log(1 - p_i + p_i \exp(u/\sqrt{n})) = \log \left(1 + p_i \frac{u}{\sqrt{n}} + p_i \frac{u^2}{2n} + O(n^{-3/2}) \right) = p_i \frac{u}{\sqrt{n}} + \frac{p_i u^2}{2n} - \frac{p_i^2 u^2}{2n} + O(n^{-3/2}).$$

Finally,

$$\mathbb{E} \left[\exp \left(-\frac{1}{\sqrt{n}} \langle t, \nabla \ell_n(\beta^*) \rangle \right) \right] = \exp \left(\underbrace{\frac{1}{2n} \sum_{i=1}^n p_i(\beta^*) (1 - p_i(\beta^*)) \langle t, x_i \rangle^2}_{t^T H_n(\beta^*) t} + O(n^{-1/2}) \right).$$

4. What is the asymptotic distribution of $-n^{-1/2} \nabla \ell_n(\beta^*)$ and of $\sqrt{n}(\hat{\beta}_n - \beta^*)$?

For all $t \in \mathbb{R}^d$, since $n^{-1} H_n(\beta^*) \rightarrow_{n \rightarrow \infty} H(\beta^*)$,

$$\mathbb{E} \left[\exp \left(-\frac{1}{\sqrt{n}} \langle t, \nabla \ell_n(\beta^*) \rangle \right) \right] \rightarrow_{n \rightarrow \infty} \exp \left(\frac{1}{2} t^T H(\beta^*) t \right).$$

Therefore, $-\nabla \ell_n(\beta^*)/\sqrt{n}$ converges in distribution to $Z \sim \mathcal{N}(0, H(\beta^*))$. On the other hand,

$$\sqrt{n}(\hat{\beta}_n - \beta^*) = - \left(\frac{1}{n} H_n(\tilde{\beta}_n) \right)^{-1} \frac{1}{\sqrt{n}} \nabla \ell_n(\beta^*).$$

As for all $n \geq 1$, $\tilde{\beta}_n \in B(\beta^*, \|\hat{\beta}_n - \beta^*\|)$, $\tilde{\beta}_n$ converges to β^* almost surely as n grows to infinity. Hence, almost surely

$$\left(\frac{1}{n} H_n(\tilde{\beta}_n) \right)^{-1} \rightarrow H(\beta^*)^{-1}$$

and, by Slutsky lemma, $\sqrt{n}(\hat{\beta}_n - \beta^*)$ converges in distribution to $Z \sim \mathcal{N}(0, H(\beta^*)^{-1})$.

5. For all $1 \leq j \leq d$ and all $\alpha \in (0, 1)$, propose a confidence interval $\mathcal{J}_{n,\alpha}$ such that $\beta_j^* \in \mathcal{J}_{n,\alpha}$ with asymptotic probability $1 - \alpha$.

According to the last question, $\sqrt{n}(\hat{\beta}_j - \beta_j^*)$ converges in distribution to a centered Gaussian random variable with variance $(H(\beta^*)^{-1})_{jj}$. On the other hand, almost surely,

$$\hat{\sigma}_{n,j}^2 = (n H_n(\hat{\beta}_n)^{-1})_{jj} \rightarrow_{n \rightarrow \infty} (H(\beta^*)^{-1})_{jj}.$$

Then,

$$\sqrt{\frac{n}{\hat{\sigma}_{n,j}^2}} (\hat{\beta}_{n,j} - \beta_j^*) \rightarrow_{n \rightarrow \infty} \mathcal{N}(0, 1).$$

An asymptotic confidence interval $\mathcal{J}_{n,\alpha}$ of level $1 - \alpha$ is then given by

$$\mathcal{J}_{n,\alpha} = \left[\hat{\beta}_{n,j} - z_{1-\alpha/2} \sqrt{\frac{\hat{\sigma}_{n,j}^2}{n}}, \hat{\beta}_{n,j} + z_{1-\alpha/2} \sqrt{\frac{\hat{\sigma}_{n,j}^2}{n}} \right],$$

where $z_{1-\alpha/2}$ is the quantile of order $1 - \alpha/2$ of $\mathcal{N}(0, 1)$.

