V - Classification

Math 728 D - Machine Learning & Data Science - Spring 2019

Contents

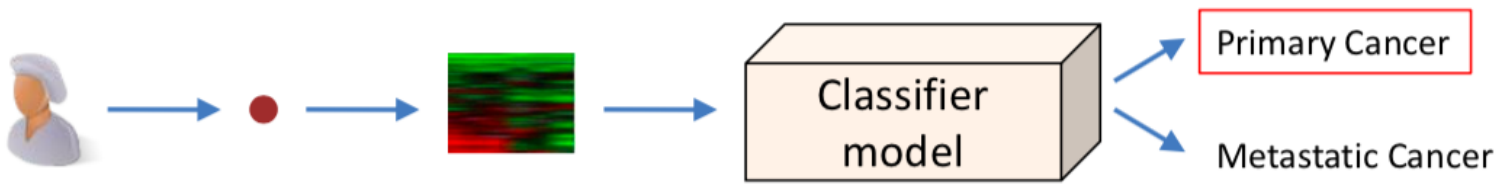
About Supervised Learning Tasks ...

- Data/observations come as samples $\mathbf{z}^i = (\mathbf{x}^i, \mathbf{y}^i)$ from a product space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ where typically $\mathcal{X} = \mathbb{R}^d$, $\mathcal{Y} = \mathbb{R}^m$. The first component $\mathbf{x}$ represents attributes/features identifying members of entity of interest (medical images, customers, etc.), the second component $\mathbf{y}^i$ represents properties associated with the features $\mathbf{x}^i$.
- It suffices to consider scalar properties $\mathbf{y}^i = y^i$, i.e., $m = 1$. The "spatial" variable d instead of the features could be large.
- Given a set $\mathfrak{Z}_N = \{\mathbf{z}^1, \ldots, \mathbf{z}^N\} \subset \mathcal{Z}$ of samples (data), the key objective is Prediction/Generalization. This means, given a new feature $\mathbf{x}$, predict its property y, using knowledge of $\mathfrak{Z}_N$.
- Without any kind of additional a priori information - prior - it is impossible to predict/learn y from the data $\mathfrak{Z}_N$. For instance, one can only quantify a polynomial interpolation error, if one knows something about the derivatives of the interpolated function.
- In Supervised Learning is given in terms of the following (Statistical) Model: The samples $\mathbf{z}^i = (\mathbf{x}^i, y^i)$ are i.i.d. drawn instances of a random variable $(\mathbf{X}, Y)$ with joint density $p(\mathbf{x}, y)$. Typically $p(\mathbf{x}, y)$ is unknown (except perhaps for information about range and support) - (non-parametric estimation).
- If y takes only finitely many values - labels, we talk about classification - this lecture. The goal then is to assign to any new feature $\mathbf{x}$ a label from this finite set. For most purposes it suffices to understand binary classification, i.e., $\mathcal{Y} = \{0, 1\}$ or $\{-1, 1\}$.
- If the properties y have a continuous range, one seeks a function $\hat{f}(\mathbf{x}) \approx y$. This is called regression - next lecture.

The following illustrations are taken from: A. Statnikov, D. Hardin, I. Guyon, C. F. Aliferis

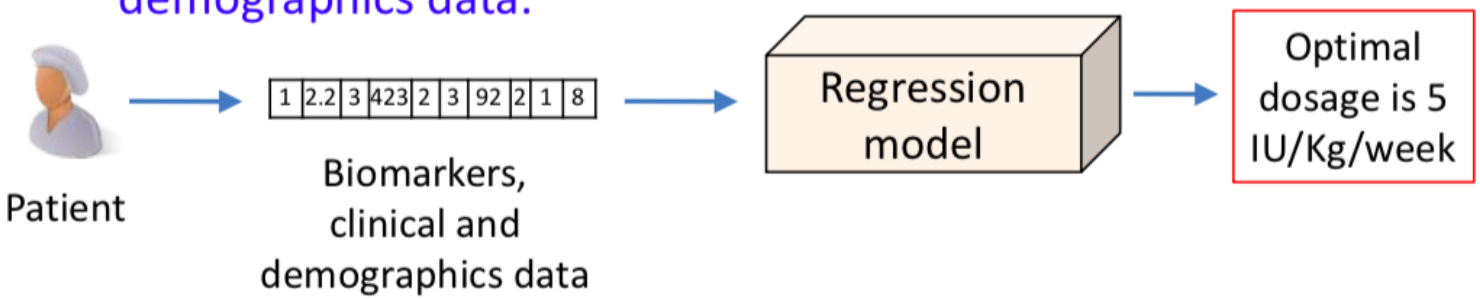
Data-analysis problems of interest

1. Build computational classification models (or "*classifiers*") that assign patients/samples into two or more classes.
   - Classifiers can be used for diagnosis, outcome prediction, and other classification tasks.
   - E.g., build a decision-support system to diagnose primary and metastatic cancers from gene expression profiles of the patients:



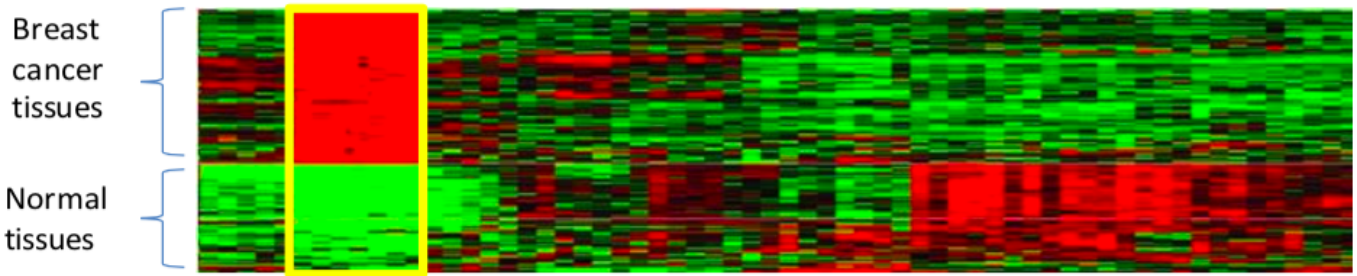
Data-analysis problems of interest

2. Build computational regression models to predict values of some continuous response variable or outcome.
   - Regression models can be used to predict survival, length of stay in the hospital, laboratory test values, etc.
   - E.g., build a decision-support system to predict optimal dosage of the drug to be administered to the patient. This dosage is determined by the values of patient biomarkers, and clinical and demographics data:



Data-analysis problems of interest

3. Out of all measured variables in the dataset, select the smallest subset of variables that is necessary for the most accurate prediction (classification or regression) of some variable of interest (e.g., phenotypic response variable).
   - E.g., find the most compact panel of breast cancer biomarkers from microarray gene expression data for 20,000 genes:



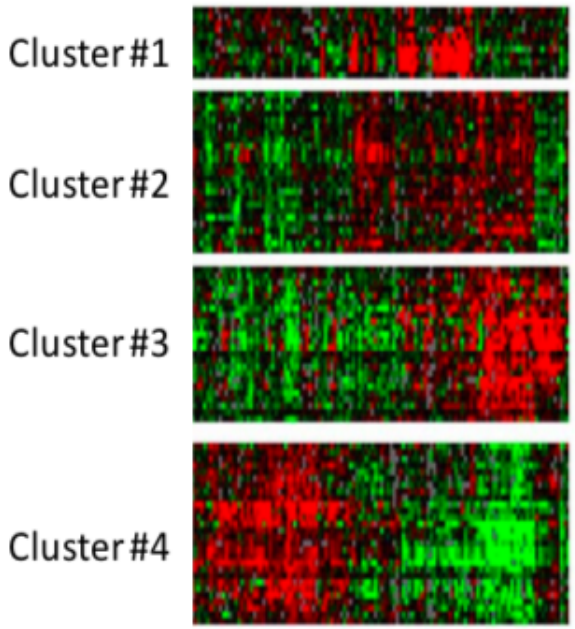
Data-analysis problems of interest

4. Build a computational model to identify novel or outlier patients/samples.
   - Such models can be used to discover deviations in sample handling protocol when doing quality control of assays, etc.
   - E.g., build a decision-support system to identify aliens.

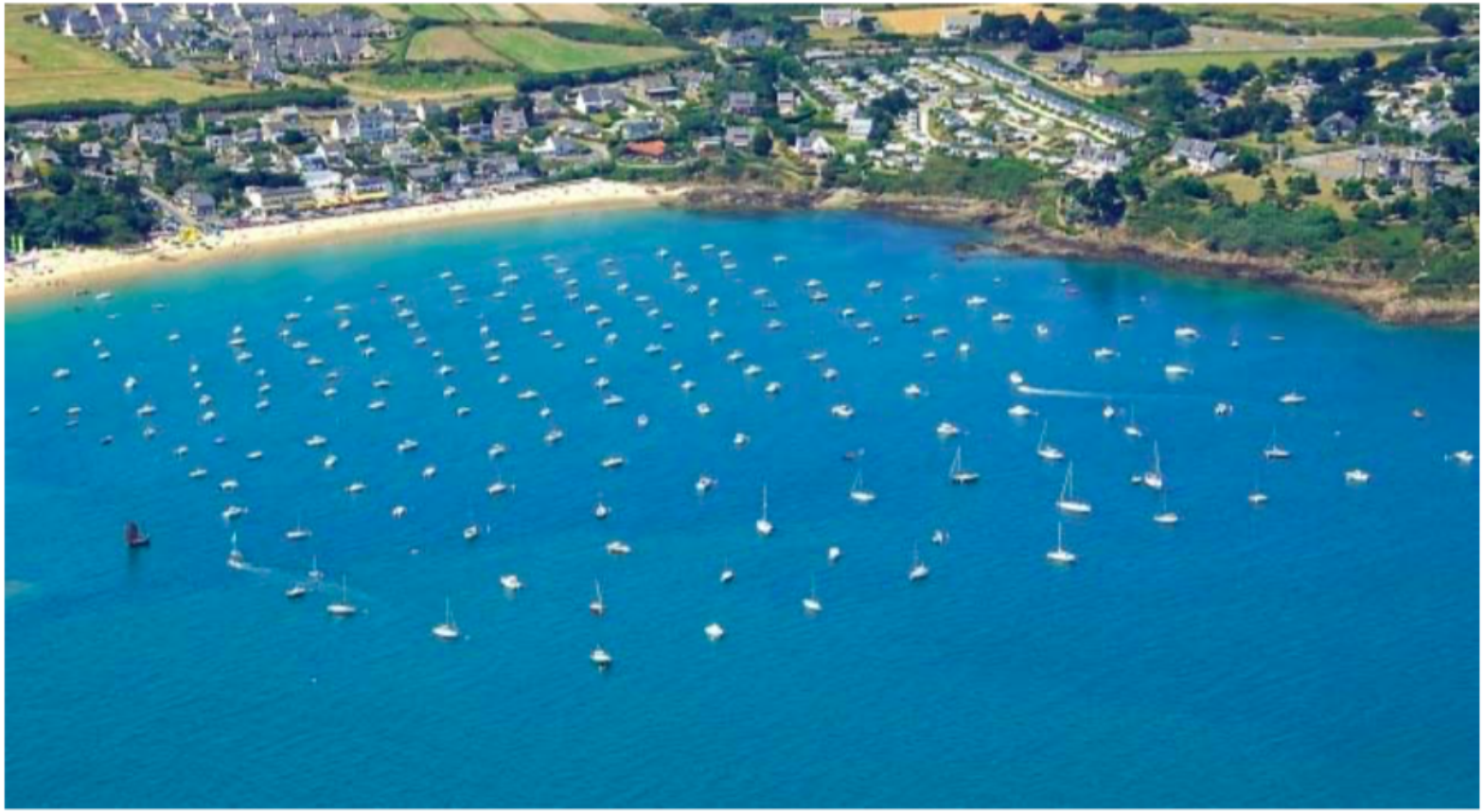
8

Data-analysis problems of interest

5. Group patients/samples into several clusters based on their similarity.
   - These methods can be used to discovery disease sub-types and for other tasks.
   - E.g., consider clustering of brain tumor patients into 4 clusters based on their gene expression profiles. All patients have the same pathological sub-type of the disease, and clustering discovers new disease subtypes that happen to have different characteristics in terms of patient survival and time to recurrence after treatment.

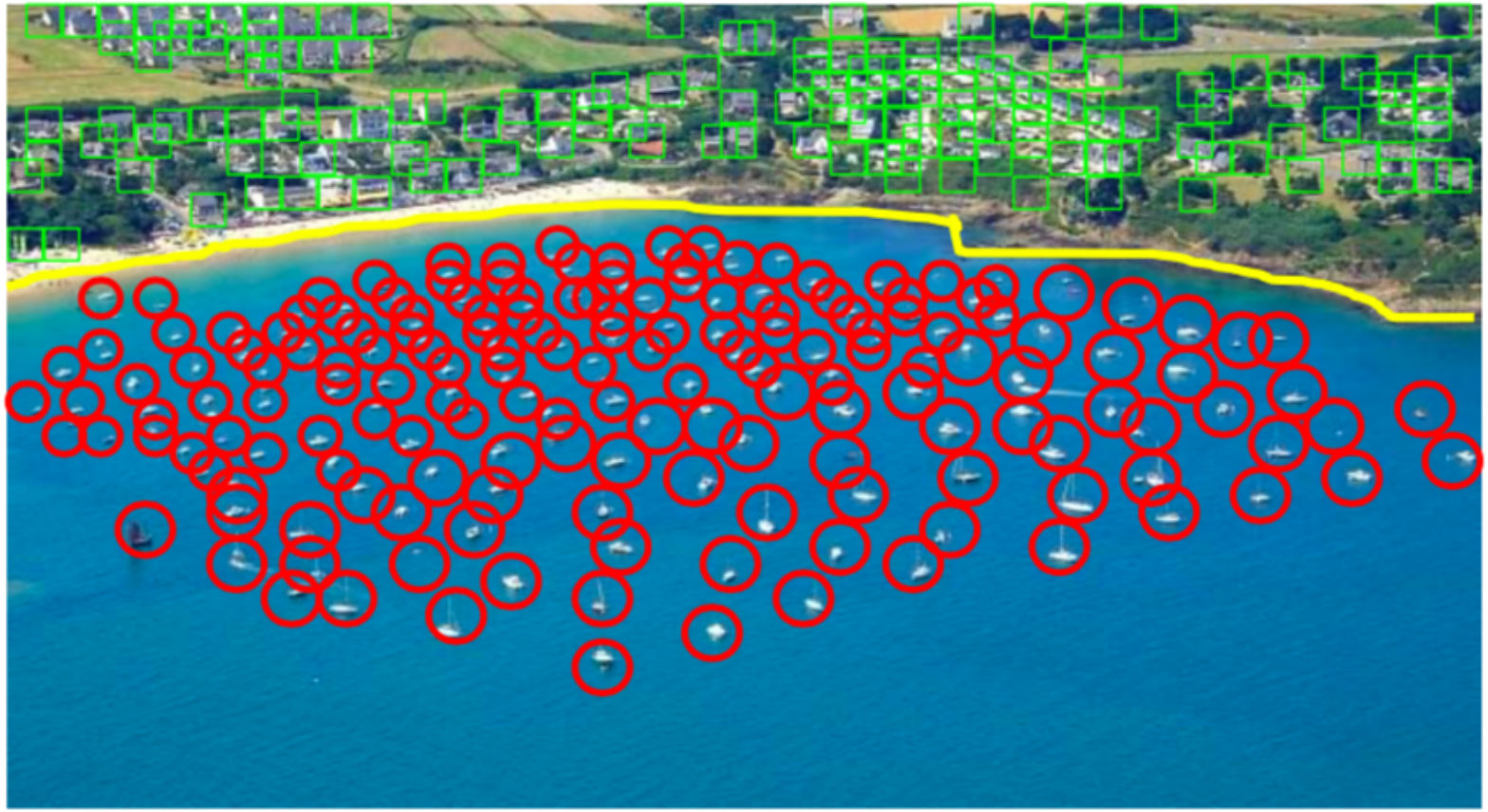


Basic principles of classification



- Want to classify objects as boats and houses.

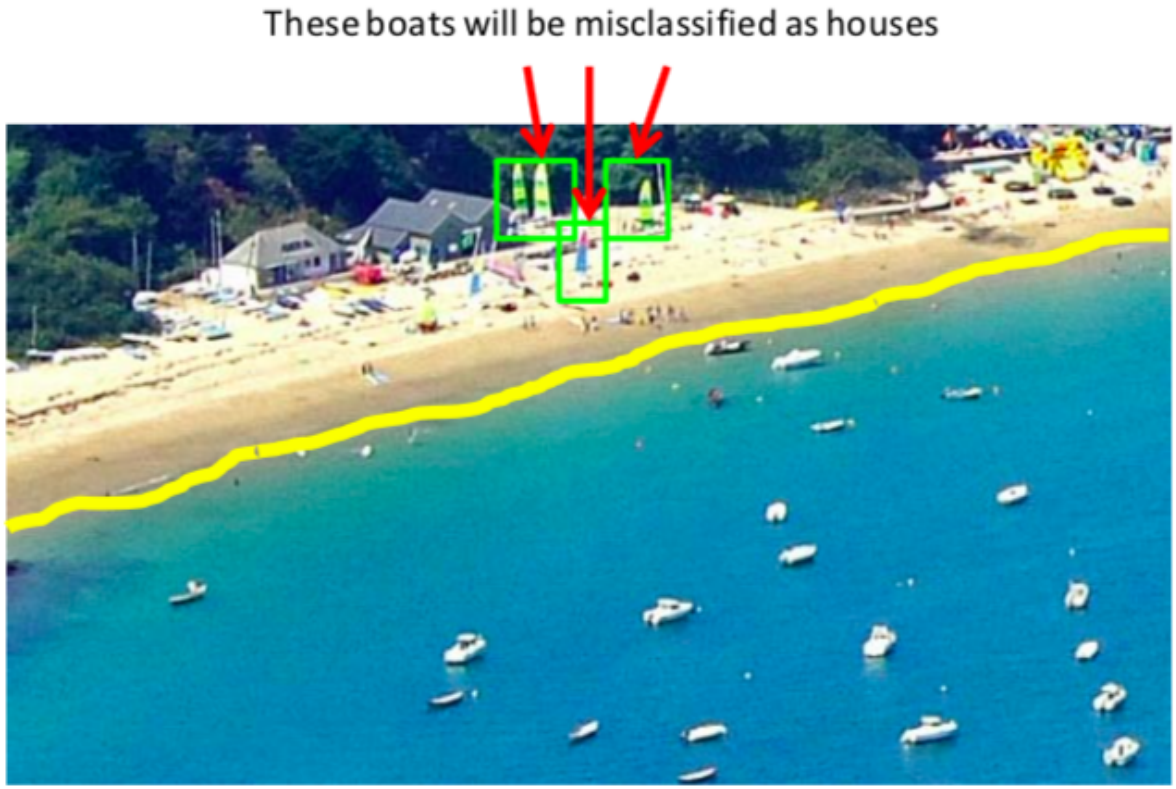
Basic principles of classification



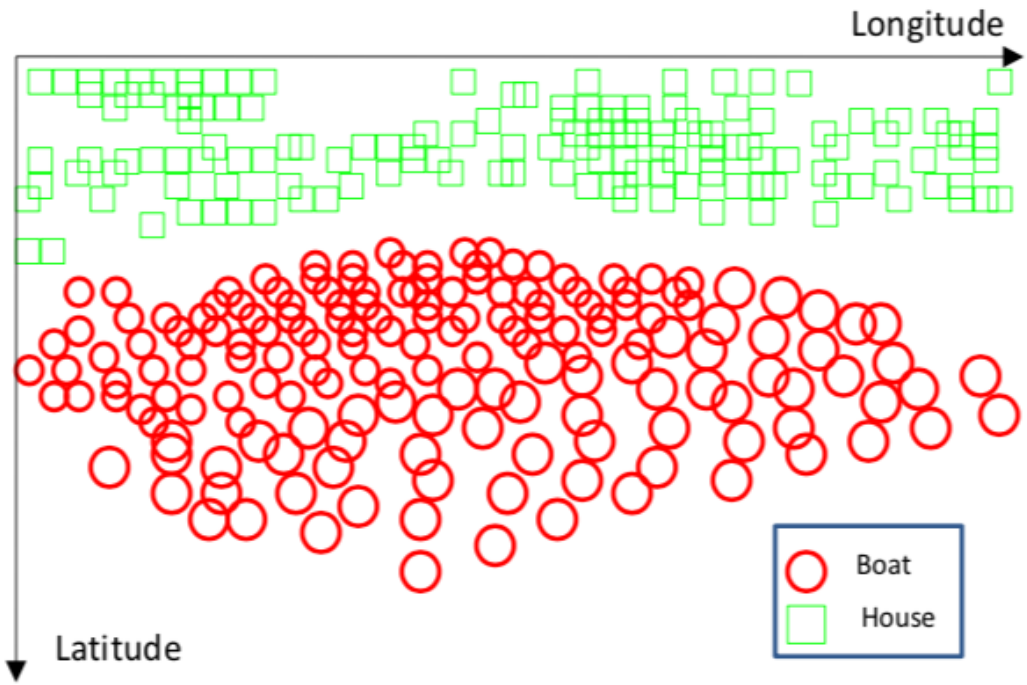
- All objects before the coast line are boats and all objects after the coast line are houses.
- Coast line serves as a *decision surface* that separates two classes.