6. Propose a confidence ellipsoid $\mathcal{E}_{n,\alpha}$ such that the probability that $\beta^* \in \mathcal{E}_{n,\alpha}$ is asymptotically $1 - \alpha$.

6.4 K-means algorithm

The K-means algorithm is a procedure which aims at partitioning a data set into K distinct, non-overlapping clusters. Consider $n \geq 1$ observations (X_1, \dots, X_n) taking values in \mathbb{R}^p . The K -means algorithm seeks to minimize over all partitions $C = (C_1, \dots, C_K)$ of $\{1, \dots, n\}$ the following criterion

$$\text{crit}(C) = \sum_{k=1}^K \frac{1}{|C_k|} \sum_{a,b \in C_k} \|X_a - X_b\|^2,$$

where for all $1 \leq i \leq n$, $1 \leq k \leq K$, $i \in C_k$ if and only if X_i is in the k -th cluster.

1. Define the distance between two clusters $1 \leq i, j \leq K$ as

$$d(C_i, C_j) = \sum_{a \in C_i \cup C_j} \|X_a - \bar{X}_{C_i \cup C_j}\|^2 - \sum_{a \in C_i} \|X_a - \bar{X}_{C_i}\|^2 - \sum_{a \in C_j} \|X_a - \bar{X}_{C_j}\|^2.$$

Prove that for all $1 \leq i, j \leq K$,

$$d(C_i, C_j) = \frac{|C_i||C_j|}{|C_i| + |C_j|} \|\bar{X}_{C_i} - \bar{X}_{C_j}\|^2.$$

For all $1 \leq i, j \leq K$, note that

$$\bar{X}_{C_i \cup C_j} = \frac{|C_i|}{|C_i| + |C_j|} \bar{X}_{C_i} + \frac{|C_j|}{|C_i| + |C_j|} \bar{X}_{C_j},$$

so that

$$\begin{aligned} \sum_{a \in C_i} \|X_a - \bar{X}_{C_i \cup C_j}\|^2 &= \sum_{a \in C_i} \left\| X_a - \bar{X}_{C_i} + \frac{|C_j|}{|C_i| + |C_j|} (\bar{X}_{C_i} - \bar{X}_{C_j}) \right\|^2, \\ &= \sum_{a \in C_i} \|X_a - \bar{X}_{C_i}\|^2 + 2 \sum_{a \in C_i} \left\langle X_a - \bar{X}_{C_i}, \frac{|C_j|}{|C_i| + |C_j|} (\bar{X}_{C_i} - \bar{X}_{C_j}) \right\rangle \\ &\quad + |C_i| \left\| \frac{|C_j|}{|C_i| + |C_j|} (\bar{X}_{C_i} - \bar{X}_{C_j}) \right\|^2, \\ &= \sum_{a \in C_i} \|X_a - \bar{X}_{C_i}\|^2 + \frac{|C_i||C_j|^2}{(|C_i| + |C_j|)^2} \|\bar{X}_{C_i} - \bar{X}_{C_j}\|^2. \end{aligned}$$

Similarly,

$$\sum_{a \in C_j} \|X_a - \bar{X}_{C_i \cup C_j}\|^2 = \sum_{a \in C_j} \|X_a - \bar{X}_{C_j}\|^2 + \frac{|C_j||C_i|^2}{(|C_i| + |C_j|)^2} \|\bar{X}_{C_i} - \bar{X}_{C_j}\|^2.$$

Therefore,

$$\sum_{a \in C_i \cup C_j} \|X_a - \bar{X}_{C_i \cup C_j}\|^2 = \sum_{a \in C_i} \|X_a - \bar{X}_{C_i}\|^2 + \sum_{a \in C_j} \|X_a - \bar{X}_{C_j}\|^2 + \frac{|C_i||C_j|}{|C_i| + |C_j|} \|\bar{X}_{C_i} - \bar{X}_{C_j}\|^2,$$

which concludes the proof.

2. Establish that

$$\text{crit}(C) = 2 \sum_{k=1}^K \frac{1}{|C_k|} \sum_{a,b \in C_k} \langle X_a, X_a - X_b \rangle = 2 \sum_{k=1}^K \sum_{a \in C_k} \|X_a - \bar{X}_{C_k}\|^2,$$

where

$$\bar{X}_{C_k} = \frac{1}{|C_k|} \sum_{b \in C_k} X_b.$$

Note that

$$\begin{aligned} \text{crit}(C) &= \sum_{k=1}^K \frac{1}{|C_k|} \sum_{a,b \in C_k} \|X_a - X_b\|^2, \\ &= \sum_{k=1}^K \frac{1}{|C_k|} \sum_{a,b \in C_k} \langle X_a - X_b, X_a - X_b \rangle, \\ &= \sum_{k=1}^K \frac{1}{|C_k|} \left\{ \sum_{a,b \in C_k} \langle X_a - X_b, X_a \rangle + \langle X_b - X_a, X_b \rangle \right\}, \\ &= 2 \sum_{k=1}^K \frac{1}{|C_k|} \sum_{a,b \in C_k} \langle X_a - X_b, X_a \rangle. \end{aligned}$$

which concludes the proof of the first inequality. For the second inequality, write

$$\begin{aligned} \sum_{k=1}^K \sum_{a \in C_k} \|X_a - \bar{X}_{C_k}\|^2 &= \sum_{k=1}^K \sum_{a \in C_k} \langle X_a - \frac{1}{|C_k|} \sum_{b \in C_k} X_b, X_a - \frac{1}{|C_k|} \sum_{c \in C_k} X_c \rangle, \\ &= \sum_{k=1}^K \frac{1}{|C_k|^2} \sum_{a,b,c \in C_k} \langle X_a - X_b, X_a - X_c \rangle, \\ &= \sum_{k=1}^K \frac{1}{|C_k|^2} \sum_{a,b,c \in C_k} \langle X_a - X_b, X_a \rangle - \sum_{k=1}^K \frac{1}{|C_k|^2} \sum_{a,b,c \in C_k} \langle X_a - X_b, X_c \rangle, \end{aligned}$$

where

$$\sum_{a,b,c \in C_k} \langle X_a - X_b, X_c \rangle = |C_k| \sum_{a,c \in C_k} \langle X_a, X_c \rangle - |C_k| \sum_{b,c \in C_k} \langle X_b, X_c \rangle = 0.$$

Thus,

$$\text{crit}(C) = 2 \sum_{k=1}^K \sum_{a \in C_k} \|X_a - \bar{X}_{C_k}\|^2.$$

3. Prove that the criterion monotonically decreases with the iterations of the K-means algorithm.

For any cluster C in and any $z \in \mathbb{R}^p$,

$$\sum_{a \in C} \|X_a - z\|^2 = \sum_{a \in C} \|X_a - \bar{X}_C\|^2 + \sum_{a \in C} \|\bar{X}_C - z\|^2 + 2 \sum_{a \in C} \langle \bar{X}_C - z, X_a - \bar{X}_C \rangle = \sum_{a \in C} \|X_a - \bar{X}_C\|^2 + |C| \|\bar{X}_C - z\|^2,$$

so that

$$\sum_{a \in C} \|X_a - z\|^2 \geq \sum_{a \in C} \|X_a - \bar{X}_C\|^2,$$

which is enough to conclude the proof.