11

Basic principles of classification



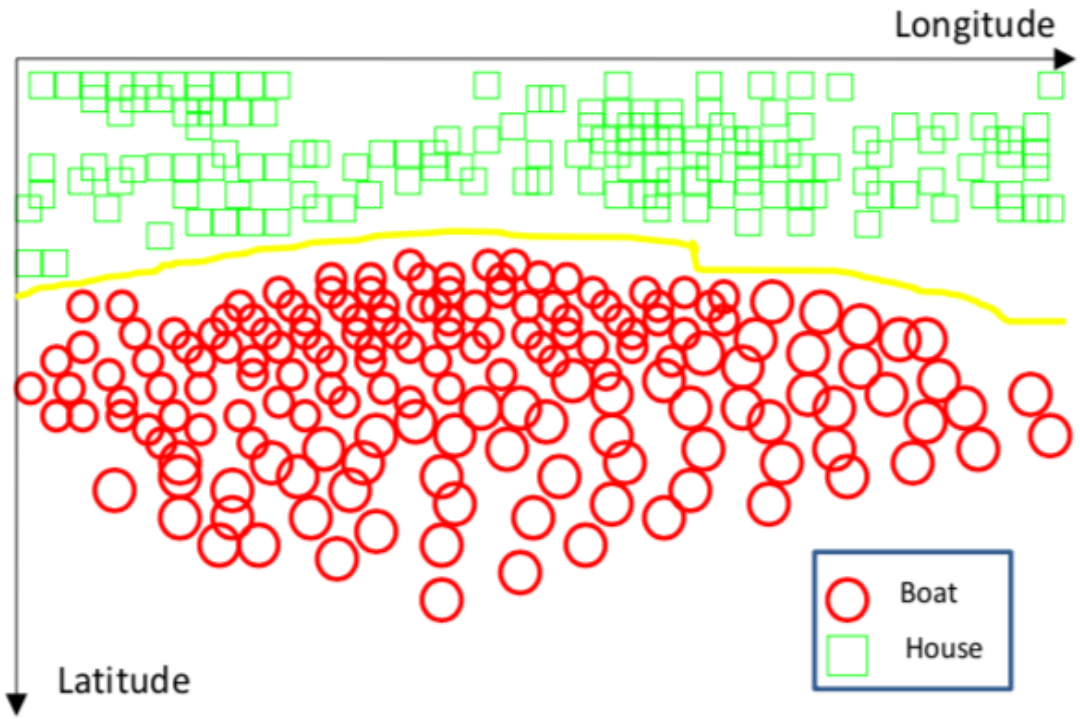
Basic principles of classification



- The methods that build classification models (i.e., *"classification algorithms"*) operate very similarly to the previous example.
- First all objects are represented geometrically.

13

Basic principles of classification



Then the algorithm seeks to find a decision surface that separates classes of objects

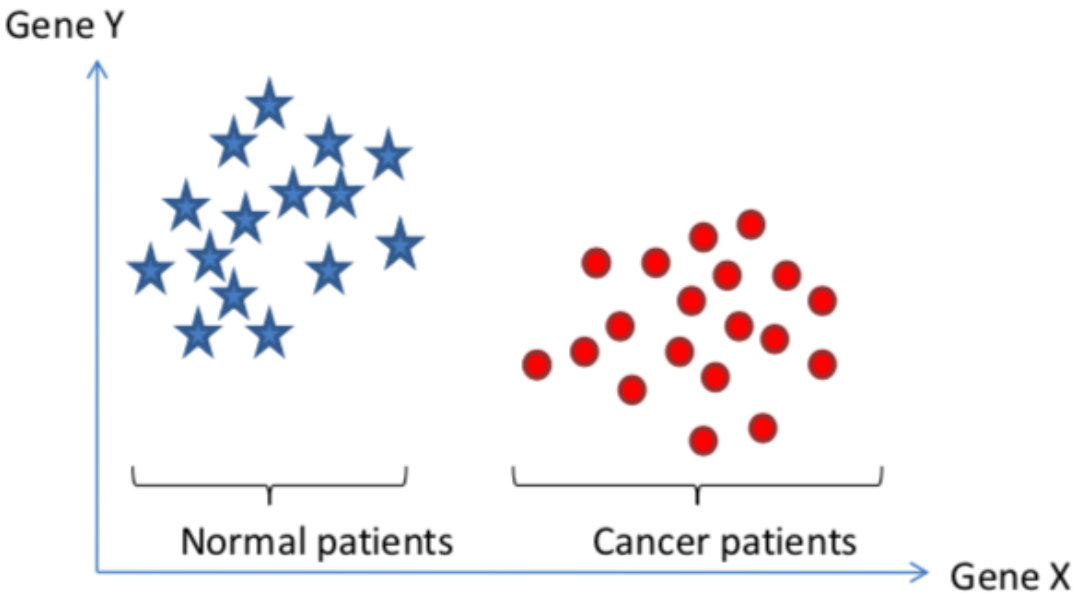
Main Methodologies

The objects to be classified, in the above examples patients, boats, and houses, are represented by feature vectors in a high-dimensional Euclidean space.

1. Nearest neighbor search;
2. Support Vector Machines (SVM) - Kernel Methods;
3. Decision trees - CART, Random forests;
4. Neural networks.

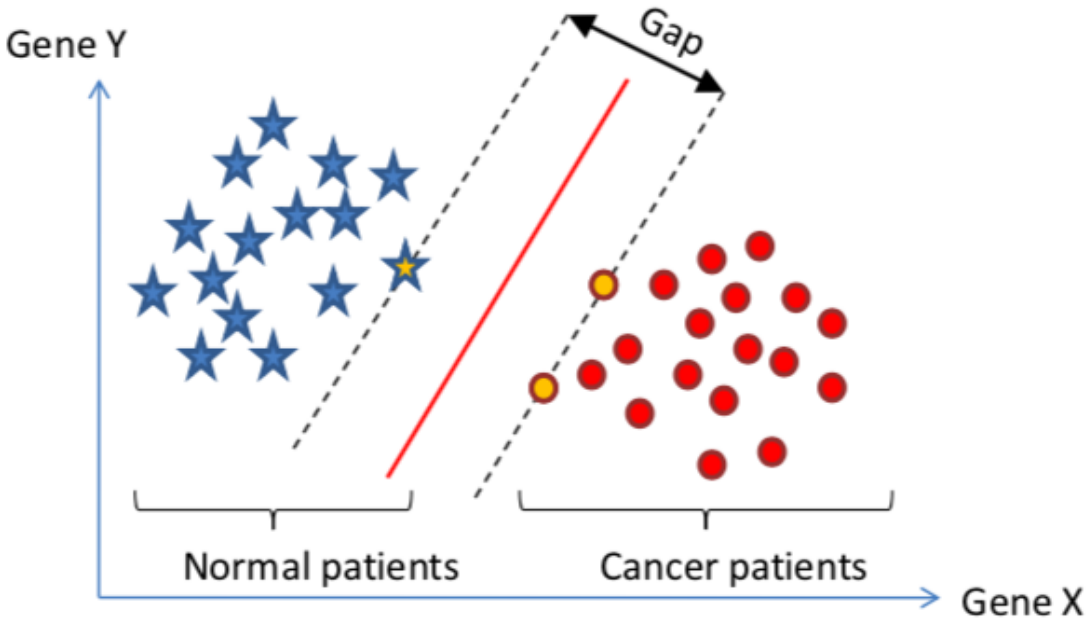
This course: 2, 4 and perhaps a bit about 1, 3

Linear Separators



- Consider example dataset described by 2 genes, gene X and gene Y
- Represent patients geometrically (by “vectors”)

Linear Separators



- Find a linear decision surface ("hyperplane") that can separate patient classes and has the largest distance (i.e., largest "gap" or "margin") between border-line patients (i.e., "support vectors");

Hyperplanes

Recall: an affine hyperplane $H \subset \mathbb{R}^d$ can be defined as

$$H = \{\mathbf{x} \in \mathbb{R}^d : \mathbf{x} \cdot \mathbf{w} + b = 0 \text{ for some } b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^d\}. \tag{3.1}$$

In fact, any two points $\mathbf{x}^1, \mathbf{x}^2$ belong to H iff $(\mathbf{x}^1 - \mathbf{x}^2) \cdot \mathbf{w} = 0$, i.e., their difference is perpendicular to the vector $\mathbf{w}$, i.e.,

$$H = H(\mathbf{w}, b) = \{\mathbf{x} \in \mathbb{R}^d : (\mathbf{x} - \mathbf{x}^0) \cdot \mathbf{w} = 0 \text{ for any fixed } \mathbf{x}^0 \in H\}, \quad \rightsquigarrow \tag{3.2}$$

Remark 1

(i) Varying b causes just a parallel shift of H. Also, given H, $\mathbf{w}$ *and b are only determined up a scalar multiple.*

(ii) For any $\mathbf{x}' \in \mathbb{R}^d$ *one has*

$$\operatorname{dist}(H, \mathbf{x}') := \inf_{\mathbf{x} \in H} \|\mathbf{x}' - \mathbf{x}\|_2 = \frac{|\mathbf{w} \cdot \mathbf{x}' + b|}{\|\mathbf{w}\|_2}. \tag{3.3}$$

(iii) The distance $\operatorname{dist}(H, H')$ *between any two parallel hyperplanes* $H(\mathbf{w}, b), H(\mathbf{w}, b')$ *is*

$$\operatorname{dist}(H, H') = \frac{|b - b'|}{\|\mathbf{w}\|_2}. \tag{3.4}$$

Since H and H′ are parallel, we have $\operatorname{dist}(H, H') = \operatorname{dist}(H, \mathbf{x}')$ *for any* $\mathbf{x}' \in H'$*. Since then* $\mathbf{w} \cdot \mathbf{x}' = -b'$*, the claim follows directly from (ii),* (3.3)*.*

Hyperplanes some arguments...

(i) follows directly from the above orthogonality statement (3.2).

Regarding (ii), for any fixed $\mathbf{x}^0 \in H$, $H - \mathbf{x}^0 =: \mathbb{U}$ is a $(d-1)$-dimensional linear subspace of $\mathbb{R}^d$. By definition $\mathbb{U} = \{\mathbf{x} : \mathbf{x} \in \mathbb{R}^d, \mathbf{x} \cdot \mathbf{w} = 0\}$. Let $\mathbf{u}^1, \ldots, \mathbf{u}^{d-1} \subset \mathbb{R}^d$ be orthonormal vectors perpendicular to $\mathbf{w}$, i.e., $\mathbf{w} \cdot \mathbf{u}^i = 0$, $i = 1, \ldots d-1$. Then the $\mathbf{u}^i$ form an orthonormal basis for $\mathbb{U}$. Clearly

$$\operatorname{dist}(H, \mathbf{x}') = \operatorname{dist}(H - \mathbf{x}^0, \mathbf{x}' - \mathbf{x}^0) = \operatorname{dist}(\mathbb{U}, \mathbf{x}' - \mathbf{x}^0) = \|(\mathbf{x}' - \mathbf{x}^0) - P_{\mathbb{U}}(\mathbf{x}' - \mathbf{x}^0)\|_2,$$

where we have used the Projection Theorem 24, Lecture I. Moreover, by Lecture I, (5.26), page 47, we know that

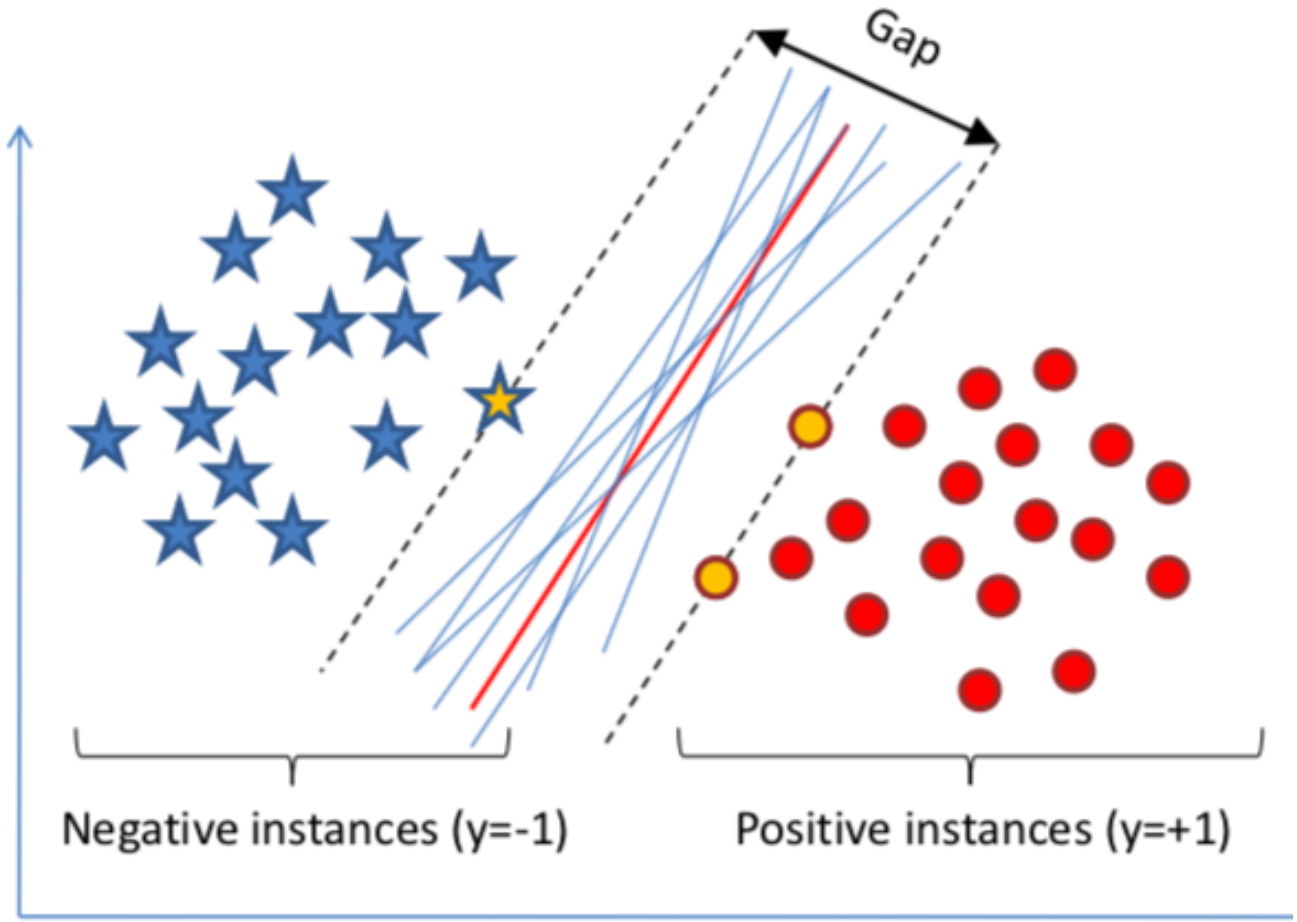
$$P_{\mathbb{U}}(\mathbf{x}' - \mathbf{x}^0) = \sum_{i=1}^{d-1}((\mathbf{x}' - \mathbf{x}^0) \cdot \mathbf{u}^i)\mathbf{u}^i, \quad (\mathbf{x}' - \mathbf{x}^0) = \frac{(\mathbf{x}' - \mathbf{x}^0) \cdot \mathbf{w}}{\|\mathbf{w}\|_2^2}\mathbf{w} + \sum_{i=1}^{d-1}((\mathbf{x}' - \mathbf{x}^0) \cdot \mathbf{u}^i)\mathbf{u}^i,$$

since $\mathbf{w}/\|\mathbf{w}\|_2, \mathbf{u}^1, \ldots, \mathbf{u}^{d-1}$ form an orthonormal basis for all of $\mathbb{R}^d$. Thus

$$\operatorname{dist}(H, \mathbf{x}') = \|(\mathbf{x}' - \mathbf{x}^0) - P_{\mathbb{U}}(\mathbf{x}' - \mathbf{x}^0)\|_2 = \left\| \frac{(\mathbf{x}' - \mathbf{x}^0) \cdot \mathbf{w}}{\|\mathbf{w}\|_2} \frac{\mathbf{w}}{\|\mathbf{w}\|_2} \right\|_2 = \frac{|\mathbf{w} \cdot (\mathbf{x}' - \mathbf{x}^0)|}{\|\mathbf{w}\|_2} = \frac{|\mathbf{w} \cdot \mathbf{x}' + b|}{\|\mathbf{w}\|_2}.$$

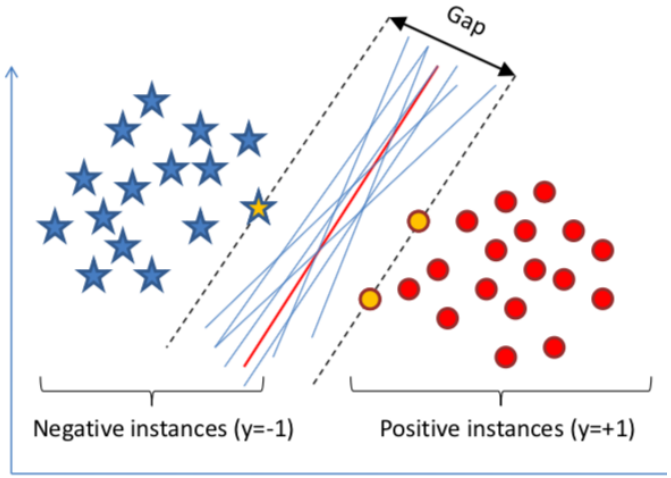
Warm Up - Linearly Separably Data

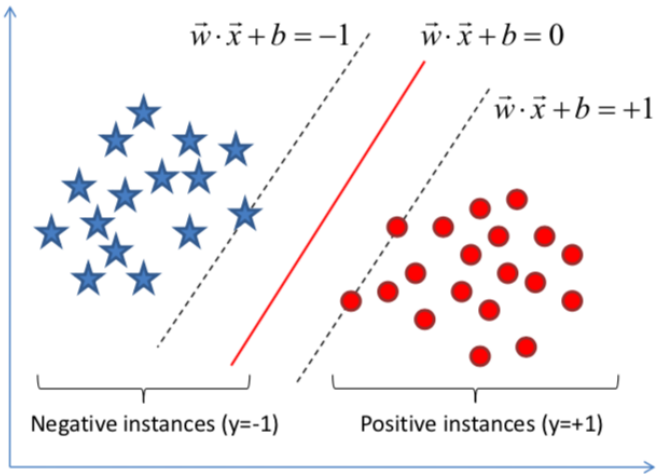
Assumption: there exists a hyperplane that separates data with positive and negative labels.



But: if data are linearly separable then there are generally infinitely many separating hyperplanes - how to compute a particularly “good ” one?

Warm Up - Linearly Separably Data

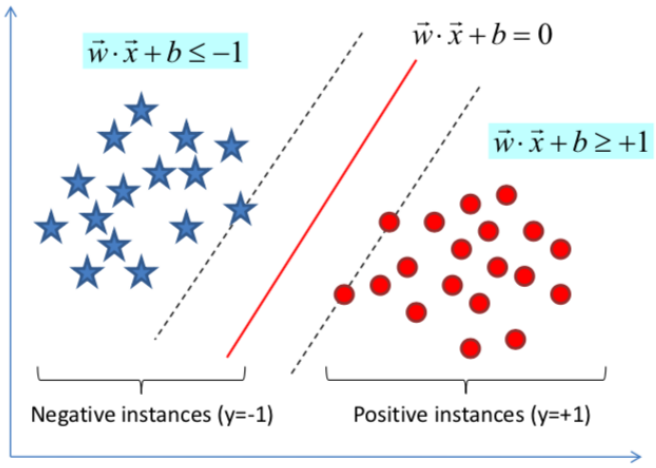




- The red hyperplane looks “good” because it maximizes the width of the margin (gap) between the two sets. We know it can be written as $H(\mathbf{w}, b)$ for some $\mathbf{w} \in \mathbb{R}^d$, $b \in \mathbb{R}$.
- $H(\mathbf{w}, b)$ is the zero set of the affine function $f(\mathbf{x}) := \mathbf{w} \cdot \mathbf{x} + b$. Since $\nabla f(\mathbf{x}) = \mathbf{w}$, the slope in direction $\mathbf{w}$ can be adjusted by scaling $\mathbf{w}$ and b which maintains the zero set $H(\mathbf{w}, b)$.
- The level sets $\{\mathbf{x} \in \mathbb{R}^d : f(\mathbf{x}) = \pm c\}$, any $c \in \mathbb{R}$, at equal distance from $H(\mathbf{w}, b)$, are the hyperplanes $\{\mathbf{x} \in \mathbb{R}^d : \mathbf{w} \cdot \mathbf{x} + b \mp c = 0\} = H(\mathbf{w}, b \mp c)$.
- $H(\mathbf{w}, b)$ is therefore optimally positioned if the points $(\mathbf{x}^{\pm}, \pm 1) \in \mathcal{Z}$ closest to $H(\mathbf{w}, b)$, belong to the graph of $\mathbf{w} \cdot \mathbf{x} + b$, i.e., $\mathbf{x}^{\pm} \in H(\mathbf{w}, b \mp 1)$. The $\mathbf{x}^{\pm}$ on the boundary of the separating margin are called support vectors.
- By Remark 1, width of the margin is

$$\frac{|(b-1)-(b+1)|}{\|\mathbf{w}\|_2} = \frac{2}{\|\mathbf{w}\|_2}. \tag{3.5}$$

Linear Support Vector Machine (SVM) Classifiers



The process of determining the margin maximizing hyperplane $H(\mathbf{w}, b)$ is called linear Support Vector Machine (SVM). The parameters $\mathbf{w} \in \mathbb{R}^d$, $b \in \mathbb{R}$ have the property that for $(\mathbf{x}^i, y_i) \in \mathcal{Z}$

$$\left.\begin{array}{lll} \mathbf{w}\cdot\mathbf{x}^i + b \geq 1 & \text{if} & y_i = +1 \\ \mathbf{w}\cdot\mathbf{x}^i + b \leq -1 & \text{if} & y_i = -1, \end{array}\right\} \Leftrightarrow\ y_i(\mathbf{w}\cdot\mathbf{x}^i + b) \geq 1. \tag{3.6}$$

By (3.5), the smaller $\|\mathbf{w}\|_2$ the wider is the margin !

In summary: solve the constrained optimization problem

$$\text{minimize over } \mathbf{w} \in \mathbb{R}^d,\ b \in \mathbb{R}: \quad \frac{1}{2}\|\mathbf{w}\|_2^2 \text{ subject to } y_i(\mathbf{w}\cdot\mathbf{x}^i + b) \geq 1,\ i = 1, \ldots, N. \tag{3.7}$$

The classifier

$$h : \mathbf{x} \mapsto h(\mathbf{x}) := \operatorname{sign}(\mathbf{w}\cdot\mathbf{x} + b) \tag{3.8}$$

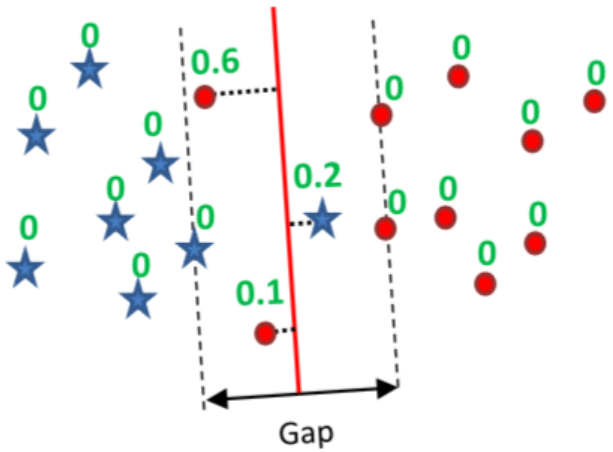
is called hard-margin linear SVM.

Remark 2

The problem (3.7) *requires minimizing a quadratic functional under linear constraints which is still a convex optimization problem. Efficient methods will be discussed later.*

Soft-Margin Linear SVM

Observations and measurements are viewed as random samples from some underlying distribution to account for measurement errors, i.e., the observed labels do not represent certain information but random events. Hence, even if a sample set $\mathfrak{Z} = \{(\mathbf{x}^1, y_1), \ldots, (\mathbf{x}^N, y_N)\}$ may not be able to be linearly separated by a hyperplane, they may be nearly so. In such a case it may still be reasonable to use a linear SVM. We discuss next how to deal with unavoidable training errors.



Instead of minimizing $\frac{1}{2}\|\mathbf{w}\|_2^2$ subject to the constraints $y_j(\mathbf{w} \cdot \mathbf{x}^j + b) \geq 1$, $j = 1, \ldots, N$, (see (3.7)), we relax the constraints as follows:

Approach: To each $j = 1, \ldots, N$ assign a slack variable $\xi_j \geq 0$, $\xi = (\xi_1, \ldots, \xi_N) \in \mathbb{R}_+^N$, $\rightsquigarrow$

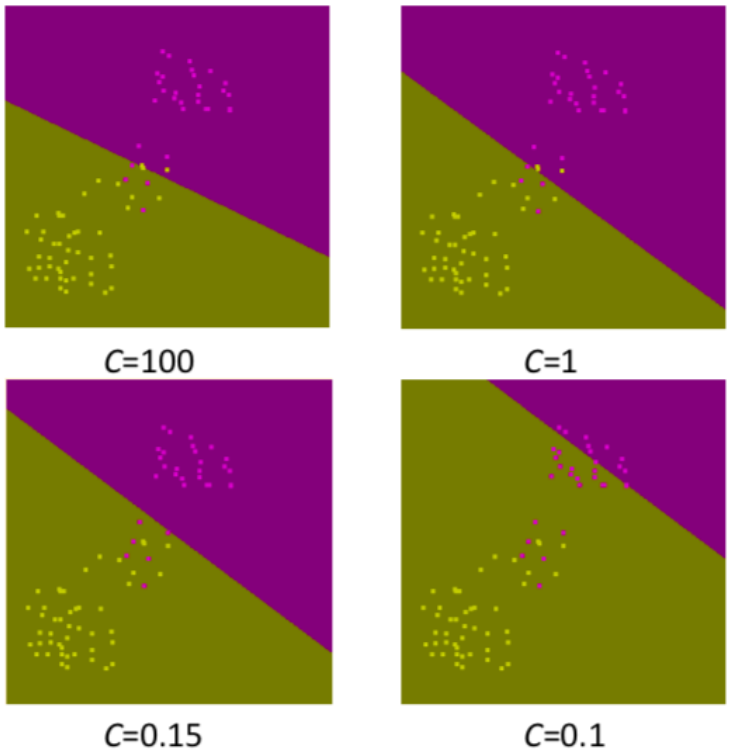
$$\text{minimize}_{\mathbf{w},b,\xi} \quad \underbrace{\frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{j=1}^{N}\xi_j}_{\text{objective functional}} \quad \text{subject to} \quad \underbrace{y_j(\mathbf{w}\cdot\mathbf{x}+b) \geq 1-\xi_j \quad , \; 1 \leq j \leq N,}_{\text{constraints}} \qquad (3.9)$$

$\rightsquigarrow \hat{\mathbf{w}}$, $\hat{b}$, take again $h(\mathbf{x}) := \operatorname{sgn}(\hat{\mathbf{w}} \cdot \mathbf{x} + \hat{b})$.

This is still a quadratic optimization problem under linear constraints, more about solvability later.

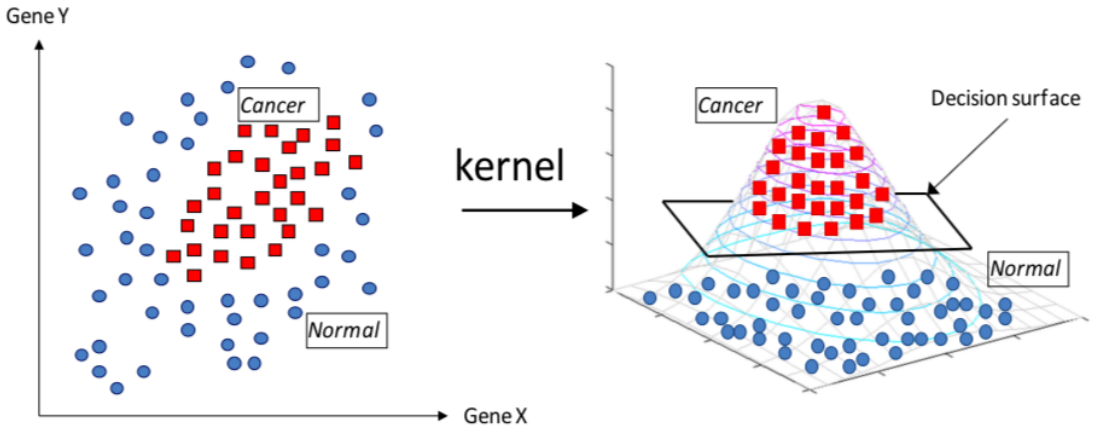
Soft-Margin Linear SVM

minimize over $\mathbf{w}, b$: $\frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{j=1}^{N}\xi_j$ subject to $y_j(\mathbf{w}\cdot\mathbf{x}^j + b) \geq 1 - \xi_j,\ 1 \leq j \leq N.$



- C very large: this is very close to hard-margin SVM;
- C very small: one achieves a small $\|bw\|_2$ and hence a wide margin (see (3.5)) but at the expense of many misclassifications.
- The proper choice of C depends on the problem and will be discussed later. It can be seen as Model Selection.

What if there is no reasonable linear separation? ...



- Find a function $f : \mathcal{X} \to \mathbb{R}$, taking positive values at data with positive labels and negative values at data with negative labels;
- Find the decision boundary $\{\mathbf{x} \in \mathbb{R}^d : f(\mathbf{x}) = 0\}$. The classifier $h : \mathcal{X} \to \{\pm 1\}$ is then defined as

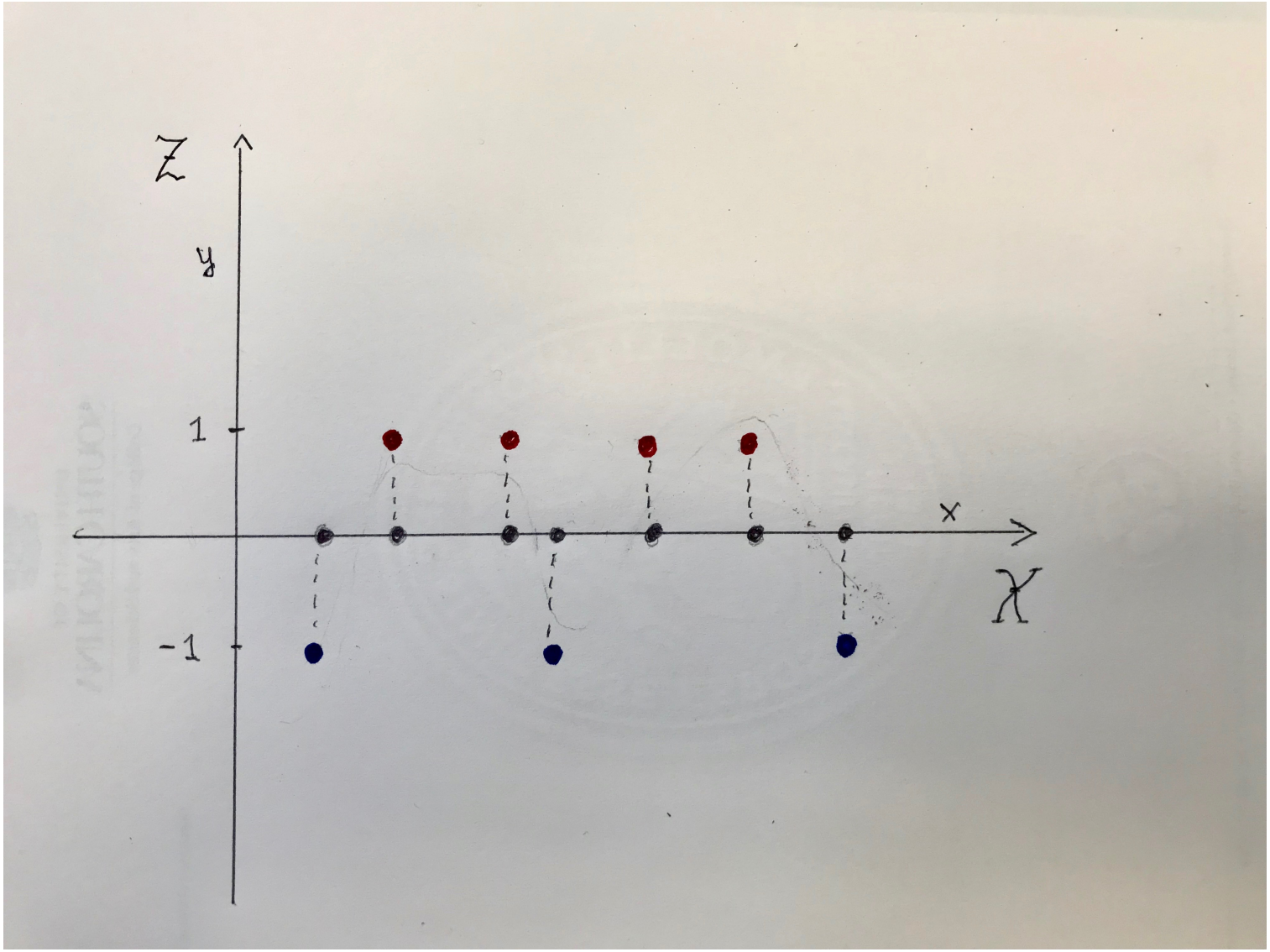
$$h(\mathbf{x}) = \left\{ \begin{array}{rll} 1 & \text{if} & f(\mathbf{x}) > 0, \\ -1 & \text{if} & f(\mathbf{x}) < 0. \end{array} \right\} = \operatorname{sgn}(f(\mathbf{x})) \qquad \text{Plug-in-Estimator.} \qquad (3.10)$$

  This corresponds to a linear separation in the higher dimensional space $\mathbb{R}^{d+1}$.
- When f has the special form

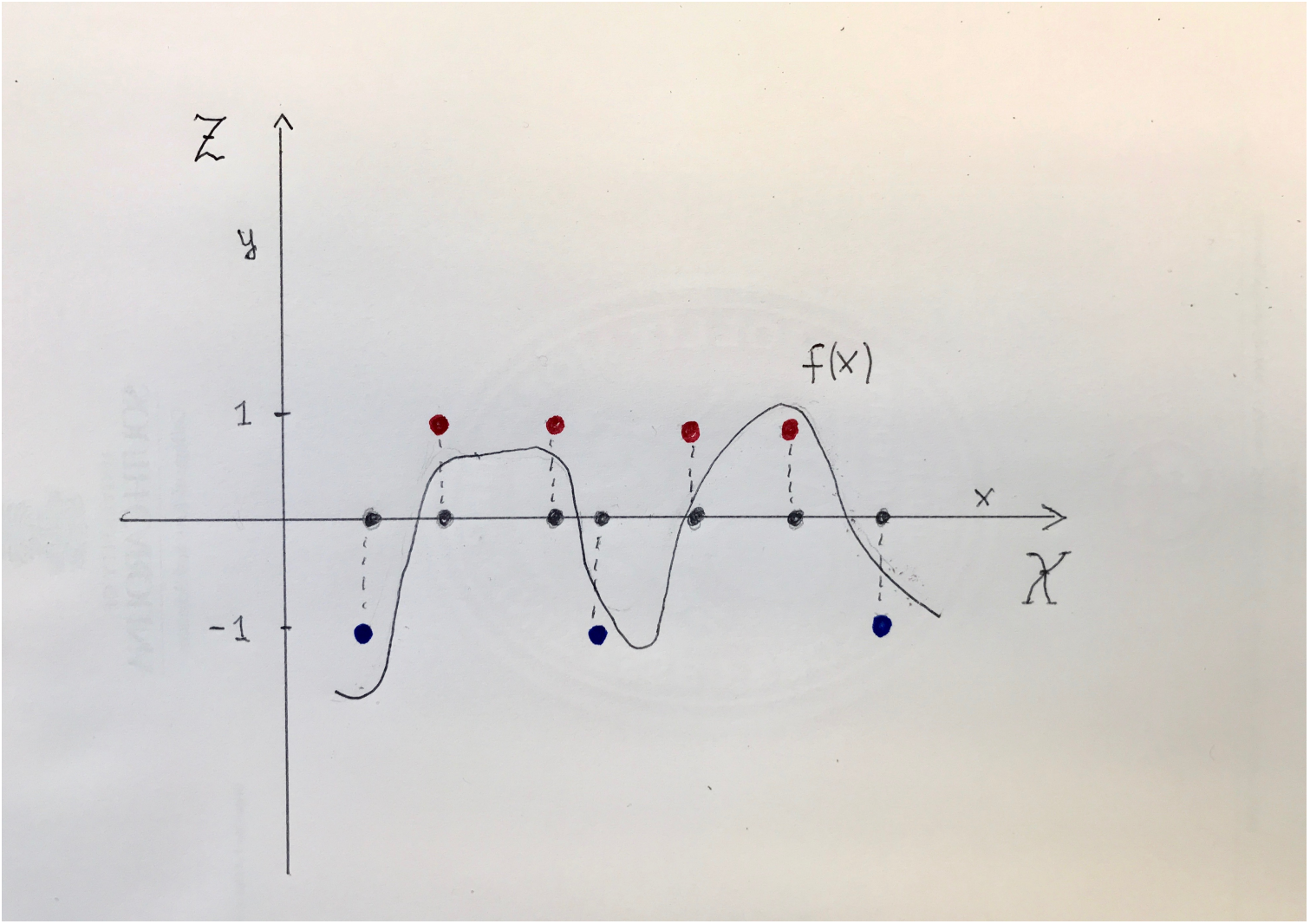
$$f(\mathbf{x}) = \mathbf{w} \cdot \Phi(\mathbf{x}) + b, \quad b \in \mathbb{R}, \mathbf{w} \in \mathbb{R}^{\bar{d}}, \quad \Phi : \mathbb{R}^d \to \mathbb{R}^{\bar{d}}, \quad \bar{d} \text{ possibly larger than } d, \quad (3.11)$$

  ($\mathbf{x}$ is replaced by $\Phi(\mathbf{x})$) this leads to the concept of Support Vector Machines (SVM).

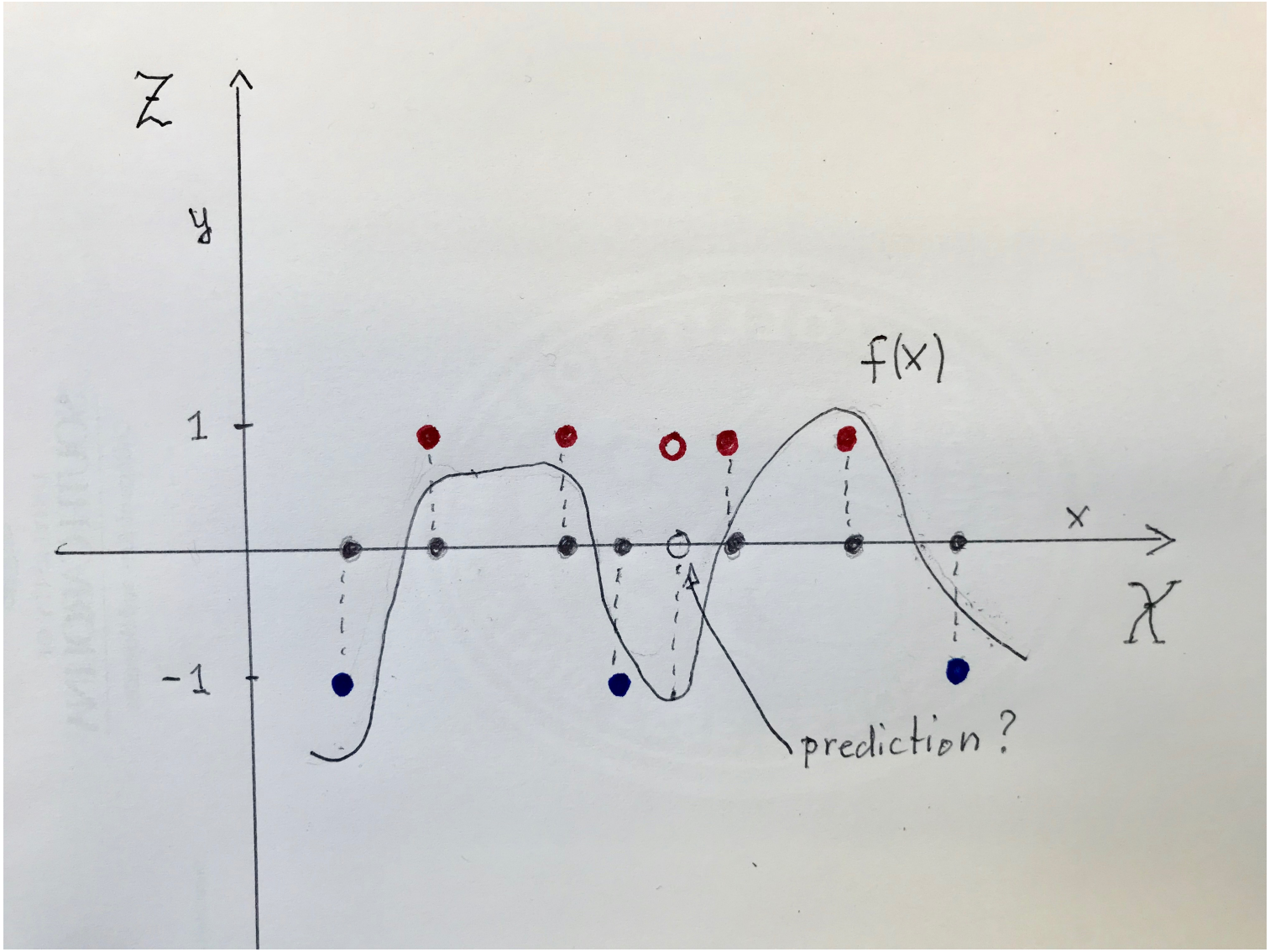
What if things are really mixed up? ...