4. Assume that the observations are independent random variables. Define $\mu_a \in \mathbb{R}^p$ as the expectation of X_a so that $X_a = \mu_a + \varepsilon_a$ with $(\varepsilon_1, \dots, \varepsilon_n)$ centered and independent. Define also $v_a = \text{trace}(\text{cov}(X_a))$. Prove that

$$\mathbb{E}[\text{crit}(C)] = \sum_{k=1}^K \frac{1}{|C_k|} \sum_{a,b \in C_k} (\|\mu_a - \mu_b\|^2 + v_a + v_b) \mathbf{1}_{a \neq b}.$$

What is the value of $\mathbb{E}[\text{crit}(C)]$ when all the within-group variables have the same mean?

6.5 Gaussian vectors

1. Let X be a Gaussian vector with mean $\mu \in \mathbb{R}^n$ and definite positive covariance matrix Σ . Prove that the characteristic function of X is given, for all $t \in \mathbb{R}^n$, by

$$\mathbb{E}[e^{i\langle t, X \rangle}] = e^{i\langle t, \mu \rangle - t' \Sigma t / 2}.$$

2. Let Σ be a positive definite matrix of $\mathbb{R}^{n \times n}$. Provide a solution to sample a Gaussian vector with covariance matrix Σ based on i.i.d. standard Gaussian variables.
3. Let ε be a random variable in $\{-1, 1\}$ such that $\mathbb{P}(\varepsilon = 1) = 1/2$. If $(X, Y)' \sim \mathcal{N}(0, I_2)$ explain why the following vectors are or are not Gaussian vectors.
 - $(X, \varepsilon X)$.
 - $(X, \varepsilon Y)$.
 - $(X, \varepsilon X + Y)$.
 - $(X, X + \varepsilon Y)$.
4. Let X be a Gaussian vector in \mathbb{R}^n with mean $\mu \in \mathbb{R}^n$ and covariance matrix $\sigma^2 I_n$. Prove that the random variables \bar{X}_n and $\hat{\sigma}_n^2$ defined as

$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i \quad \text{and} \quad \hat{\sigma}_n^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2$$

are independent.

6.6 Regression: prediction of a new observation

Consider the regression model given, for all $1 \leq i \leq n$, by

$$Y_i = X_i \beta_\star + \xi_i,$$

where $X \in \mathbb{R}^{n \times d}$ the $(\xi_i)_{1 \leq i \leq n}$ are i.i.d. centered Gaussian random variables with variance σ_\star^2 . Assume that $X'X$ has full rank and that β_\star and σ_\star^2 are estimated by

$$\hat{\beta}_n = (X'X)^{-1} X'Y \quad \text{and} \quad \hat{\sigma}_n^2 = \frac{\|Y - X\hat{\beta}_n\|^2}{n-d}.$$

Let $x_\star \in \mathbb{R}^d$ and assume that its associated observation $Y_\star = x_\star' \beta_\star + \varepsilon_\star$ is predicted by $\hat{Y}_\star = x_\star' \hat{\beta}_n$.

1. Provide the expression of $\mathbb{E}[(\hat{Y}_\star - x_\star' \beta_\star)^2]$?
2. Provide a confidence interval for $x_\star' \beta_\star$ with statistical significance $1 - \alpha$ for $\alpha \in (0, 1)$?

6.7 Regression: linear estimators

Consider the regression model given, for all $1 \leq i \leq n$, by

$$Y_i = f^*(X_i) + \xi_i ,$$

where for all $1 \leq i \leq n$, $X_i \in \mathcal{X}$, and the $(\xi_i)_{1 \leq i \leq n}$ are i.i.d. centered Gaussian random variables with variance σ^2 . In this exercise, f^* is estimated by a linear estimator of the form

$$\hat{f}_n : x \mapsto \sum_{i=1}^n w_i(x) Y_i .$$

Prove that

$$\frac{1}{n} \mathbb{E} \left[\sum_{i=1}^n (\hat{f}_n(X_i) - f^*(X_i))^2 \right] = \|W f^*(X) - f^*(X)\|^2 + \frac{\sigma^2}{n} \text{Trace}(W'W) ,$$

where $W = (w_i(X_j))_{1 \leq i, j \leq n}$ and $f^*(X) = (f^*(X_1), \dots, f^*(X_n))'$.