What if things are really mixed up? ...



What if things are really mixed up? ...



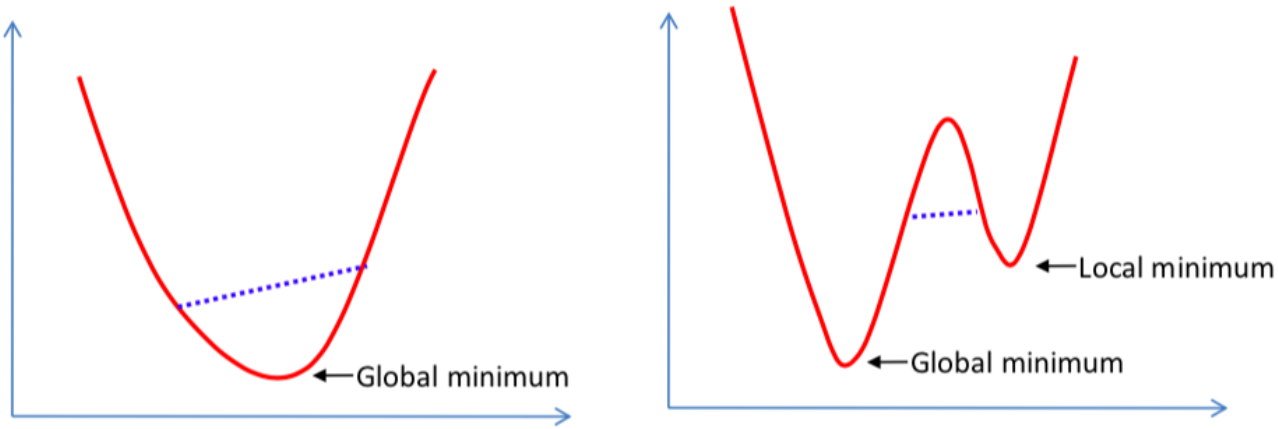
What does this tell us? ...

1. Data contain outliers and reflect only statistical information about an underlying process.
2. Building classifiers that are exact on the training data may become very complex.
3. It may also cause overfitting which, in turn, may result in poor generalization properties, i.e., due to increased variance, the ability to predict well on new samples.
4. One should therefore not insist on classifying the training data exactly.
5. Instead one should seek for a proper balance between exactness on training data and generalization, i.e., the ability to predict well on new samples.
6. The main strategies for arriving at such a balance are:

   (a) Complexity Penalization the classifier is constructed as the solution of an optimization problem where the objective functional consists of a data-fit-term and a penalty term that controls the complexity of the estimator; see the soft-margin linear SVM (3.9).

   (b) Model Selection: one builds several classifiers subject to different design parameters and “compares”.

Convexity



- A function $f : \mathcal{X} \to \mathbb{R}$ is called convex if for any $\mathbf{x}, \mathbf{v} \in \mathcal{X}$ one has

$$f\Big(\frac{1}{2}(\mathbf{x}+\mathbf{v})\Big) \leq \frac{1}{2}\big(f(\mathbf{x})+f(\mathbf{v})\big),$$

  i.e., the value of f at the average of two points is always bounded from above by the average of the values of f Exercise: show that this is equivalent to the statement: for any $\mathbf{x}^1, \ldots, \mathbf{x}^n \in \mathcal{X}$, $\lambda_1, \ldots, \lambda_n \geq 0$ such that $\lambda_1 + \cdots + \lambda_n = 1$ one has $f\big(\lambda_1 \mathbf{x}^1 + \cdots + \lambda_n \mathbf{x}^n\big) \leq \lambda_1 f(\mathbf{x}^1) + \cdots + \lambda_n f(\mathbf{x}^n)$.

- Property: any local minimum of a convex function is also a global minimum.

QP-Problems

Recall from (3.9): linear soft-margin SVM leads to (given the data $(\mathbf{x}^j, y_j)$)

$$\text{minimize}_{\mathbf{w},b,\xi} \quad \underbrace{\frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{j=1}^{N}\xi_j}_{\text{objective}} \quad \text{subject to} \quad \underbrace{y_j(\mathbf{w}\cdot\mathbf{x}^j + b) \geq 1 - \xi_j \quad , \; 1 \leq j \leq N,}_{\text{constraints}}.$$

- Since $\frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{j=1}^{N}\xi_j$ the objective is a quadratic function in $(\mathbf{w}, b, \xi)$.
- The constraints are linear.
- Such optimization problems are called Convex Quadratic Programming (QP) problems.
- QP-problems can be solved by descent or greedy methods that decrease the objective in an iterative fashion.

Remark 3

For zero slacks, i.e., $\xi_j = 0$, $j = 1, \ldots, N$, the problem has the form

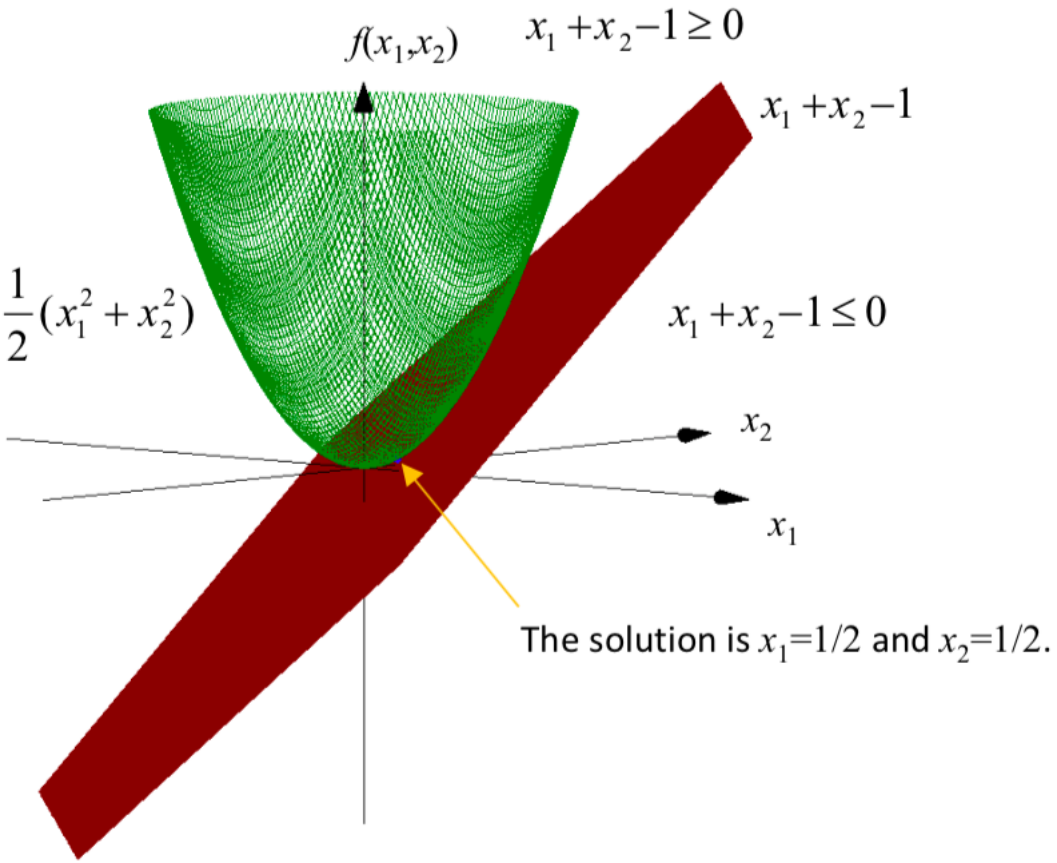
$$\textit{minimize}_{\mathbf{x}\in\mathbb{R}^d} \quad \frac{1}{2}\mathbf{x}^\top \boldsymbol{A}\mathbf{x} + \mathbf{c}^\top\mathbf{x} \quad \textit{subject to} \quad (\mathbf{B}\mathbf{x} - \mathbf{b}) \geq \mathbf{0} \; (\textit{componentwise}) \tag{3.12}$$

where above $\boldsymbol{A}_{\leq d,\leq d} = \mathbf{I} \in \mathbb{R}^{d\times d}$, $\mathbf{B} \in \mathbb{R}^{N\times d}$ has rows $y_j\mathbf{x}^j$, $j = 1, \ldots, N$, $\mathbf{b} \in \mathbb{R}^N$ has components $b_j := y_j b - 1$, $j = 1, \ldots, N$, $\mathbf{c} = \mathbf{0}$.

Model Problem

Consider the case $d = 1$ ($\mathbf{w} \leftrightarrow \mathbf{x}$)

$$\text{minimize over } \mathbf{x}: \quad \frac{1}{2}(x_1^2 + x_2^2) \quad \text{subject to} \quad x_1 + x_2 - 1 \geq 0.$$



Lagrange Multipliers A Short Digression

Optimization with equality constraints: $F : \mathbb{R}^d \to \mathbb{R}$ convex, $g : \mathbb{R}^d \to \mathbb{R}$

$$\text{Minimize}_{\mathbf{x}\in\mathbb{R}^d} \quad F(\mathbf{x}) \quad \text{subject to} \quad g(\mathbf{x}) = 0. \tag{3.13}$$

Idea: find necessary conditions for $\mathbf{x}$ to be a local constrained minimum:

- Recall that the direction of steepest descent of a function F at a point $\mathbf{x}$ is $-\nabla_{\mathbf{x}} F(\mathbf{x})$.
- Suppose $\bar{\mathbf{x}}$ is a feasible point, i.e., $g(\bar{\mathbf{x}}) = 0$. For $F(\bar{x} + t\mathbf{z}) < F(\bar{\mathbf{z}})$ for some $t > 0$ and some direction $\mathbf{z}$, one must have

$$\nabla F(\bar{\mathbf{x}}) \cdot \mathbf{z} < 0, \quad \text{(why?)} \tag{3.14}$$

- For $\bar{\mathbf{x}} + \mathbf{z}$ to be tangential to the constraint surface, $\mathbf{z}$ must be perpendicular to the constraint surface, i.e.,

$$\mathbf{z} \cdot \nabla_{\mathbf{x}} g(\bar{\mathbf{x}}) = 0. \tag{3.15}$$

- If $\nabla_{\mathbf{x}} F(\bar{\mathbf{x}})$ and $\nabla_{\mathbf{x}} g(\bar{\mathbf{x}})$ point in the same direction, i.e.,

$$\nabla_{\mathbf{x}} F(\bar{\mathbf{x}}) = -a \nabla_{\mathbf{x}} g(\bar{\mathbf{x}}) \quad \text{for some } a \in \mathbb{R} \tag{3.16}$$

one cannot satisfy (3.14) and (3.15) simultaneously. Hence, one cannot decrease $F(\bar{\mathbf{x}})$ any further while staying feasible.

Lagrange Multipliers A Short Digression

Conclusion: A necessary condition for $\bar{\mathbf{x}}$ to be a minimizer of

$$\text{Minimize}_{\mathbf{x}\in\mathbb{R}^d} \quad F(\mathbf{x}) \quad \text{subject to} \quad g(\mathbf{x}) = 0$$

is in view of (3.16): $\exists\, \bar{a} \in \mathbb{R}$ s.t.

$$\nabla_{\mathbf{x}} F(\bar{\mathbf{x}}) + \bar{a}\nabla_{\mathbf{x}} g(\bar{\mathbf{x}}) = 0 \quad \text{and} \quad g(\bar{\mathbf{x}}) = 0 \ (\Leftrightarrow\ \partial_a(F(\mathbf{x}) + ag(\mathbf{x})) = 0). \tag{3.17}$$

By similar arguments one can show the following:

Proposition 4

Let $\mathbf{g} = (g_1, \ldots, g_m)^\top : \mathbb{R}^d \to \mathbb{R}^m$ *be vector-valued and define the Lagrangian* $L(\mathbf{x}, \mathbf{a}) := F(\mathbf{x}) + \mathbf{a} \cdot \mathbf{g}(\mathbf{x}) = F(\mathbf{x}) + \sum_{i=1}^m a_i g_i(\mathbf{x})$. *Then if* $\bar{x} \in \mathbb{R}^d$ *is a local minimizer of*

$$\textit{Minimize}_{\mathbf{x}\in\mathbb{R}^d} \quad F(\mathbf{x}) \quad \textit{subject to} \quad \mathbf{g}(\mathbf{x}) = 0, \tag{3.18}$$

there exists a $\bar{\mathbf{a}} \in \mathbb{R}^m$ *such that* $(\bar{\mathbf{x}}, \bar{\mathbf{a}}) \in \mathbb{R}^d \times \mathbb{R}^m$ *is a critical point of the Lagrangian, i.e.,*

$$\nabla_{\mathbf{x}} L(\bar{\mathbf{x}}, \bar{\mathbf{a}}) = 0 \quad \textit{and} \quad \nabla_{\mathbf{a}} L(\bar{\mathbf{x}}, \bar{\mathbf{a}}) = 0, \tag{3.19}$$

The vector $\mathbf{a} \in \mathbb{R}^m$ is called Lagrange Multiplier.

KKT-Conditions A Short Digression

Optimization with inequality constraints: $F : \mathbb{R}^d \to \mathbb{R}$ convex, $\mathbf{g} : \mathbb{R}^d \to \mathbb{R}^m$

$$\text{Minimize}_{\mathbf{x}\in\mathbb{R}^d} \quad F(\mathbf{x}) \quad \text{subject to} \quad \mathbf{g}(\mathbf{x}) \geq 0 \quad (\text{meaning } g_j(\mathbf{x}) \geq 0,\ j = 1,\ldots,m). \tag{3.20}$$

Theorem 5

Karush-Kuhn-Tucker Conditions: *Let* $L(\mathbf{x}, \mathbf{a}) := F(\mathbf{x}) + \mathbf{a} \cdot \mathbf{g}(\mathbf{x})$. *Then, if* $\bar{\mathbf{x}}$ *is a local minimum of* (3.20) *there exists a* $\bar{\mathbf{a}} \in \mathbb{R}^m$ *such that the KKT-conditions hold:*

$$\begin{aligned}
\nabla_{\mathbf{x}} L(\bar{\mathbf{x}}, \bar{\mathbf{a}}) &= 0 && && (3.21)\\
\bar{\mathbf{a}} &\geq \mathbf{0} && \textit{componentwise} && (3.22)\\
\bar{\mathbf{a}} \cdot \mathbf{g}(\bar{\mathbf{x}}) &= 0 && && (3.23)\\
\mathbf{g}(\bar{\mathbf{x}}) &\geq \mathbf{0} && \textit{componentwise.} && (3.24)
\end{aligned}$$

Note: (3.22) - (3.24) *replace* $\nabla_{\mathbf{a}} L(\bar{\mathbf{x}}, \bar{\mathbf{a}}) = \mathbf{0}$.

Proof: Let $R := \{\mathbf{x} \in \mathbb{R}^d : \mathbf{g}(\mathbf{x}) > \mathbf{0}\}$ be the feasible region (constraints are satisfied).

Case 1: the unconstrained local minimum $\bar{\mathbf{x}}$ belongs to R, i.e., $g(\bar{\mathbf{x}}) > 0$. Then $\bar{\mathbf{x}}$ must be a critical point of F, i.e., $\nabla_{\mathbf{x}} F(\bar{\mathbf{x}}) = \mathbf{0}$ and (3.24) holds. Taking $\bar{\mathbf{a}} = \mathbf{0}$ satisfies (3.22) and (3.23). Because of $\bar{\mathbf{a}} = \mathbf{0}$ and $\nabla_{\mathbf{x}} F(\bar{\mathbf{x}}) = \mathbf{0}$ (3.21) follows. Hence the KKT-conditions hold at $\bar{\mathbf{x}}$.

KKT-Conditions A Short Digression

Proof of Theorem 5 continued: Consider now:

Case 2: The unconstrained local minimum lies outside the feasible region R. Consider first the case $m = 1$, i.e., $g : \mathbb{R}^d \to \mathbb{R}$, the case $m > 1$ follows the same ideas and will be sketched later.

Then $\bar{\mathbf{x}} \in \partial R$, i.e., $g(\bar{\mathbf{x}}) = 0$ which is (3.23) and (3.24). Thus, we have a minimization problem with one equality constraint.

By the previous reasoning we must therefore have that $\nabla_{\mathbf{x}} F(\bar{\mathbf{x}}) = -\bar{a}\nabla_{\mathbf{x}} g(\bar{\mathbf{x}})$ for some $\bar{a} \in \mathbb{R}$ which is (3.21). Also the direction of steepest descent of F at $\bar{\mathbf{x}}$ must point outside the feasible reagion. Since $g(\mathbf{x}) > 0$ inside R, g grows when moving inside the feasible reagion, i.e., the direction of steepest descent $-\nabla_{\mathbf{x}} g(\bar{\mathbf{x}})$ must also point outside R. Hence, we must have $\bar{a} > 0$ which is (3.22). This completes the argument for $m = 1$.

Now consider $m > 1$. Again, we know that the constrained minimizer must lie on the boundary ∂R of the feasible region R, i.e., $\bar{\mathbf{x}} \in \partial R$.

The argument is slightly more complicated because the feasible region $R := \{\mathbf{x} \in \mathbb{R}^d : \mathbf{g}(\mathbf{x}) > \mathbf{0}\} = \{\mathbf{x} \in \mathbb{R}^d : g_j(\mathbf{x}) > 0,\ j = 1, \ldots, m\}$ is slightly more complicated. In fact, defining the componentwise feasible regions

$$R_j := \{\mathbf{x} \in \mathbb{R}^d : g_j(\mathbf{x}) > 0\}, \quad j = 1, \ldots, m, \quad \text{one has} \quad R = \bigcap_{i=1}^{m} R_j.$$

KKT-Conditions A Short Digression

Proof of Theorem 5 continued: Hence, the boundary of R is more complicated

$$\partial R = \bigcup_{j=1}^{m} (\partial R_j \cap \overline{R}).$$

Suppose for a moment that the constrained minimizer $\bar{\mathbf{x}}$ belongs to just a singe boundary facet $\partial R_i \cap \overline{R}$ which means $g_i(\bar{\mathbf{x}}) = 0$, $g_l(\bar{\mathbf{x}}) > 0$ for all $l \neq i$. By the same argument as before we can conclude that there must exist a $\bar{a}_i > 0$ such that $\nabla_{\mathbf{x}} F(\bar{\mathbf{x}}) + \bar{a}_i \nabla_{\mathbf{x}} g_l(\bar{\mathbf{x}}) = \mathbf{0}$. If we set $\bar{a}_l = 0$, $l \neq i$, it follows that $\nabla_{\mathbf{x}} L(\bar{\mathbf{x}}, \bar{\mathbf{a}}) = \mathbf{0}$ which is (3.21), and (3.22) - (3.24) are obviously valid.

The case where the constrained minimizer $\bar{\mathbf{x}}$ belongs to the intersection of several facets, $\bigcap_{r=1}^{s} (\partial R_{i_r} \cap \overline{R})$, one cannot move in any of these facets to decrease F. This means the direction $-\nabla_{\mathbf{x}} F(\bar{\mathbf{x}})$ of steepest descent must point into the cone formed as the intersection of half spaces defined by the tangent planes to R at $\bar{\mathbf{x}}$. This means $-\nabla_{\mathbf{x}} F(\bar{\mathbf{x}})$ must be a linear combination

$$-\nabla_{\mathbf{x}} F(\bar{\mathbf{x}}) = \sum_{r=1}^{s} \bar{a}_{i_r} \nabla_{\mathbf{x}} g_{i_r}(\bar{\mathbf{x}}), \quad \bar{a}_{i_r} \geq 0,\ r = 1, \ldots, s.$$

Setting again $\bar{a}_l = 0$, for $l \notin \{i_1, \ldots, i_s\}$, it follows that (3.21) - (3.24) hold for $\bar{\mathbf{x}}, \bar{\mathbf{a}}$. This finishes the proof. □

Active Constraints

Remark 6

The above discussion can be summarized as follows:

- *The ith constraint is called active at $\bar{\mathbf{x}}$ if $g_i(\bar{\mathbf{x}}) = 0$, i.e., $\bar{\mathbf{x}}$ lies on ∂R_j.*
- *For $\bar{\mathbf{x}}$ to belong to the boundary ∂R of the feasible region R, at least one constraint has to be active.*
- *If the ith constraint is active the above reasoning showed that $\nabla_{\mathbf{x}} F(\bar{\mathbf{x}}) = -\bar{a}_i \nabla_{\mathbf{x}} g_i(\bar{\mathbf{x}})$ for some positive $\bar{a}_i$.*
- *If the jth constraint is inactive, i.e., $g_j(\bar{\mathbf{x}}) > 0$, the two KKT-conditions (3.22), (3.23) just say that $\bar{a}_j = 0$.*

Hence, the support $\operatorname{supp}(\bar{\mathbf{a}}) := \{i : \bar{a}_i \neq 0\}$ identifies the active constraints at a local minimum $\bar{\mathbf{x}}$, i.e., when $\bar{\mathbf{x}}, \bar{a}$ satisfy the KKT-conditions, then

$$\{i \in \{1, \ldots, m\} : p_i(\bar{\mathbf{x}}) = 0 \text{ (} i \text{ is active)}\} = \operatorname{supp}(\bar{\mathbf{a}}). \tag{3.25}$$

KKT-Conditions A Short Digression

We specialize this to a quadratic problem: Let $\boldsymbol{A} \in \mathbb{R}^{d\times d}$ be symmetric positive semi-definite and $\mathbf{B} \in \mathbb{R}^{N\times d}$, $\mathbf{b} \in \mathbb{R}^N$, see Remark 3. Consider

$$\text{minimize}_{\mathbf{x}\in\mathbb{R}^d} \quad \underbrace{\frac{1}{2}\mathbf{x}^\top \boldsymbol{A}\mathbf{x} + \mathbf{c}^\top \mathbf{x}}_{=:F(\mathbf{x})} \quad \text{subject to} \quad \underbrace{\mathbf{B}\mathbf{x} - \mathbf{b}}_{=:\mathbf{g}(\mathbf{x})} \geq \mathbf{0}. \tag{3.26}$$

Remark 7

Since If $F(\mathbf{x}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{A}\mathbf{x}$ is convex when $\boldsymbol{A}$ is symmetric positive definite, and since the feasible region for $\mathbf{g}(\mathbf{x}) = \mathbf{B}\mathbf{x} - \mathbf{b}$ is the intersection of half-spaces and hence again convex, the constrained minimization problem has a solution. By convexity each local solution is a global solution which must satisfy the KKT-conditions. Thus, $\bar{\mathbf{x}}$ is a constrained minimizer if and only if the KKT-conditions hold.

The Lagrangian reads

$$L(\mathbf{x}, \mathbf{a}) = \frac{1}{2}\mathbf{x}^\top \boldsymbol{A}\mathbf{x} + \mathbf{c}^\top \mathbf{x} + \mathbf{a}^\top(\mathbf{B}\mathbf{x} - \mathbf{b}), \qquad \mathbf{x} \in \mathbb{R}^d,\ \mathbf{a} \in \mathbb{R}^N. \tag{3.27}$$

The KKT-conditions (3.21) - (3.24) are

$$\nabla_{\mathbf{x}} L(\bar{\mathbf{x}}, \bar{\mathbf{a}}) = \boldsymbol{A}\bar{\mathbf{x}} + \mathbf{B}^\top \bar{\mathbf{a}} = \mathbf{0}, \quad \bar{\mathbf{a}}^\top(\mathbf{B}\bar{\mathbf{x}} - \mathbf{b}) = 0, \quad \bar{\mathbf{a}} \geq \mathbf{0}, \quad \mathbf{B}\bar{\mathbf{x}} \geq \mathbf{b}. \tag{3.28}$$

Saddle Points A Geometric Interpretation

Suppose for a moment that we have only equality constraints $\mathbf{Bx} - \mathbf{b} = \mathbf{0}$. Then, Proposition 4 (see (3.19)) says that the KKT-conditions (3.28) reduce to

$$\boldsymbol{A}\bar{\mathbf{x}} + \mathbf{B}^\top \bar{\mathbf{a}} = 0, \quad \mathbf{B}\bar{\mathbf{x}} = \mathbf{b}, \tag{3.29}$$

or equivalently

$$\nabla_{\mathbf{x},\mathbf{a}} L(\mathbf{x}, \mathbf{a}) = \begin{pmatrix} \boldsymbol{A} & \mathbf{B}^\top \\ \mathbf{B} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{a} \end{pmatrix} - \begin{pmatrix} \mathbf{0} \\ \mathbf{b} \end{pmatrix} = \mathbf{0} \tag{3.30}$$

Thus, the Hessian of $L(\mathbf{x}, \mathbf{a})$ is given by

$$D^2 L(\mathbf{x}, \mathbf{a}) = \begin{pmatrix} \boldsymbol{A} & \mathbf{B}^\top \\ \mathbf{B} & \mathbf{0} \end{pmatrix}.$$

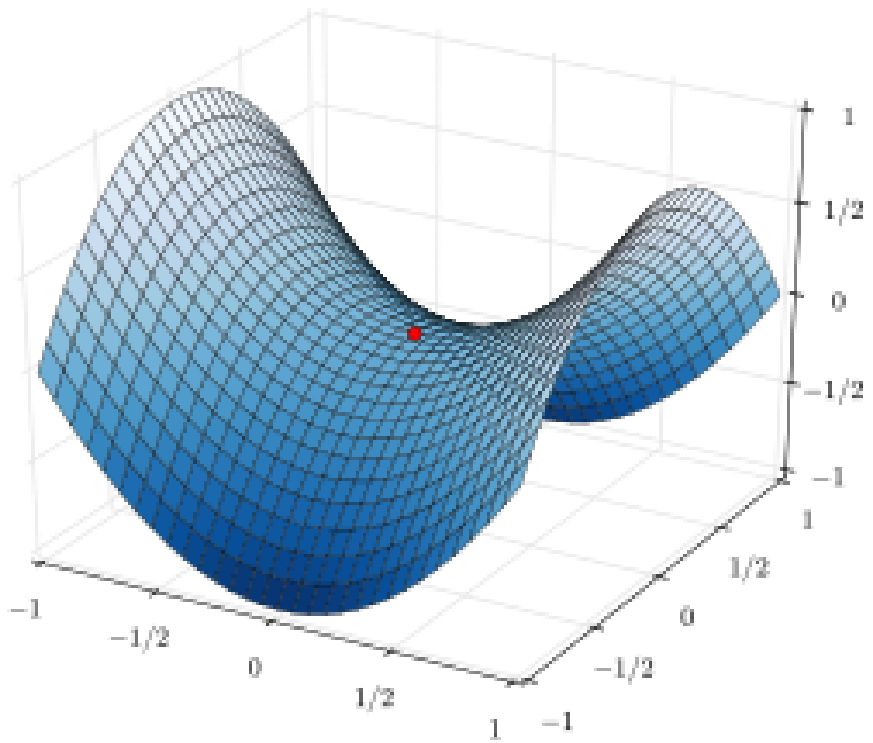
Remark 8

For $\boldsymbol{A}$ positive semi-definite the Hessian $D^2 L$ is in general indefinite, i.e., it has positive and negative eigenvalues (by symmetry all eigenvalues are real, see Lecture I). This is easily confirmed when $A = a > 0$, $B = b \in \mathbb{R} \setminus \{0\}$ since the determinant is then $-b^2 < 0$.

Saddle Points

By Remark 8, the critical point $(\bar{\mathbf{x}}, \bar{\mathbf{a}})$ of $L(\mathbf{x}, \mathbf{a})$ satisfying (3.30), cannot be a minimum of L. It is a saddle point

$$L(\bar{\mathbf{x}}, \bar{\mathbf{a}}) = \inf_{\mathbf{x} \in \mathbb{R}^d} L(\mathbf{x}, \bar{\mathbf{a}}) = \sup_{\mathbf{a} \in \mathbb{R}^N} L(\bar{\mathbf{x}}, \mathbf{a}) = \sup_{\mathbf{a} \in \mathbb{R}^N} \inf_{\mathbf{x} \in \mathbb{R}^d} L(\mathbf{x}, \mathbf{a}). \tag{3.31}$$



The Dual Problem equality constraints

An important strategy in quadratic programming is the passage to the so called dual problem which is to solve the right most version (3.31), i.e.,

$$H(\mathbf{a}) := \inf_{\mathbf{x}\in\mathbb{R}^d} L(\mathbf{x},\mathbf{a}) \quad \rightsquigarrow \quad \sup_{\mathbf{a}\in\mathbb{R}^N} H(\mathbf{a}). \tag{3.32}$$

$H(\mathbf{a})$ is often called the dual function.

Remark 9

Notice that (3.32) *is now an unconstrained optimization problem. Here this is, however, a consequence of the fact that we had only equality constraints and changes slightly when dealing with inequality constraints. - But: how to compute* $H(\mathbf{a})$*?*

Idea: Assume for a moment that $\boldsymbol{A}$ is non-singular.

1. For fixed $\mathbf{a}$ the Lagrangian $L(\mathbf{x},\mathbf{a})$ as a function of $\mathbf{x}$ is convex. Therefore, its (unconstrained) minimum is a critical point $\nabla_{\mathbf{x}} L(\mathbf{x},\mathbf{a}) = \mathbf{0}$ which is the first KKT-condition (3.21).
2. Eliminate $\mathbf{x}$ from the first KKT-condition (3.21) $\rightsquigarrow$ $\mathbf{x} = -\boldsymbol{A}^{-1}\mathbf{B}^\top\mathbf{a}$ and substitute this into $L(\mathbf{x},\mathbf{a})$.
3. This yields
$$\mathbf{x}^\top\boldsymbol{A}\mathbf{x} = \mathbf{a}^\top\mathbf{B}\boldsymbol{A}^{-1}\boldsymbol{A}\boldsymbol{A}^{-1}\mathbf{B}^\top\mathbf{a} = \mathbf{a}^\top\underline{\mathbf{B}}\boldsymbol{A}^{-1}\mathbf{B}^\top\mathbf{a}, \quad \mathbf{a}^\top(\mathbf{B}\mathbf{x}-\mathbf{b}) = -\mathbf{a}^\top\underline{\mathbf{B}}\boldsymbol{A}^{-1}\mathbf{B}^\top\mathbf{a} - \mathbf{a}^\top\mathbf{b}$$
which gives
$$H(\mathbf{a}) = -\frac{1}{2}\mathbf{a}^\top\underline{\mathbf{B}}\boldsymbol{A}^{-1}\mathbf{B}^\top\mathbf{a} - \mathbf{a}^\top\mathbf{b}. \tag{3.33}$$

The Dual Problem equality constraints

$$\text{maximize}_{\mathbf{a}} \quad H(\mathbf{a}) = -\frac{1}{2}\mathbf{a}^\top \mathbf{B}\boldsymbol{A}^{-1}\mathbf{B}^\top \mathbf{a} - \mathbf{a}^\top \mathbf{b}$$

$$\Leftrightarrow$$

$$\text{Minimize}_{\mathbf{a}\in\mathbb{R}^m} \quad \frac{1}{2}\mathbf{a}^\top \mathbf{M}\mathbf{a} + \mathbf{a}^\top \mathbf{b}, \quad \text{where} \quad \mathbf{M} := \mathbf{B}\boldsymbol{A}^{-1}\mathbf{B}^\top \text{ is symm. pos. def.} \tag{3.34}$$

$$\Leftrightarrow$$

$$\mathbf{0} = \nabla H(\bar{\mathbf{a}}) = \mathbf{M}\bar{\mathbf{a}} + \mathbf{b} \quad \Leftrightarrow \quad \mathbf{M}\bar{\mathbf{a}} = -\mathbf{b} \quad \rightsquigarrow \quad \bar{\mathbf{x}} \overset{(3.29)}{=} -\boldsymbol{A}^{-1}\mathbf{B}^\top \bar{\mathbf{a}}. \tag{3.35}$$

Remark 10

- *Thus, if one only has equality constraints, passing to the dual formulation, reduces the QP problem to solving a linear positive definite system of equations.*
- *We have assumed that* $\det \boldsymbol{A} \neq 0$ *to form the matrix* $\mathbf{M}$*. If* $\boldsymbol{A}$ *is singular the system*

$$\nabla_{\mathbf{x}} L(\mathbf{x}, \mathbf{a}) = \mathbf{0} \quad \overset{(3.27)}{\Leftrightarrow} \quad \boldsymbol{A}\mathbf{x} = -\mathbf{B}^\top \mathbf{a},$$

has more than one solution (infinitely many in this case, but it always has at least one). We can solve then instead the least squares problems

$$\min_{\mathbf{a}} \|\mathbf{M}\mathbf{a} + \mathbf{b}\|_2 \to \mathbf{a}, \quad \min_{\mathbf{x}} \|\boldsymbol{A}\mathbf{x} + \mathbf{B}^\top \mathbf{a}\|_2^2 \quad (\textit{e.g. by SVD})$$

The Dual Problem inequality constraints

Back to inequality constraints (3.26): $\text{minimize}_{\mathbf{x}\in\mathbb{R}^d} \quad \frac{1}{2}\mathbf{x}^\top \boldsymbol{A}\mathbf{x} \quad \text{subject to} \quad \mathbf{B}\mathbf{x}-\mathbf{b}\geq \mathbf{0}.$

The associate dual problem now reads (with $H(\mathbf{a})$ from (3.33))

$$\text{Maximize}_{\mathbf{a}} \quad H(\mathbf{a}) \quad \text{subsect to} \quad \mathbf{a}\geq \mathbf{0}. \tag{3.36}$$

Remark 11

- *Even if the objective function F is not convex, the dual function $H(\mathbf{a}) = \inf_{\mathbf{x}\in\mathbb{R}^d}\left\{F(\mathbf{x}) + \mathbf{a}^\top \mathbf{g}(\mathbf{x})\right\}$ as a pointwise limit is affine in $\mathbf{a}$ and hence convex. Hence it is in general easier to optimize.*
- *The value $\delta^* := \sup_{\mathbf{a}\in\mathbb{R}^d,\mathbf{a}\geq\mathbf{0}} H(\mathbf{a})$ is in general strictly smaller than $\pi^* := \min_{\mathbf{x}\in R} F(\mathbf{x})$. This is called duality gap . For convex F, as in our case, and if the feasibility region $R = \{\mathbf{x}\in\mathbb{R}^d : \mathbf{B}\mathbf{x}-\mathbf{b} > \mathbf{0}\}$ is not empty (Slater condition), then one can show that $\delta^* = \pi^*$ (strong duality holds), that is the duality gap is zero, see e.g. [3, Chapter 5].*
- *A particular interest in the dual formulation arises if $d \gg m$, i.e. there are many more primal variables than constraints. In fact, the dual problem involves than far fewer variables.*

SVM Optimization Problem — Primal/Dual Formulations

Back to linear SVM (slacks $\xi_j = 0$ hard-margin SVM): Recall

$$\text{minimize over } \mathbf{w}, b \quad \underbrace{\frac{1}{2}\sum_{j=1}^{N} w_j^2}_{\text{objective}} \quad \text{subject to} \quad \underbrace{y_j(\mathbf{w}\cdot\mathbf{x}^j + b) \geq 1, \quad 1 \leq j \leq N}_{\text{constraints}}. \tag{3.37}$$

- This is the primal formulation of a QP problem with $d+1$ variables $b, w_i, i = 1, \ldots, d$, and N constraints, i.e. $d \leftrightarrow d+1$, $m \leftrightarrow N$ in the above general situation.
- By Remark 3, this is a special case of the QP-problem (3.12) with

$$(w_1, \ldots, w_d, b)^\top \leftrightarrow \mathbf{x}, \quad \boldsymbol{A} = \begin{pmatrix} \mathbf{I}_d & \mathbf{0} \\ \mathbf{0}^\top & 0 \end{pmatrix}, \qquad \begin{aligned} B_{i,d+1} &= y_i, && , i = 1, \ldots, N, \\ B_{i,k} &= y_i x_k^i, && i = 1, \ldots, N,\ k = 1, \ldots, d, \\ b_i &= 1, && i = 1, \ldots, N. \end{aligned} \tag{3.38}$$

  i.e., $\mathbf{b} \in \mathbb{R}^N$, $\mathbf{B} \in \mathbb{R}^{N\times(d+1)}$.
- SVMs work robustly even when $d \gg N$. Therefore, it is of interest to consider the dual formulation.
- In this case: $\boldsymbol{A}$ is singular.

Derivation of the Dual Formulation

Primal Formulation:

$$\text{minimize over } \mathbf{w}, b \quad \underbrace{\frac{1}{2}\sum_{j=1}^{N} w_j^2}_{\text{objective}} \quad \text{subject to} \quad \underbrace{y_j(\mathbf{w}\cdot\mathbf{x}^j + b) \geq 1, \quad 1 \leq j \leq N,}_{\text{constraints}}.$$

In the present case the Lagrangian associated with the above problem reads

$$L(\mathbf{w}, b; \mathbf{a}) := \frac{1}{2}\sum_{j=1}^{d} w_j^2 - \sum_{j=1}^{N} a_j \left(y_j(\mathbf{w}\cdot\mathbf{x}^j + b) - 1\right) \tag{3.39}$$

For fixed $\mathbf{a}$ it is convex in $(\mathbf{w}, b)$. Therefore, a(n unconstrained) minimizer of $L(\mathbf{w}, b; \mathbf{a})$ over $(\mathbf{w}, b)$ for fixed $\mathbf{a}$ must be a critical point:

$$\mathbf{0} = \nabla_{\mathbf{w},b} L(\mathbf{w}, b; \mathbf{a}).$$

Computation of the Dual Function

$$L(\mathbf{w}, b; \mathbf{a}) := \frac{1}{2}\sum_{j=1}^{d} w_j^2 - \sum_{j=1}^{N} a_j \left(y_j(\mathbf{w}\cdot\mathbf{x}^j + b) - 1\right)$$

To compute $H(\mathbf{a}) := \inf_{(\mathbf{w},b)\in\mathbb{R}^{d+1}} L(\mathbf{w}, b; \mathbf{a})$ compute the critical points $\nabla_{\mathbf{w},b} L(\mathbf{w}, b; \mathbf{a}) \stackrel{!}{=} \mathbf{0}$:

$$\partial_{w_i} L(\mathbf{w}, b; \mathbf{a}) = w_i - \sum_{j=1}^{N} a_j y_j x_i^j \stackrel{!}{=} 0, \quad i = 1, \ldots, d, \tag{3.40}$$

$$\partial_b L(\mathbf{w}, b; \mathbf{a}) = -\sum_{j=1}^{N} a_j y_j \stackrel{!}{=} 0, \tag{3.41}$$

Thus

$$\begin{aligned} (3.40) \quad &\Rightarrow \quad \mathbf{w} = \sum_{j=1}^{N} a_j y_j \mathbf{x}^j; \\ (3.41) \quad &\Rightarrow \quad \sum_{j=1}^{N} a_j y_j = 0 = \mathbf{a}\cdot\mathbf{y}, \quad \mathbf{y} \in \{-1, 1\}^N. \end{aligned} \tag{3.42}$$

Computation of the Dual Function

Using the relations (3.42) for $\mathbf{w}$, yields:

$$\frac{1}{2}\sum_{j=1}^{d} w_j^2 = \frac{1}{2}\mathbf{w}\cdot\mathbf{w} \overset{(3.42)}{=} \frac{1}{2}\Big(\sum_{j=1}^{N} a_j y_j \mathbf{x}^j\Big)^\top \Big(\sum_{q=1}^{N} a_q y_q \mathbf{x}^q\Big) = \frac{1}{2}\sum_{i,j=1}^{N} a_i a_j y_i y_j \mathbf{x}^i \cdot \mathbf{x}^j$$

$$\begin{aligned}\sum_{j=1}^{N} a_j\big(y_j(\mathbf{w}\cdot\mathbf{x}^j + b) - 1\big) &\overset{(3.42)}{=} \sum_{j=1}^{N} a_j(y_j(\mathbf{w}\cdot\mathbf{x}^j) - 1) \quad (b \text{ drops out})\\ &= \sum_{j=1}^{N} a_j y_j\Big(\sum_{q=1}^{N} a_q y_q \mathbf{x}^q\cdot\mathbf{x}^j\Big) - \sum_{i=1}^{N} a_i\\ &= \sum_{i,j=1}^{N} a_i a_j y_i y_j \mathbf{x}^i\cdot\mathbf{x}^j - \sum_{i=1}^{N} a_i.\end{aligned}$$

Subtracting yields

$$H(\mathbf{a}) = \sum_{i=1}^{N} a_i - \frac{1}{2}\sum_{i,j=1}^{N} a_i a_j y_i y_j \mathbf{x}^i\cdot\mathbf{x}^j = \sum_{i=1}^{N} a_i - \frac{1}{2}\mathbf{a}^\top\mathbf{M}\mathbf{a}, \quad \mathbf{M} = (y_i\mathbf{x}^i\cdot\mathbf{x}^j y_j)_{i,j=1}^{N,N}. \tag{3.43}$$

The Dual Problem

$$\text{maximize}_{\mathbf{a}\in\mathbb{R}^N}: \underbrace{\sum_{i=1}^N a_i - \frac{1}{2}\sum_{i,j=1}^N a_i a_j y_i y_j \mathbf{x}^i \cdot \mathbf{x}^j}_{\text{Objective: } \mathbf{a}^\top \mathbf{1} - \frac{1}{2}\mathbf{a}^\top \mathbf{M}\mathbf{a}} \quad \text{subject to} \quad \underbrace{a_i \geq 0, \text{ and } \sum_{j=1}^N a_j y_j = 0}_{\text{Constraints}}. \quad (3.44)$$

1. Retrieve $\mathbf{w}$: (3.42) $\rightsquigarrow$ $\mathbf{w} := \sum_{j=1}^N a_j y_j \mathbf{x}^j$
2. Retrieve b: let $\mathcal{I}_+ := \{i \in \operatorname{supp} \bar{\mathbf{a}} : y_i = +1\}$, take $i_0 := \operatorname{argmin}\{\mathbf{w}\cdot\mathbf{x}^i : i \in \mathcal{I}_+\}$, set $b := 1 - \mathbf{w}\cdot\mathbf{x}^{i_0}$
3. Classifier: $h(\mathbf{x}) := \operatorname{sgn}(\mathbf{w}\cdot\mathbf{x} + b)$.