6.8 Kernels

Let \mathcal{H} be a RKHS associated with a positive definite kernel $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$.

1. Prove that for all $(x, y) \in \mathcal{X} \times \mathcal{X}$,

$$|f(x) - f(y)| \leq \|f\|_{\mathcal{H}} |k(x, \cdot) - k(y, \cdot)|_{\mathcal{H}} .$$

2. Prove that the kernel k associated with \mathcal{H} is unique, i.e. if \tilde{k} is another positive definite kernel satisfying the RKHS properties for \mathcal{H} , then $k = \tilde{k}$.
3. Prove that for all $x \in \mathcal{X}$, the function defined on \mathcal{H} by $\delta_x : f \mapsto f(x)$ is continuous.

6.9 Penalized kernel regression

Consider the regression model given, for all $1 \leq i \leq n$, by

$$Y_i = f^*(X_i) + \xi_i ,$$

where for all $1 \leq i \leq n$, $X_i \in \mathcal{X}$, and the $(\xi_i)_{1 \leq i \leq n}$ are i.i.d. centered Gaussian random variables with variance σ^2 . In this exercise, f^* is estimated by

$$\hat{f}_n = \arg \min_{f \in \mathcal{H}} \left\{ \frac{1}{n} \sum_{i=1}^n (y_i - f(X_i))^2 + \frac{\lambda}{n} \|f\|_{\mathcal{H}}^2 \right\} ,$$

with $\lambda > 0$ and \mathcal{H} a RKHS on \mathcal{X} with kernel k .

1. Check that $\hat{f}(x) = \sum_{j=1}^n \hat{\beta}_{n,j} k(X_j, x)$ where $\hat{\beta}_n$ is solution to

$$\hat{\beta}_n = \arg \min_{\beta \in \mathbb{R}^n} \{ \|y - K\beta\|^2 + \lambda \beta' K \beta \} ,$$

with K defined, for all $1 \leq i, j \leq n$, by $K_{i,j} = k(X_i, X_j)$. Provide the explicit expression of $\hat{\beta}_n$ when K is nonsingular.

2. Assume that $f^* \in \mathcal{H}$ and write

$$f_V^* : x \mapsto \sum_{i=1}^n \beta_i^* k(X_i, x)$$

the projection of f^* onto the vector subspace generated by $(k(X_i, \cdot))_{1 \leq i \leq n}$, with respect to the scalar product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$. Let $K = \sum_{i=1}^n \lambda_i u_i u_i^T$ be an eigenvalue decomposition of K . Check that

$$K\hat{\beta} = \sum_{i=1}^n \frac{\lambda_i}{\lambda_i + \lambda} \langle Y, u_i \rangle u_i$$

and

$$\|\mathbb{E}[K\hat{\beta}_n] - K\beta^*\|^2 = \sum_{i=1}^n \left(\frac{\lambda \lambda_i}{\lambda_i + \lambda} \right)^2 \langle \beta^*, u_i \rangle^2.$$

3. Prove that

$$\mathbb{V}[K\hat{\beta}_n] = \sum_{i=1}^n \left(\frac{\lambda_i \sigma}{\lambda_i + \lambda} \right)^2 u_i u_i'.$$

6.10 Expectation Maximization algorithm

In the case where we are interested in estimating unknown parameters $\theta \in \mathbb{R}^m$ characterizing a model with missing data, the Expectation Maximization (EM) algorithm (Dempster et al. 1977) can be used when the joint distribution of the missing data X and the observed data Y is explicit. For all $\theta \in \mathbb{R}^m$, let p_θ be the probability density function of (X, Y) when the model is parameterized by θ with respect to a given reference measure μ . The EM algorithm aims at computing iteratively an approximation of the maximum likelihood estimator which maximizes the observed data loglikelihood:

$$\ell(\theta; Y) = \log p_\theta(Y) = \log \int f_\theta(x, Y) \mu(dx).$$

As this quantity cannot be computed explicitly in general cases, the EM algorithm finds the maximum likelihood estimator by iteratively maximizing the expected complete data loglikelihood. Start with an initial value $\theta^{(0)}$ and let $\theta^{(t)}$ be the estimate at the t -th iteration for $t \geq 0$, then the next iteration of EM is decomposed into two steps.