Comments:

- If the dimension d is very large (e.g. document classification $d \geq 10^6$) and $N \ll d$ the dual problem involves much fewer variables (e.g. if a microarray dataset contains $20,000$ genes and $N = 100$.
- One doesn't have to access the original data but only the inner products $\mathbf{x}^i \cdot \mathbf{x}^j$, $\mathbf{w}\cdot\mathbf{x}$.
- The optimization problem can actually be reduced to a(n often) much smaller size. In fact, the solution $\bar{\mathbf{a}}$ of the dual problem has has only $\#\operatorname{supp}(\bar{\mathbf{a}}) = \#$ of active constraints positive entries. For an active constraint j one has $y_j(\mathbf{w}\cdot\mathbf{x}^j + b) - 1 = 0$. This means $\mathbf{x}^j$ is a support vector. So what counts is the number of support vectors, identified by $\operatorname{supp}(\bar{\mathbf{a}})$.

Summary

Recall from (3.9): linear soft-margin SVM leads to (given the data $(\mathbf{x}^i, y_i)$, $i = 1, \ldots, N$,

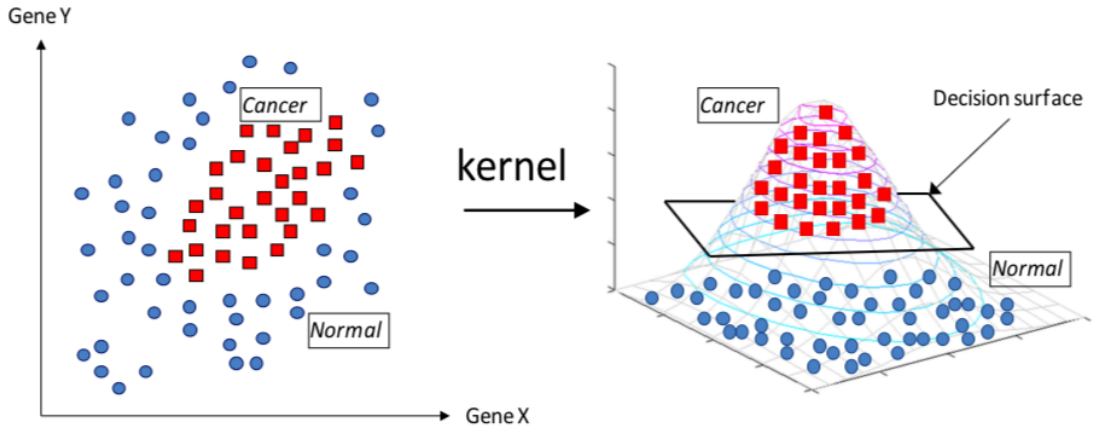
$$\text{minimize}_{\mathbf{w},b,\xi} \quad \underbrace{\frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{j=1}^{N}\xi_j}_{\text{objective}} \quad \text{subject to} \quad \underbrace{y_j(\mathbf{w}\cdot\mathbf{x}^j + b) \geq 1 - \xi_j \quad , \; 1 \leq j \leq N,}_{\text{constraints}}.$$

This is again the primal form of a QP problem (this time with primal variables $\mathbf{w} = (w_1, \ldots, w_d)^\top, \xi = (\xi_1, \ldots, \xi_N)^\top, b \in \mathbb{R}$) and can therefore be treated in the same way as before. Concluding Remarks:

- We have shown so far only how to formulate linear SVMs QP problems.
- Depending on d and N (which one is larger) the actual optimization code can be applied to the primal or dual formulation.
- The complexity of the dual formulation depends essentially on the number of active constraints.
- The discussion so far therefore concerned only setting up the mathematical problem. Discussing the concrete numerical algorithms needed to solve such QP problems is a course by itself.
- QP-algorithms are typically iterative, often based on a prediction step, eg. as a Newton-step for solving (3.21) followed by a correction step to restore feasibility. Key words: interior point methods, barrier methods, see e.g. [2, 3, 8].

Kernel Trick

The class of problems where data with different lables can be (nearly) linearly separated is, of course, restricted. A way of extending the viability of linear separators is to “lift” the feature vectors $\mathbf{x}^j$ first into a higher dimensional feature space where they can be (at least approximately) linearly separated.



Kernel Trick:

- Choose a mapping $\Phi := \mathcal{X}(= \mathbb{R}^d) \to \mathcal{H}(= \mathbb{R}^D)$, called feature map, that takes the data $\mathbf{x}^j$, $j = 1, \ldots, N$, into a (typically) higher dimensional space $\mathcal{H}$.
- Use a linear SVM to separate the data in $\mathcal{H}$.

Kernel SVMs

Given $\Phi : \mathcal{X} \to \mathcal{H}$, solve

$$\text{Minimize}_{\mathbf{w}\in\mathbb{R}^D, b} : \quad \frac{1}{2}\|\mathbf{w}\|_2^2 \text{ subject to } y_j(\mathbf{w}\cdot\Phi(\mathbf{x}^j)+b) \geq 1,\ 1 \leq j \leq N. \tag{3.45}$$

Now, the dimension is D in which the optimization takes place. So, it is the more important to go by the dual method: In view of (3.44), this reads

$$\text{maximize}_{\mathbf{a}\in\mathbb{R}^N} : \quad \sum_{i=1}^N a_i - \frac{1}{2}\sum_{i,j=1}^N a_i a_j y_i y_j \underbrace{\Phi(\mathbf{x}^i)\cdot\Phi(\mathbf{x}^j)}_{K(\mathbf{x}^i,\mathbf{x}^j)} \quad \text{subject to} \quad a_i \geq 0, \text{ and } \sum_{j=1}^N a_j y_j = 0. \tag{3.46}$$

Once $\mathbf{a} = (a_1, \ldots, a_N)^\top \in \mathbb{R}_+^N$ has been computed:

1. Retrieve $\mathbf{w}$: (3.42) $\rightsquigarrow$ $\mathbf{w} := \sum_{j=1}^N a_j y_j \Phi(\mathbf{x}^j)$
2. Retrieve b: let $\mathcal{I}_+ := \{i \in \operatorname{supp}\bar{\mathbf{a}} : y_i = +1\}$, take $i_0 := \operatorname{argmin}\{\mathbf{w}\cdot\Phi(\mathbf{x}^j) : i \in \mathcal{I}_+\}$, set $b := 1 - \mathbf{w}\cdot\Phi(\mathbf{x}^{i_0})$
3. Classifier: (3.42) $\Rightarrow$

$$\begin{aligned} h(\mathbf{x}) &:= \operatorname{sgn}(\mathbf{w}\cdot\Phi(\mathbf{x})+b) \overset{(1)}{=} \operatorname{sgn}\left(\left\{\sum_{j=1}^N a_j y_j \Phi(\mathbf{x}^j)\cdot\Phi(\mathbf{x})\right\}+b\right) \\ &= \operatorname{sgn}\left(\sum_{j=1}^N a_j y_j K(\mathbf{x}^j,\mathbf{x})+b\right). \end{aligned} \tag{3.47}$$

Comments:

- One does not need to know the mapping Φ. One only needs to know the kernel $K(\mathbf{x}, \mathbf{y}) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$.
- The objective function in (3.46) can be rewritten as

$$\sum_{i=1}^{N} a_i - \frac{1}{2} \sum_{i,j=1}^{N} a_i \left(y_i K(\mathbf{x}^i, \mathbf{x}^j) y_j\right) a_j = \mathbf{1} \cdot \mathbf{a} - \frac{1}{2} \mathbf{a}^\top \hat{\mathbf{K}} \mathbf{a}, \tag{3.48}$$

where

$$\hat{\mathbf{K}} = \left(y_i K(\mathbf{x}^i, \mathbf{x}^j) y_j\right)_{i,j}^{N} \in \mathbb{R}^{N \times N}. \tag{3.49}$$

- Maximizing $\mathbf{1} \cdot \mathbf{a} - \frac{1}{2} \mathbf{a}^\top \hat{\mathbf{K}} \mathbf{a}$ is equivalent to minimizing $\frac{1}{2} \mathbf{a}^\top \hat{\mathbf{K}} \mathbf{a} - \mathbf{1} \cdot \mathbf{a}$ subject to the same constraints from (3.46). This is again precisely a QP-problem of the general type (3.26) provided that the matrix $\hat{\mathbf{K}}$ is symmetric positive definite.
- Note: $\hat{\mathbf{K}}$ is symmetric positive (semi-)definite if and only if $\mathbf{K} := \left(K(\mathbf{x}^i, \mathbf{x}^j)\right)_{i,j=1}^{N}$ is symmetric positive (semi-)definite - verify !.
- This is where Φ enters the picture:

$$\begin{aligned} \mathbf{v}^\top \mathbf{K} \mathbf{v} &= \sum_{i,j=1}^{N} v_i \Phi(\mathbf{x}^i) \cdot \Phi(\mathbf{x}^j) v_j = \sum_{i,j=1}^{N} v_i \Big\{ \sum_{k=1}^{D} \phi_k(\mathbf{x}^i) \phi_k(\mathbf{x}^j) \Big\} v_j \\ &= \sum_{k=1}^{D} \sum_{i,j=1}^{N} (v_i \phi_k(\mathbf{x}^i)(v_j \phi_k(\mathbf{x}^j)) = \sum_{k=1}^{D} \Big(\sum_{i=1}^{N} v_i \phi_k(\mathbf{x}^i) \Big)^2 \geq 0. \end{aligned} \tag{3.50}$$

Comments continued:

- Algorithmically things work therefore as in the linear case.
- The transition to the dual problem is the more important since
  - The vector $\mathbf{w}$ belongs to $\mathbb{R}^D$ with $D > d$, so the primal problem is posed in an even higher dimensional space and hence harder to handle.
  - The constraints in the primal problem are now in general nonlinear because of Φ.
  - The size of the dual problem is still the same N, the number of data and the constraints are still linear.
  - The main operations involve inner products.
- The classifier (3.47) is of the form $h(\mathbf{x}) = \operatorname{sgn}\big(f(\mathbf{x})\big)$ where here

$$f(\mathbf{x}) = \sum_{j=1}^{N} a_j y_j K(\mathbf{x}^j, \mathbf{x}) + b. \tag{3.51}$$

Remark 12

The estimator is a plug-in-estimator, where f is a linear combination (up to a constant shift b) of the functions $k_j(\mathbf{x}) = K(\mathbf{x}^j, \mathbf{x})$, $j = 1, \ldots, N$ (which happens to depend on the data), and is therefore (for fixed data) a linear estimator.

Which kernels should be used?

Frequently Used Kernels

Since computationally only the kernel (not the explicit feature map Φ) matters we wish to better understand what makes a kernel useful. A kernel is called legal kernel if it is a dot-product:

$$K(\mathbf{x}^i, \mathbf{x}^j) = \Phi(\mathbf{x}^i) \cdot \Phi(\mathbf{x}^j) = \sum_{k=1}^{D} \phi_k(\mathbf{x}^i)\phi_k(\mathbf{x}^j), \tag{3.52}$$

where $\Phi : \mathbb{R}^d \to \mathbb{R}^D$ could be any feature map. For the actual computations only the kernel matters. The following are legal kernels.

Examples:

1. $K(\mathbf{x}^i, \mathbf{x}^j) = \mathbf{x}^i \cdot \mathbf{x}^j$ — linear SVM
2. $K(\mathbf{x}^i, \mathbf{x}^j) = \epsilon^{-\gamma\|\mathbf{x}^i - \mathbf{x}^j\|_2^2}$, $\gamma > 0$ — Gaussian kernel
3. $K(\mathbf{x}^i, \mathbf{x}^j) = (a + \mathbf{x}^i \cdot \mathbf{x}^j)^q$, $a \geq 0, q \in \mathbb{N}$ — polynomial kernel
4. $K(\mathbf{x}^i, \mathbf{x}^j) = (a + \mathbf{x}^i \cdot \mathbf{x}^j)^q \epsilon^{-\gamma\|\mathbf{x}^i - \mathbf{x}^j\|_2^2}$, $a \geq 0, q \in \mathbb{N}$ — hybrid kernel
5. $K(\mathbf{x}^i, \mathbf{x}^j) = \tanh(k\mathbf{x}^i \cdot \mathbf{x}^j + b)$ — sigmoidal kernel

Why are these indeed legal kernels, i.e., have the structure (3.52)?

Useful Kernel Rules

Proposition 13

1. *A positive constant is a legal kernel.*
2. *Let $K(\mathbf{x}, \mathbf{y})$ be a legal kernel and c be a positive consant. Then $cK(\mathbf{x}, \mathbf{y})$ is a legal kernel.*
3. *If $K(\mathbf{x}, \mathbf{y})$ is a legal kernel and f any scalar function, then $f(\mathbf{x})f(\mathbf{y})K(\mathbf{x}, \mathbf{y})$ is a legal kernel (with the same feature dimension as before).*
4. *The sum of legal kernels is a legal kernel where the new feature dimension is the sum of the original ones.*
5. *The product of legal kernels is a legal kernel where the new feature dimension is the product of the original ones.*

Proof: (1), (2) and (3) are obvious. As for (4), let $K_1(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{y})$, $K_2(\mathbf{x}, \mathbf{y}) = \Psi(\mathbf{x}) \cdot \Psi(\mathbf{y})$. Then

$$\begin{aligned} \Phi(\mathbf{x}) \cdot \Phi(\mathbf{y}) + \Psi(\mathbf{x}) \cdot \Psi(\mathbf{y}) &= \sum_{k=1}^{D_\Phi} \phi_k(\mathbf{x})\phi_k(\mathbf{y}) + \sum_{k=1}^{D_\Psi} \psi_k(\mathbf{x})\psi_k(\mathbf{y}) \\ &= \begin{pmatrix} \Phi \\ \Psi \end{pmatrix}(\mathbf{x}) \cdot \begin{pmatrix} \Phi \\ \Psi \end{pmatrix}(\mathbf{y}), \quad D = D_\Phi + D_\Psi . \end{aligned}$$

Proof of Proposition 13 continued: Concerning (5),

$$\begin{aligned}(\Phi(\mathbf{x})\cdot\Phi(\mathbf{y}))((\Psi(\mathbf{x})\cdot\Psi(\mathbf{y})) &= \Big(\sum_{k=1}^{D_\Phi}\phi_k(\mathbf{x})\phi_k(\mathbf{y})\Big)\Big(\sum_{j=1}^{D_\Psi}\psi_j(\mathbf{x})\psi_j(\mathbf{y})\Big)\\ &= \sum_{k,j=1}^{D_\Phi,D_\Psi}(\phi_k(\mathbf{x})\psi_j(\mathbf{x}))(\phi_k(\mathbf{y})\psi_j(\mathbf{y})) =: \sum_{\nu=1}^{D_\Theta}\theta_\nu(\mathbf{x})\theta_\nu(\mathbf{y})\\ &= \Theta(\mathbf{x})\cdot\Theta(\mathbf{y}),\quad D_\Theta = D_\Phi\cdot D_\Psi. \qquad \square\end{aligned}$$

One can now confirm that the above examples of kernels are indeed legal:

Example (1): trivial

Example (3): follows from Proposition 13, (1), (2), (4), and (5).

Example (2): write

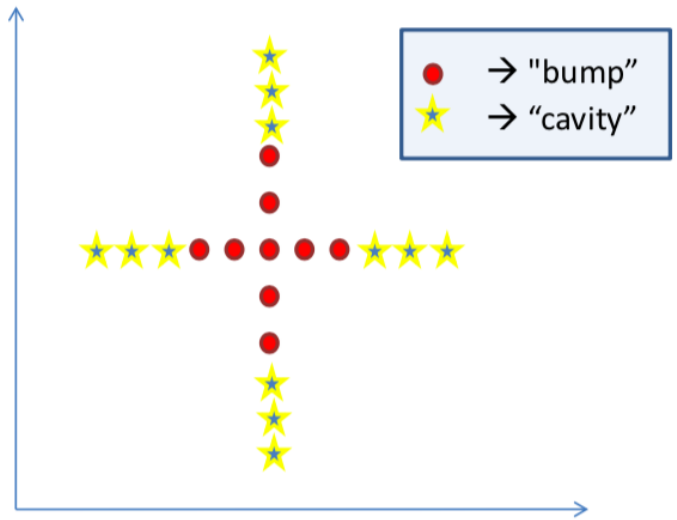
$$e^{-\gamma\|\mathbf{x}-\mathbf{y}\|_2^2} = e^{-\gamma\|\mathbf{x}\|_2^2}e^{-\gamma\|\mathbf{y}\|_2^2}e^{2\gamma\mathbf{x}\cdot\mathbf{y}} = f(\mathbf{x})f(\mathbf{y})\Big(\sum_{k=0}^{\infty}\frac{(\mathbf{x}\cdot\mathbf{y})^k}{k!}\Big).$$

By Proposition 13, (2), (5), each summand in the series on the right is a legal kernel. By Proposition 13, (4), the series is a legal kernel as well. The rest follows from Proposition 13, (3).

Some Illustrations (see [9])

The Gaussian kernel: $K(\mathbf{x}, \mathbf{x}^j) = K(\mathbf{x}^j, \mathbf{x}) = e^{-\gamma \|\mathbf{x}-\mathbf{x}^j\|_2^2}$, some $\gamma > 0$

Geometrically: a local "bump" which is very concentrated when γ is large and flat when γ is small.



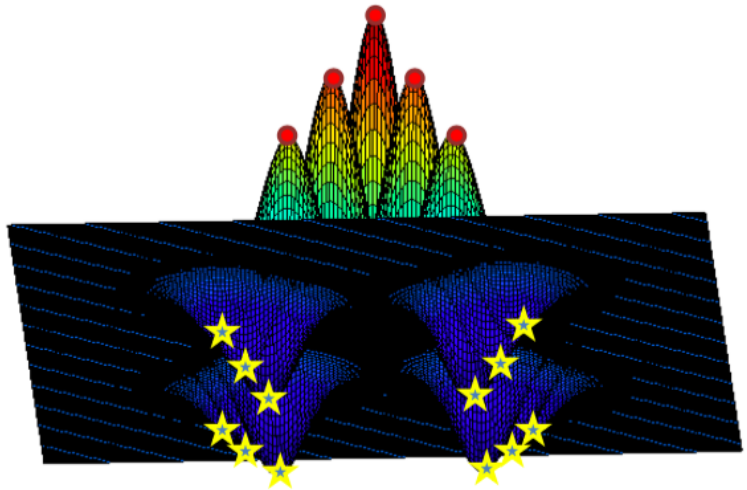
The resulting mapping function is a **combination** of bumps and cavities.

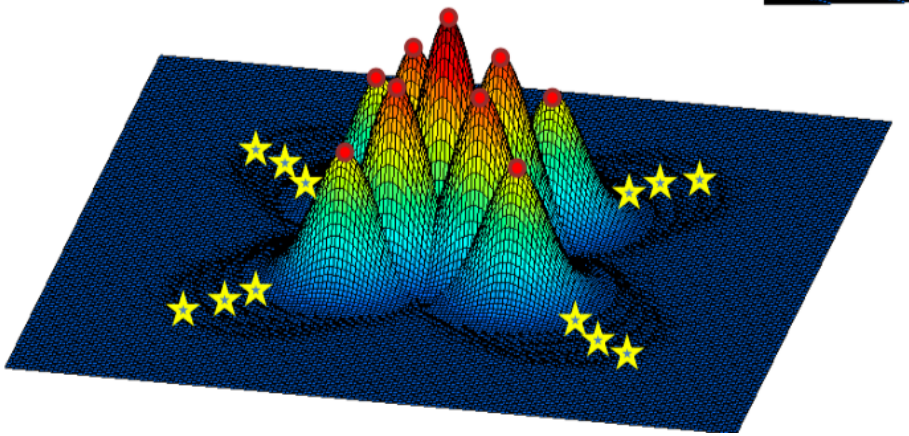
61

The plug-in-function *f* from (3.51) is in this case a linear combinations of Gaussian "bumps" and "cavities".