1. **E step.** Compute the expectation of the complete data loglikelihood, with respect to the conditional distribution of the missing data given the observed data parameterized by $\theta^{(t)}$:

$$Q(\theta, \theta^{(t)}) = \mathbb{E}_{\theta^{(t)}} [\log p_\theta(X, Y) | Y].$$

2. **M step.** Determine $\theta^{(t+1)}$ by maximizing the function Q :

$$\theta^{(t+1)} \in \operatorname{argmax}_\theta Q(\theta, \theta^{(t)}).$$

1. Prove the following crucial property motivates the EM algorithm. For all $\theta, \theta^{(t)}$,

$$\ell(Y; \theta) - \ell(Y; \theta^{(t)}) \geq Q(\theta, \theta^{(t)}) - Q(\theta^{(t)}, \theta^{(t)}).$$

This may be proved by noting that

$$\ell(Y; \theta) = \log \left(\frac{p_\theta(X, Y)}{p_\theta(X | Y)} \right).$$

Considering the conditional expectation of both terms given Y when the parameter value is $\theta^{(t)}$ yields

$$\ell(Y; \theta) = Q(\theta, \theta^{(t)}) - \mathbb{E}_{\theta^{(t)}} [\log p_\theta(X | Y) | Y].$$

Then,

$$\ell(Y; \theta) - \ell(Y; \theta^{(t)}) = Q(\theta, \theta^{(t)}) - Q(\theta^{(t)}, \theta^{(t)}) + H(\theta, \theta^{(t)}) - H(\theta^{(t)}, \theta^{(t)}),$$

where

$$H(\theta, \theta^{(t)}) = -\mathbb{E}_{\theta^{(t)}}[\log p_{\theta}(X|Y)|Y].$$

The proof is completed by noting that

$$H(\theta, \theta^{(t)}) - H(\theta^{(t)}, \theta^{(t)}) \geq 0,$$

as this difference is a Kullback-Leibler divergence.

In the following, $X = (X_1, \dots, X_n)$ and $Y = (Y_1, \dots, Y_n)$ where $\{(X_i, Y_i)\}_{1 \leq i \leq n}$ are i.i.d. in $\{-1, 1\} \times \mathbb{R}^d$. For $k \in \{-1, 1\}$, write $\pi_k = \mathbb{P}(X_1 = k)$. Assume that, conditionally on the event $\{X_1 = k\}$, Y_1 has a Gaussian distribution with mean $\mu_k \in \mathbb{R}^d$ and covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$. In this case, the parameter $\theta = (\pi_1, \mu_1, \mu_{-1}, \Sigma)$ belongs to the set $\Theta = [0, 1] \times \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^{d \times d}$.

1. Write the complete data loglikelihood.

The complete data loglikelihood is given by

$$\begin{aligned} \log p_{\theta}(X, Y) &= -\frac{nd}{2} \log(2\pi) + \sum_{i=1}^n \sum_{k \in \{-1, 1\}} \mathbb{1}_{X_i=k} \left(\log \pi_k - \frac{\log \det \Sigma}{2} - \frac{1}{2} (Y_i - \mu_k)^T \Sigma^{-1} (Y_i - \mu_k) \right), \\ &= -\frac{nd}{2} \log(2\pi) - \frac{n}{2} \log \det \Sigma + \left(\sum_{i=1}^n \mathbb{1}_{X_i=1} \right) \log \pi_1 + \left(\sum_{i=1}^n \mathbb{1}_{X_i=-1} \right) \log(1 - \pi_1) \\ &\quad - \frac{1}{2} \sum_{i=1}^n \mathbb{1}_{X_i=1} (Y_i - \mu_1)^T \Sigma^{-1} (Y_i - \mu_1) - \frac{1}{2} \sum_{i=1}^n \mathbb{1}_{X_i=-1} (Y_i - \mu_{-1})^T \Sigma^{-1} (Y_i - \mu_{-1}). \end{aligned}$$