Some Illustrations (see [9])

Several more views of the data is mapped to the feature space by Gaussian kernel

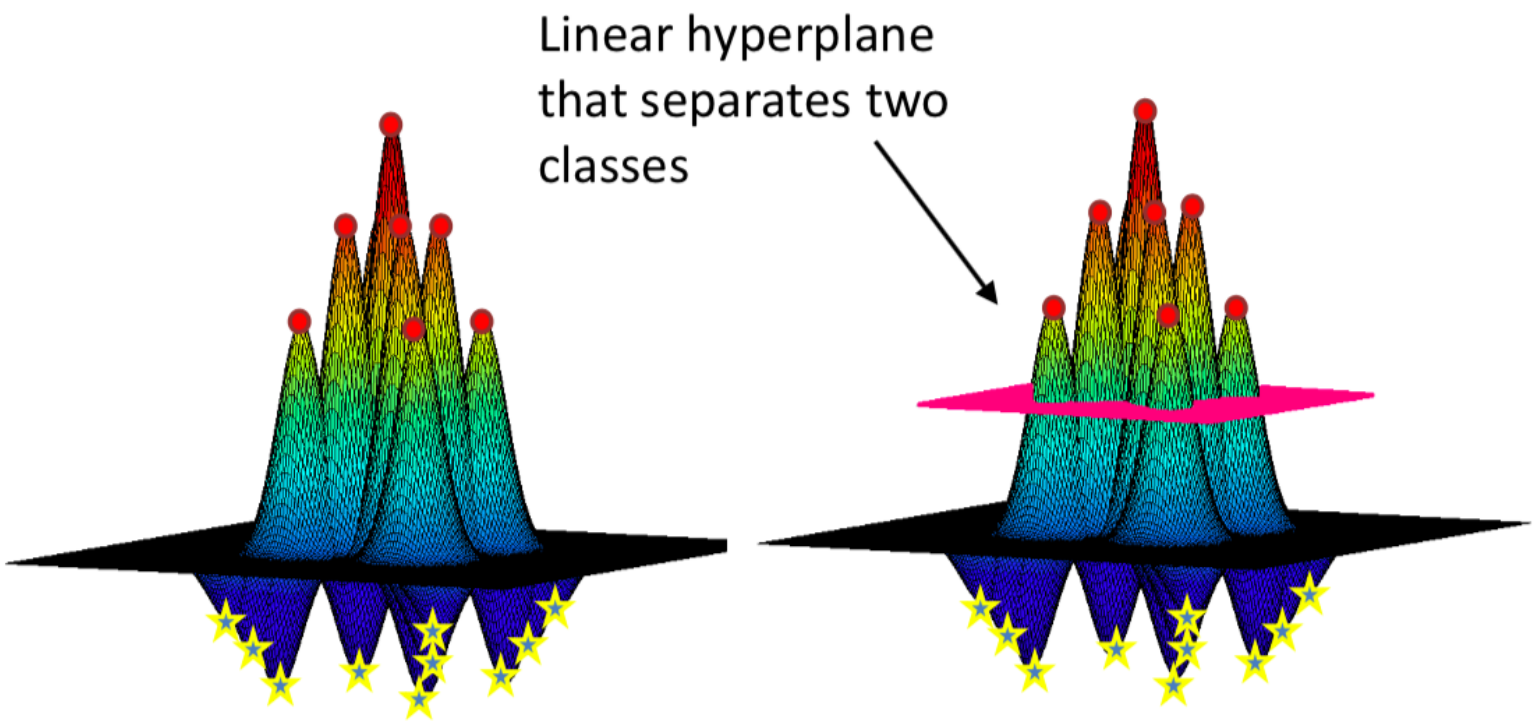




62

Linear combinations of Gaussian “bumps” and “cavities”.

Some Illustrations (see [9])



Linear combinations of Gaussian “bumps” and “cavities”.

Some Illustrations (see [9])

Polynomial kernel: $K(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x} \cdot \mathbf{y})^3$, $d = 2$, $D = 10$

Assume that we are dealing with 2-dimensional data (i.e., in $\mathbb{R}^2$). Where will this kernel map the data?

2-dimensional space

$x_{(1)} \quad x_{(2)}$

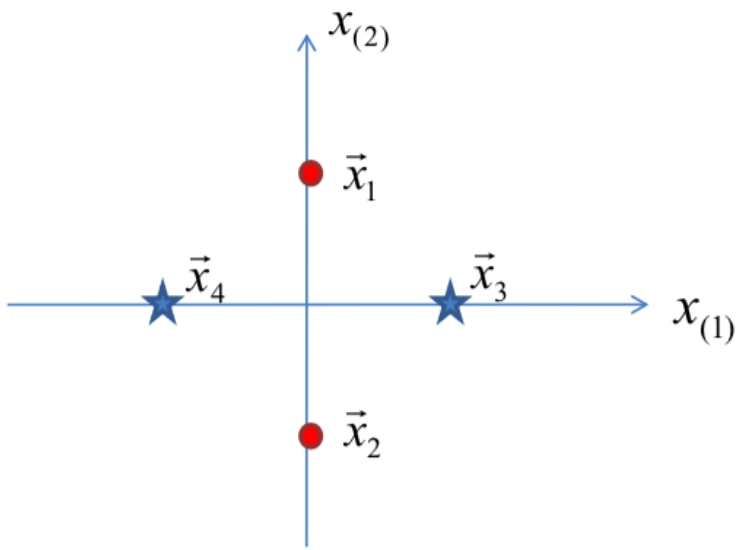
kernel

10-dimensional space

$1 \quad x_{(1)} \quad x_{(2)} \quad x_{(1)}^2 \quad x_{(2)}^2 \quad x_{(1)}x_{(2)} \quad x_{(1)}^3 \quad x_{(2)}^3 \quad x_{(1)}x_{(2)}^2 \quad x_{(1)}^2x_{(2)}$

Some Illustrations (see [9])

Polynomial kernel: $K(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y})^2$, $d = 2$, $D = 3$



- Data is not linearly separable in the input space ($\mathbb{R}^2$).
- Apply kernel $K(\vec{x}, \vec{z}) = (\vec{x} \cdot \vec{z})^2$ to map data to a higher dimensional space (3-dimensional) where it is linearly separable.

$$K(\vec{x}, \vec{z}) = (\vec{x} \cdot \vec{z})^2 = \left[\begin{pmatrix} x_{(1)} \\ x_{(2)} \end{pmatrix} \cdot \begin{pmatrix} z_{(1)} \\ z_{(2)} \end{pmatrix} \right]^2 = \left[x_{(1)} z_{(1)} + x_{(2)} z_{(2)} \right]^2 =$$

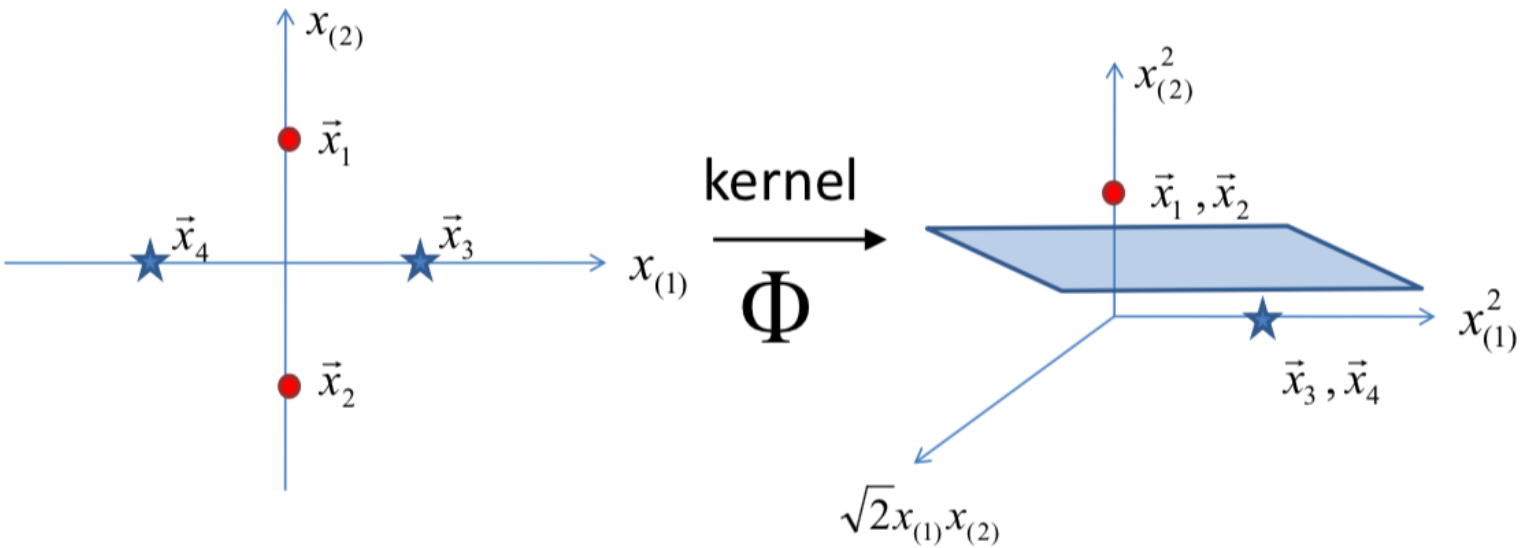
$$= x_{(1)}^2 z_{(1)}^2 + 2 x_{(1)} z_{(1)} x_{(2)} z_{(2)} + x_{(2)}^2 z_{(2)}^2 = \begin{pmatrix} x_{(1)}^2 \\ \sqrt{2} x_{(1)} x_{(2)} \\ x_{(2)}^2 \end{pmatrix} \cdot \begin{pmatrix} z_{(1)}^2 \\ \sqrt{2} z_{(1)} z_{(2)} \\ z_{(2)}^2 \end{pmatrix} = \Phi(\vec{x}) \cdot \Phi(\vec{z})$$

65

Some Illustrations (see [9])

Polynomial kernel: $K(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y})^2$, $d = 2$, $D = 3$

Therefore, the explicit mapping is $\Phi(\vec{x}) = \begin{pmatrix} x_{(1)}^2 \\ \sqrt{2}x_{(1)}x_{(2)} \\ x_{(2)}^2 \end{pmatrix}$



Mercer Kernels

- Question: What Qualifies a function $K(\mathbf{x}, \mathbf{y})$ on $\mathcal{X} \times \mathcal{X}$ as a legal kernel?
- The Key Property: that makes the dual formulation work is that for any samples $\mathbf{x}^j \in \mathcal{X}$, $j = 1, \ldots, N$, the matrix

$$\mathbf{K} = \left(K(\mathbf{x}^i, \mathbf{x}^j)\right)_{i,j=1}^{N} \in \mathbb{R}^{N \times N}, \tag{3.53}$$

  is symmetric positive (semi-)definite.

Definition 14

A continuous symmetric funktion $K \in C(\mathcal{X} \times \mathcal{X})$ such that for any $\mathbf{x}^j \in \mathcal{X}$, $j = 1, \ldots, N$, any $N \in \mathbb{N}$, the matrix $\mathbf{K}$ from (3.53) is symmetric positive definite, is called a positive definite kernel or Mercer kernel, see [6].

- Being induced by a feature map $K(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{y}) = \sum_{k=1}^{D} \phi_k(\mathbf{x})\phi_k(\mathbf{y})$ is by (3.50) sufficient for a kernel to be positive definite. $D = \infty$ could happen.
- We'll see that it is in essence also necessary. The following Theorem is a special case of what is called Hilbert-Schmidt Decomposition tailored to the current needs. It can be regarded as an infinite-dimensional version of the SVD or the spectral decomposition theorem (see Lecture I, Theorem 31, Theorem 39).

Hilbert Schmidt Decompositions Background Material

Theorem 15

Assume that p is a separable density on $\mathcal{X} \times \mathcal{Y}$ (i.e., $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y})$) and $G \in L_2(\mathcal{X} \times \mathcal{Y}; p) = \|G\|_{L_2(\mathcal{X}\times\mathcal{Y};p)}$, i.e., $\int_{\mathcal{X}\times\mathcal{Y}} |G(\mathbf{x}, \mathbf{y})|^2 dP(\mathbf{x}, \mathbf{y}) < \infty$. Then, the operator $M_G : \mathcal{Y} \to \mathcal{X}$ defined by

$$(M_G)(v) := \int_{\mathcal{Y}} G(\cdot, \mathbf{y}) v(\mathbf{y}) dP(\mathbf{y}), \tag{3.54}$$

is compact and there exist orthonormal systems $\{\phi_k : k \in \mathcal{I}\} \subset \mathcal{X}$, $\{\psi_k : k \in \mathcal{I}\} \subset \mathcal{Y}$ such that

$$G(\mathbf{x}, \mathbf{y}) = \sum_{k\in\mathcal{I}} \sigma_k \phi_k(\mathbf{x}) \psi_k(\mathbf{y}), \quad \text{a.e.} \quad \text{where} \quad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq \cdots \geq 0, \ \sigma_k \to 0. \tag{3.55}$$

Moreover

$$\|M_G\|_{\mathcal{L}(\mathcal{Y},\mathcal{X})} = \sigma_1, \quad \|M_G\|_{\mathrm{HS}} = \|G\|_{L_2(\mathcal{X}\times\mathcal{Y};p)} = \Big(\sum_{j\in\mathcal{I}} \sigma_k^2\Big)^{1/2}. \tag{3.56}$$

Here $\mathcal{I} \subseteq \mathbb{N}$ is the largest subset for which the σ_k are strictly positive.

cf. Lecture I, Theorem 3.9

Spectral Decomposition for Compact Positive Definite Operators

Theorem 16

Assume that p is a probability density on $\mathcal{X}$ and $G \in L_2(\mathcal{X} \times \mathcal{X}; p \otimes p) = \|G\|_{L_2(\mathcal{X}\times\mathcal{X}; p\otimes p)}$, i.e., $\int_{\mathcal{X}\times\mathcal{X}} |G(\mathbf{x},\mathbf{y})|^2 dP(\mathbf{x})dP(\mathbf{y}) < \infty$. Moreover, assume that G is symmetric positive definite, i.e.,

$$\int_{\mathcal{X}\times\mathcal{X}} v(\mathbf{x})G(\mathbf{x},\mathbf{y})v(\mathbf{y})dP(\mathbf{x})dP(\mathbf{y}) \geq 0 \;\forall\; v \in \mathcal{H}, \quad G(\mathbf{x},\mathbf{y}) = G(\mathbf{y},\mathbf{x}),\ \mathbf{x},\mathbf{y} \in \mathcal{X}. \tag{3.57}$$

Then, $(M_G)(v) := \int_{\mathcal{X}} G(\cdot,\mathbf{y})v(\mathbf{y})dP(\mathbf{y})$ (see (3.54)) is a compact symmetric positive definite operator and there exists an orthonormal system $\{\phi_k : k \in \mathcal{I}\} \subset \mathcal{X}$ such that

$$G(\mathbf{x},\mathbf{y}) = \sum_{k\in\mathcal{I}} \lambda_k \phi_k(\mathbf{x})\phi_k(\mathbf{y}), \quad \text{a.e.} \quad \text{where} \quad \lambda_1 \geq \lambda_2 \geq \cdots \geq 0,\ \lambda_k \to 0, \tag{3.58}$$

i.e., $M_G\phi_k = \lambda_k\phi_k$, $k \in \mathcal{I}$ (eigensystem of M_G). Moreover, one has

$$\|M_G\|_{\mathcal{L}(\mathcal{X},\mathcal{X})} = \sigma_1, \quad \|M_G\|_{\mathrm{HS}} = \|G\|_{L_2(\mathcal{X}\times\mathcal{X}; p\otimes p)} = \Big(\sum_{j\in\mathcal{I}} \lambda_k^2\Big)^{1/2}. \tag{3.59}$$

cf. Lecture I, Theorem 3.1

Mercer Kernels

- The compactness of the operator M_G induced by G, i.e., the fact that $\|G\|_{L_2(\mathcal{X}\times\mathcal{X},\otimes^2 p)} < \infty$ is responsible for the fact that M_G has always a discrete spectrum also when $\mathcal{I} = \mathbb{N}$ with $\sigma_k, \lambda_k \to 0$.
- Without positive definiteness but only symmetry in (3.57) one could still conclude a decomposition like (3.58) but without the sign condition on the spectrum $\{\lambda_k\}_{k\in\mathcal{I}}$.
- Theorem 16 says that if G is symmetric positive definite and square integrable in $\mathcal{X}\times\mathcal{X}$ then, there exists a feature map $\Phi : \mathcal{X} \to L_2(\mathcal{X}, p)^{\mathcal{I}}$, where

$$\Phi(\mathbf{x}) = (\sqrt{\lambda_k}\phi_k(\mathbf{x}))_{k\in\mathcal{I}}, \quad D = \#(\mathcal{I})(= \infty \text{ if } \mathcal{I} = \mathbb{N}) \tag{3.60}$$

- This almost what we need for G to qualify as a kernel, except that pointwise evaluations need to be well-defined, i.e., a kernel has to be continuous.

Mercer's Theorem

Theorem 17

Suppose that K is a Mercer kernel on $\mathcal{X} \times \mathcal{X}$, i.e., $K(\cdot,\cdot) \in L_2(\mathcal{X} \times \mathcal{X}, \otimes^2 p) \cap C(\mathcal{X} \times \mathcal{X})$ be symmetric positive definite (3.57)*. Then, K can be expanded as*

$$K(\mathbf{x},\mathbf{y}) = \sum_{k \in \mathcal{I}} \lambda_k \phi_k(\mathbf{x}) \phi_k(\mathbf{y}), \quad \text{a.e.} \quad \text{where} \quad \lambda_1 \geq \lambda_2 \geq \cdots \geq 0, \ \lambda_k \to 0. \tag{3.61}$$

where $\{\phi_k : k \in \mathcal{I}\} \subset L_2(\mathcal{X} \times \mathcal{X}, \otimes^2 p)$ is an orthonormal system and the series (3.61) *converges absolutely and uniformly on compact subsets of $\mathcal{X}$,i.e., for any compact $D \subseteq \mathcal{X} \times \mathcal{X}$ one has*

$$\lim_{n \to \infty} \sup_{(\mathbf{x},\mathbf{y}) \in D} \left| K(\mathbf{x},\mathbf{y}) - \sum_{k \leq n} \lambda_k \phi_k(\mathbf{x}) \phi_k(\mathbf{y}) \right| = 0.$$

Moreover, for any $\mathbf{x}^j \in \mathcal{X}$, $j = 1, \ldots, N$, any $N \in \mathbb{N}$, the matrix

$$\mathbf{K} = \left(K(\mathbf{x}^i, \mathbf{x}^j)\right)_{i,j=1}^N \in \mathbb{R}^{N \times N}$$

is symmetric positive definite.

Comments on the Proof of Theorem 17: The existence of the decomposition (3.61) follows directly from Theorem 16.

Since K is continuous and $K \to \int_{\mathcal{X}} K(\mathbf{x},\mathbf{y})\phi_k(y)dP(\mathbf{y}) = \lambda_k\phi_k(\mathbf{x})$ is continuous we conclude that the ϕ_k are continuous. One can also show that M_K maps into $C(\mathcal{X})$.