2. Let $\theta^{(t)}$ be the current parameter estimate. Compute $\theta \mapsto Q(\theta, \theta^{(t)})$.

Write $\omega_t^i = \mathbb{P}_{\theta^{(t)}}(X_i = 1|Y_i)$. The intermediate quantity of the EM algorithm is given by

$$\begin{aligned} Q(\theta, \theta^{(t)}) &= -\frac{nd}{2} \log(2\pi) - \frac{n}{2} \log \det \Sigma + \left(\sum_{i=1}^n \omega_t^i \right) \log \pi_1 + \sum_{i=1}^n (1 - \omega_t^i) \log(1 - \pi_1) \\ &\quad - \frac{1}{2} \sum_{i=1}^n \omega_t^i (Y_i - \mu_1)^T \Sigma^{-1} (Y_i - \mu_1) - \frac{1}{2} \sum_{i=1}^n (1 - \omega_t^i) (Y_i - \mu_{-1})^T \Sigma^{-1} (Y_i - \mu_{-1}). \end{aligned}$$

3. Compute $\theta^{(t+1)}$.

The gradient of $Q(\theta, \theta^{(t)})$ with respect to θ is therefore given by

$$\begin{aligned} \frac{\partial Q(\theta, \theta^{(t)})}{\partial \pi_1} &= \frac{\sum_{i=1}^n \omega_t^i}{\pi_1} - \frac{n - \sum_{i=1}^n \omega_t^i}{1 - \pi_1}, \\ \frac{\partial Q(\theta, \theta^{(t)})}{\partial \mu_1} &= \sum_{i=1}^n \omega_t^i (2\Sigma^{-1} Y_i - 2\Sigma^{-1} \mu_1), \\ \frac{\partial Q(\theta, \theta^{(t)})}{\partial \mu_{-1}} &= \sum_{i=1}^n (1 - \omega_t^i) (2\Sigma^{-1} Y_i - 2\Sigma^{-1} \mu_{-1}), \\ \frac{\partial Q(\theta, \theta^{(t)})}{\partial \Sigma^{-1}} &= \frac{n}{2} \Sigma - \frac{1}{2} \sum_{i=1}^n \omega_t^i (Y_i - \mu_1) (Y_i - \mu_1)^T - \frac{1}{2} \sum_{i=1}^n (1 - \omega_t^i) (Y_i - \mu_{-1}) (Y_i - \mu_{-1})^T. \end{aligned}$$

Then, $\theta^{(t+1)}$ is defined as the only parameter such that all these equations are set to 0. It is given by

$$\begin{aligned}\hat{\pi}_1^{(t+1)} &= \frac{1}{n} \sum_{i=1}^n \omega_t^i, \\ \hat{\mu}_1^{(t+1)} &= \frac{1}{\sum_{i=1}^n \omega_t^i} \sum_{i=1}^n \omega_t^i Y_i, \\ \hat{\Sigma}^{(t+1)} &= \frac{1}{n} \sum_{i=1}^n \omega_t^i (Y_i - \mu_1) (Y_i - \mu_1)^T + \frac{1}{n} \sum_{i=1}^n (1 - \omega_t^i) (Y_i - \mu_{-1}) (Y_i - \mu_{-1})^T.\end{aligned}$$

References

- Hotelling, 1933. Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24(6):417–441.
- James et al., 2013. James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.