Thus, the feature map $\Phi(\mathbf{x}) := (\sqrt{\lambda_k}\phi_k(\mathbf{x}))_{k\in\mathcal{I}}$ is well defined and (3.61) says that $K(\mathbf{x},\mathbf{y}) = \Phi(\mathbf{x})\cdot\Phi(\mathbf{y})$ and therefore

$$\begin{aligned}
\mathbf{v}^\top \mathbf{K}\mathbf{v} &= \sum_{i,j=1}^N v_i K(\mathbf{x}^i,\mathbf{x}^j)v_j = \sum_{i,j=1}^N v_i\Big(\sum_{k\in\mathcal{I}}\lambda_k\phi(\mathbf{x}^i)\phi_k(\mathbf{x}^j)v_j\Big)\\
&= \sum_{k\in\mathcal{I}}\lambda_k\sum_{i,j=1}^N v_i\phi_k(\mathbf{x}^i)\phi_k(\mathbf{x}^j)v_j = \sum_{k\in\mathcal{I}}\lambda_k\Big(\sum_{i=1}^N v_i\phi_k(\mathbf{x}^i)\Big)\Big(\sum_{j=1}^N v_j\phi_k(\mathbf{x}^j)\Big)\\
&= \sum_{k\in\mathcal{I}}\lambda_k\Big(\sum_{i=1}^N v_i\phi_k(\mathbf{x}^i)\Big)^2 \geq 0. \qquad \square
\end{aligned}$$

The next observation shows that $K(\mathbf{x},\mathbf{y})$ can be viewed as a similarity measure for a pair of features $\mathbf{x},\mathbf{y}$

$$K(\mathbf{x},\mathbf{y})^2 \leq K(\mathbf{x},\mathbf{x})\,K(\mathbf{y},\mathbf{y}). \tag{3.62}$$

In fact, applying Cauchy-Schwartz to (3.61) gives

$$|K(\mathbf{x},\mathbf{y})| \leq \Big(\sum_{k\in\mathcal{I}}\lambda_k\phi_k(\mathbf{x})^2\Big)^{1/2}\Big(\sum_{k\in\mathcal{I}}\lambda_k\phi_k(\mathbf{y})^2\Big)^{1/2} = K(\mathbf{x},\mathbf{x})^{1/2}K(\mathbf{y},\mathbf{y})^{1/2}.$$

Positive Definite Functions

An important class of kernels are of the form

$$K(\mathbf{x}, \mathbf{y}) = g(\|\mathbf{x} - \mathbf{y}\|_2), \quad \mathbf{x}, \mathbf{y} \in \mathcal{X}, \tag{3.63}$$

where g is a scalar valued function. Collections $g_j(\mathbf{x}) = g(\|\mathbf{x} - \mathbf{x}^j\|_2)$ are often called radial basis functions (RBFs). For instance, the Gaussian kernel is of this form

There is an elaborate Theorie about such radial basis systems that are positive definite which means that the corresponding kernels are psoitive definite in the above sense, see e.g. [7].

Reproducing Kernel Hilbert Space

This is all closely related to the concept of Reproducing Kernel Hilbert Space (RKHS). Roughly speaking this means the following: suppose $\mathcal{H}$ is a Hilbert space of functions on $\mathcal{X}$ with norm $\|v\|_{\mathcal{H}} = \langle \cdot, \cdot \rangle^{1/2}$ for which point evaluations are continuous, i.e., there exists a constant C such that for $f \in \mathcal{H}$ one has $|f(\mathbf{x})| \leq C\|f\|_{\mathcal{H}}$. Then, essentially as a consequence of the Riesz Representation Theorem there exists a reproducing kernel $K \in \mathcal{H} \otimes \mathcal{H}$ such that

$$\langle K(\cdot, \mathbf{x}), f \rangle_{\mathcal{H}} = f(\mathbf{x}), \tag{3.64}$$

that is, the kernel K represents the Dirac functional in $\mathcal{H}$.

L_2-spaces do not have a reproducing kernel, since point evaluation is not well-defined. But they contain subspaces for which this is the case.

There is an important result by Aronszajn-Moore [1] stating the converse, namely whenever one has a kernel $K(\cdot, \cdot)$ with certain properties, then there exists a Hilbert space for which this kernel is a reproducing kernel.

In the present context, this is relevant for the following reasons: kernel methods can be also used for regression in which case accuracy is naturally measured in an L_2-norm (with respect to the underlying probability density). However, the training is based on samples, i.e., point evaluations with which the estimators, of course, should comply. This is the situation we are in.

Mercer Kernels and RKHS

Theorem 18

Suppose that $K(\cdot,\cdot) \in L_2(\mathcal{X}\times\mathcal{X}, \otimes^2 p) \cap C(\mathcal{X}\times\mathcal{X})$ is a Mercer kernel. Then, there exists a Hilbert space $\mathcal{H}_K \subset L_2(\mathcal{X}\times\mathcal{X}, \otimes^2 p)$ with inner product $\langle\cdot,\cdot\rangle_K$ in which K is a reproducing kernel and point evaluations are continuous. $\mathcal{H}_K$ is generated by elements of the form $f(\mathbf{x}) = \sum_{i=1}^N f_i K(\mathbf{x}, \mathbf{x}^i)$, see the plug-ins for SVM classifiers in (3.51).

Step 1. Given (3.61), consider the bilinear form

$$\langle u, v\rangle_K := \sum_{k\in\mathcal{I}} \lambda_k^{-1} (u,\phi_k)_{\mathcal{X}} (v,\phi_k)_{\mathcal{X}}, \quad \text{where} \quad (f,g)_{\mathcal{X}} := \int_{\mathcal{X}} f(\mathbf{x})\overline{g(\mathbf{x})} dP(\mathbf{x}). \tag{3.65}$$

One easily checks that this is indeed a bilinear form, see Lecture I, page 6. Clearly,

$$\langle u, u\rangle_K = \sum_{k\in\mathcal{I}} \lambda_k^{-1} (u,\phi_k)_{\mathcal{X}}^2 \geq 0$$

Since $\lambda_k^{-1} \to \infty$ one may have $\langle u, u\rangle = \infty$. We are looking for a closed subspace $\mathcal{H}_K \subset L_2(\mathcal{X}, p)$ for which $\langle\cdot,\cdot\rangle_K$ is an inner product and a closed subspace $\|v\|_K^2 := \langle v, v\rangle_K < \infty$.

Reproducing Kernel Hilbert Space

Step 2. Reproducing property: with the above bilinear form $\langle \cdot\cdot \rangle_K$ one has

$$\langle K(\cdot, \mathbf{x}), K(\cdot, \mathbf{y}) \rangle_K = K(\mathbf{x}, \mathbf{y}) \tag{3.66}$$

In fact, by (3.61) and orthonormality of the ϕ_k, one has $(K(\cdot, \mathbf{x}), \phi_k)_{\mathcal{X}} = \lambda_k \phi_k(\mathbf{x})$ and therefore

$$\begin{aligned} \langle K(\cdot, \mathbf{x}), K(\cdot, \mathbf{y}) \rangle_K &= \sum_{k \in \mathcal{I}} \lambda_k^{-1} (K(\cdot, \mathbf{x}), \phi_k)_{\mathcal{X}} (K(\cdot, \mathbf{y}), \phi_k)_{\mathcal{X}} \\ &= \sum_{k \in \mathcal{I}} \lambda_k^{-1} \lambda_k^2 \phi_k(\mathbf{x}) \phi_k(\mathbf{y}) = K(\mathbf{x}, \mathbf{y}). \end{aligned} \tag{3.67}$$

Proposition 19

As in (3.64)*, consider the operator*

$$f(\mathbf{x}) \to (P_K f)(\mathbf{x}) := \langle K(\cdot, \mathbf{x}), f \rangle_K. \tag{3.68}$$

Then, for any $f \in L_2(\mathcal{X}, p)$ of the form $f(\mathbf{x}) = \sum_{i=1}^N f_i K(\mathbf{x}, \mathbf{x}^i)$, one has

$$(P_K f)(\mathbf{x}) = f(\mathbf{x}). \tag{3.69}$$

Moreover, $P_K : L_2(\mathcal{X}; p) \to \overline{\operatorname{span}\{\phi_k : k \in \mathcal{I}\}}$ is the L_2-orthogonal projection to the span of the ϕ_k, $k \in \mathcal{I}$.

Proof of Proposition 19: First, by the reproducing property (3.67),one has

$$(P_K f)(\mathbf{x}) = \sum_{j=1}^{N} f_j \langle K(\cdot, \mathbf{x}), K(\cdot, \mathbf{x}^j)\rangle_K \overset{(3.67)}{=} \sum_{j=1}^{N} f_j K(\mathbf{x}, \mathbf{x}^j) = f(\mathbf{x}), \tag{3.70}$$

which shows (3.69).

As for the rest of the claim, expanding K according to (3.61), using the definition of $\langle \cdot, \cdot \rangle_K$, and orthonormality of the ϕ_k, yields

$$\begin{aligned}(P_K f)(\mathbf{x}) &= \langle K(\cdot, \mathbf{x}), f\rangle_K = \sum_{k\in\mathcal{I}} \lambda_k^{-1} (K(\cdot, \mathbf{x}), \phi_K)_{\mathcal{X}} (f, \phi_k)_{\mathcal{X}} \overset{(3.61)}{=} \sum_{k\in\mathcal{I}} \lambda_k^{-1} \lambda_k \phi_k(\mathbf{x}) (f, \phi_k)_{\mathcal{X}} \\ &= \sum_{k\in\mathcal{I}} (f, \phi_k)_{\mathcal{X}} \phi_k(\mathbf{x}), \qquad \text{as claimed} \qquad \square\end{aligned}$$

Now for f of the form

$$f(\mathbf{x}) = \sum_{i=1}^{N} f_i K(\mathbf{x}, \mathbf{x}^i), \quad \mathbf{f} := (f_1 \ldots, f_N)^\top \in \ker \Big(\underbrace{(K(\mathbf{x}^i, \mathbf{x}^j))_{i,j=1}^{N}}_{=\mathbf{K}} \Big)^{\perp}, \tag{3.71}$$

one has for $\|\cdot\|_K = \langle \cdot, \cdot \rangle_K^{1/2}$, defined by (3.65),

$$\|f\|_K^2 = \langle f, f\rangle_K = \sum_{i,j=1}^{N} f_i f_j \langle K(\cdot, \mathbf{x}^i), K(\cdot \mathbf{x}^j)\rangle_K = \sum_{i,j=1}^{N} f_i f_j K(\mathbf{x}^i, \mathbf{x}^j) = \mathbf{f}^\top \mathbf{K}\mathbf{f} > 0.$$

Thus, $\|\cdot\|_K$ is a norm on the class of those functions.

Step 3. RKHS and Continuity:

- Let $\mathcal{H}_K$ be the closure of all f of the form (3.71) under the norm $\|\cdot\|_K$.
- As a closed subspace of $L_2(\mathcal{X}, p)$ it is a Hilbert space with inner product $\langle\cdot,\cdot\rangle_K$.
- Since by (3.68), (3.69), $f(\mathbf{x}) = \langle K(\cdot,\mathbf{x}), f\rangle_K$, Cauchy-Schwartz yields

$$|f(\mathbf{x})| = |\langle K(\cdot,\mathbf{x}), f\rangle_K| \le \|K(\cdot,\mathbf{x})\|_K \|f\|_K = \langle K(\cdot,\mathbf{x}), K(\cdot,\mathbf{x})\rangle_K^{1/2} \|f\|_K \overset{(3.67)}{=} K(\mathbf{x},\mathbf{x})^{1/2}\|f\|_K,$$

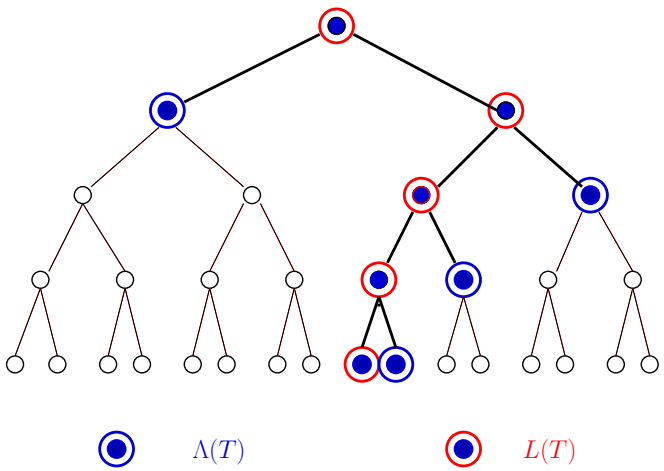
which says that point-evaluation is continuous in $\mathcal{H}_K$, i.e, $C(\mathcal{X}) \subset \mathcal{H}_K$ with a continuous embedding:

$$|f(\mathbf{x})| \le \sup_{\mathbf{x}\in\mathcal{X}} K(\mathbf{x},\mathbf{x})\|f\|_K \;\rightsquigarrow\; \|f\|_{L_\infty(\mathcal{X})} \le C\|f\|_K, \quad f \in \mathcal{H}_K. \tag{3.72}$$

A Bigger Picture ...

- SVMs work very well in high dimensions $d \gg 1$, especially when $d \gg N$, the number of samples. The computational complexity does not depend too strongly on d.
- An important emerging concept is Deep Neural Networks which contain linear SVMs as "nuclei" and can treat all regimes of d, [5].
- The performance of SVMs with regard to generalization and classification accuracy will be discussed in the next section.
- For lower spatial dimension d, especially when $N \gg d$, SVMs are not necessarily the method of choice. One of the many alternatives are decision trees or nearest neighbor methods.

A simple key ingredient: Partition Trees



- Split the domain $\mathcal{X}$ into a fixed number of cells - the "children";
- repeating this splitting for selected cells, creates a tree;
- the cells are the nodes. The leaf nodes form a partition of $\mathcal{X}$

The Idea of Decision Trees

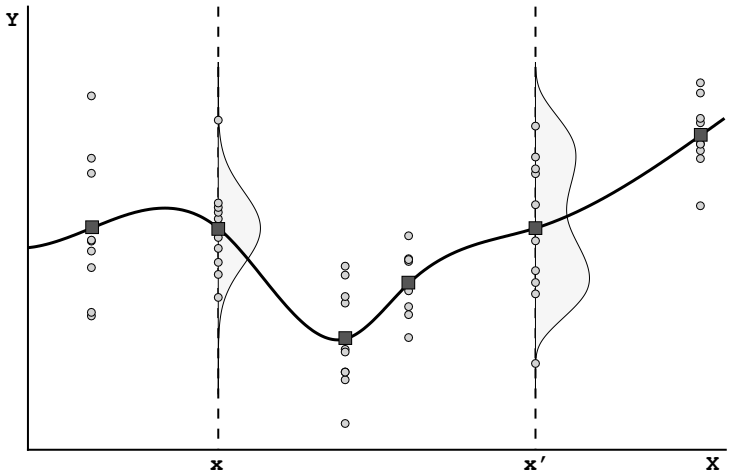
- Given a sample set $\mathfrak{Z}_N = \{(\mathbf{x}^1, y_1), \ldots, (\mathbf{x}^N, y_N)\} \subset \mathcal{Z} = \mathcal{X} \times \{-1, +1\}$, constructing a classifier can be viewed as finding a subset $S \subset \mathcal{X}$ such that for any new $\mathbf{x} \in \mathcal{X}$

$$\mathbf{x} \in S \quad \rightsquigarrow \quad y = +1, \quad \mathbf{x} \in S^c := \mathcal{X} \setminus S \quad \rightsquigarrow \quad y = -1.$$

- The set S should be generated from the data $\mathfrak{Z}_N$. The classifier has no loss if $\mathbf{x}^i \in S$ iff $y_i = 1$.
- The boundary ∂S is called the decision boundary, for any new query $\mathbf{x}$ the label depends on which side of the boundary ∂S this point is located.
- For linear SVMs the decision boundary is a hyperplane. For general kernel based SVMs the decision boundary is the zerolevel set of a trained linear combination of the kernel snapshots $K(\cdot, \mathbf{x}^i)$, $i = 1, \ldots, N$.
- Adaptive partition trees can be used to "zoom" into the decision boundary of the "ideal set" S^*.
- Such trees are called decision trees. With every leaf cell C of the tree - a cell in the partition generated by the tree - one associates a label $y(C) \in \{\pm 1\}$. For a new query $\mathbf{x}$, one finds the leaf cell C containing $\mathbf{x}$ and assignes to $\mathbf{x}$ the label $y(C)$.

There is more than classification ...

Regression not only asks for label decisions but for the whole functional relation behind the data:



P unknown probability measure on $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$

Factorization into conditional and marginal densities $dP(x, y) = dP(y|x)dP_X(x)$

Goal: estimate the regression function

$$f_p(x) := \int_Y y dP(y|x) = \mathbb{E}(y|x)$$

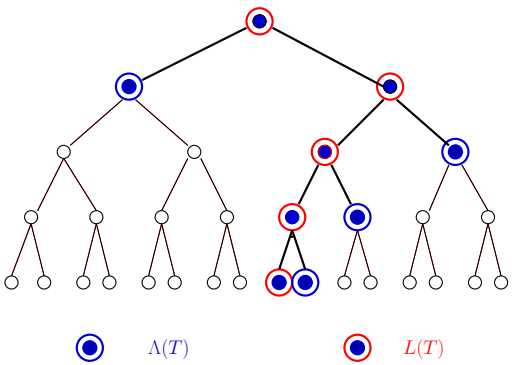
see Lecture II, (8.3)

Risk functional: $\mathcal{R}[f] := \int_Z (y - f(x))^2 dP \rightsquigarrow$

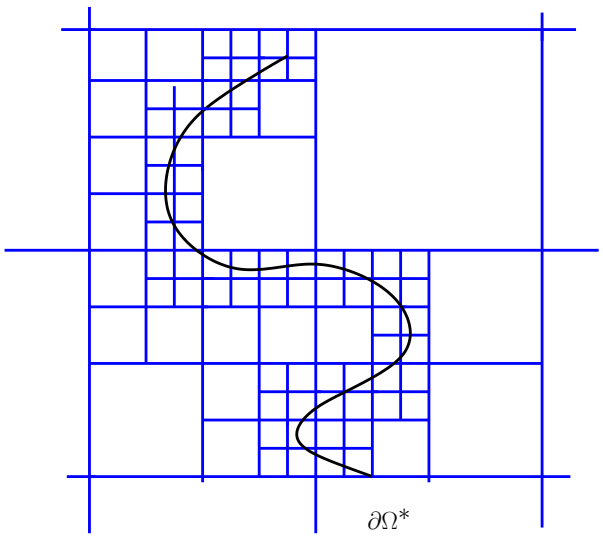
$$\mathcal{R}[f] = \mathcal{R}[f_\rho] + \|f - f_p\|^2_{L_2(X,p_X)}, \quad \|\cdot\| := \|\cdot\|_{L_2(X,p_X)}$$

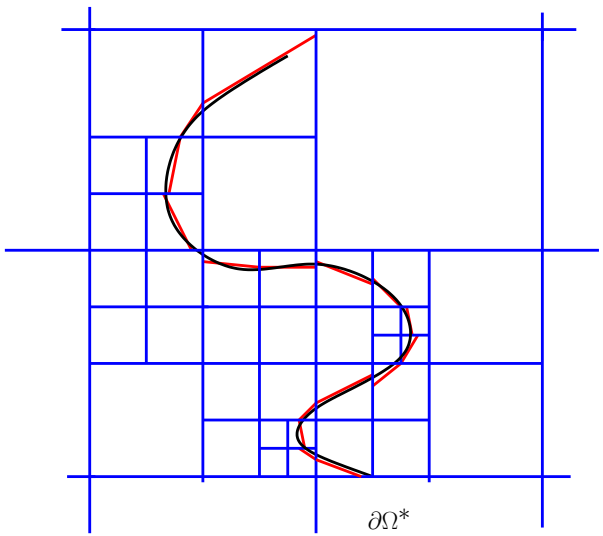
Task: construct an estimator $\hat{f}_\mathcal{Z}$, e.g. minimizing the least squares risk $\frac{1}{N}\sum_{i=1}^N \left(y_i - \hat{f}_\mathcal{Z}(\mathbf{x}^i)\right)^2$ using kernel SVMs, that approximates f_p well in $L_2(X, p_X)$.

The Idea of Decision Trees



- One should not keep refining until each leaf cell contains only one $\mathbf{x}^i$ $\rightsquigarrow$ overfitting
- The principles of generating the "right" tree are connected with the learning principles discussed in the next section.
- This concerns e.g. complexity penalization and model selection. An important variant that first refines too much and then prunes back is CART, [4]
- There are different ways of assigning a label to a leaf cell based on several data contained in that cell. The simplest one is to take the sign of the average of the labels in that cell. Alternatively, one could use an SVM for the data in that cell.





References I

[1] N. Aronszajn, Theory of Reproducing Kernels, Transactions of the American Mathematical Society. 68 (3) (1950), 337–404. doi:10.1090/S0002- 9947-1950-0051437-7

[2] D. P. Bertsekas. Nonlinear Programming. Athena Scientific, Belmont, MA, 1995.

[3] S. Boyd, L. Vandenberghe, *Convex Optimization, Cambridge, University Press, 2009*

[4] L.Breiman, J.H. Friedman, R.A. Olshen, C.J. Stone, Classification and Regression Trees. Wadsworth, Belmont, 1984, CA. MR0726392

[5] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, Cambridge Massachusetts, 2016.

[6] J. Mercer. Functions of positive and negative type and their connection with the theory of integral equations. Philosophical Transactions of the Royal Society, London, A 209:415–446, 1909.

[7] C.A. Michelli, Interpolation of scattered data: Distance matrices and conditionally positive definite functions, Constructive Approximation, 2 (1986), 11–22.

[8] J.Nocedal, S.J. Wright, *Numerical Optimization, 2nd Edition, Springer, 2006.*

[9] A. Statnikov, D. Hardin. I. Guyon, C.F. Aliferis, A gentle Introduction to Support Vector Machines in Biomedicine, AMIA 2009, Sab Fracisco, Nov.14-18,Biomedical and Health Informatics: From Foundations to Applications

[10] B. Schölkopf and A. J. Smola. Learning with Kernels. MIT Press, Cambridge, MA, 2002.

[11] V. N. Vapnik. Statistical Learning Theory. Wiley, New York, 1998.