Lecture V: Support Vector Machines

Marina Meilă
mmp@stat.washington.edu

Department of Statistics
University of Washington

November, 2022

Linear SVM's

The margin and the expected classification error
Maximum Margin Linear classifiers
Linear classifiers for non-linearly separable data

Non linear SVM

The "kernel trick"
Kernels
Prediction with SVM

Extensions

L_1 SVM
Multi-class and One class SVM
SV Regression

Reading HTF Ch.: Ch. 12.1–3, Murphy Ch.: Ch 14 (14.1,14.2–14.2.4 kernels, 14.4 and equations (14.28,14.29) kernel trick, 14.5.1.–3 Support Vector Machines), Bach Ch.: 7.1–7.4, 7.7
Additional Reading: C. Burges - "A tutorial on SVM for pattern recognition"
These notes: Appendices (convex optimization) are optional.

The margin and the expected classification error

Theorem Let $\mathcal{F} = \{\mathrm{sgn}\,(w^T x),\ ||w|| \leq \Lambda,\ ||x|| \leq R\}$ and let $\rho > 0$ be any "margin". Then for any $f \in \mathcal{F}$, w.p $1-\delta$ over training sets

$$L_{01}(f) \;\leq\; \hat{L}_\rho + \sqrt{\frac{c}{n}\left(\frac{R^2\Lambda^2}{\rho^2}\ln n^2 + \ln\frac{1}{\delta}\right)} \tag{5}$$

where c is a universal constant and $\hat{L}_\rho$ is the fraction of the training examples for which

$$y^i w^T x_i < \rho \tag{6}$$

- a data point i that satisfies (6) for some ρ is called a **margin error**
- For $\rho = 0$ the margin error rate $\hat{L}_\rho$ is equal to $\hat{L}_{01}$

Maximum Margin Linear classifiers

Support Vector Machines appeared from the convergence of Three Good Ideas

Assume (for the moment) that the data are linearly separable.

- Then, there are an infinity of linear classifiers that have $\hat{L}_{01} = 0$. Which one to choose?

irst idea Select the classifier that has **maximum margin** ρ on the training set.

  - For any parameters (w, b) that perfectly classify the data $\hat{L}(w, b) = 0$.
  - Among these, the best (w, b) is the one that minimizes ρ in 5
  - Hence, we should choose

$$\underset{\rho, w, b : \hat{L}(w,b)=0}{\text{argmax}} \rho, \quad \text{s.t. } d(x, H_{w,b}) \geq \rho \text{ for } i = 1 : n, \tag{7}$$

where $d()$ denotes the Euclidean distance and $H_{w,b} = \{\, x \,|\, w^T x + b = 0\}$ is the decision boundary of the linear classifier.

- Because $d(x, H_{w,b}) = \frac{|w^T x + b|}{||w||}$ (proof in a few slides) (7) becomes

$$\underset{\rho, w, b : \hat{L}(w,b)=0}{\text{argmax}} \rho, \quad \text{s.t. } \frac{|w^T x^i + b|}{||w||} \geq \rho \text{ for } i = 1 : n, \tag{8}$$

Maximum Margin Linear classifiers

We continue to transform (8)

- If all data correctly classified, then $y^i(w^Tx^i + b) = |w^Tx^i + b|$. Therefore (8) has the same solution as

$$\underset{\rho,w,b}{\text{argmax}}\, \rho, \quad \text{s.t. } \frac{y^i(w^Tx^i + b)}{||w||} \geq \rho \text{ for } i = 1 : n, \tag{9}$$

- Note now that the problem (9) is underdetermined. Setting $w \leftarrow Cw, b \leftarrow Cb$ with $C > 0$ does not change anything.
- We add a cleverly chosen constraint to remove the indeterminacy; this is $||w|| = 1/\rho$, which allows us to eliminate the variable ρ. We get

$$\underset{w,b}{\text{argmax}}\, \frac{1}{w}, \quad \text{s.t. } y^i(w^Tx^i + b) \geq 1 \text{ for } i = 1 : n, \tag{10}$$

Note: the successive problems (7),(8),(9),... are equivalent in the sense that their optimal solution is the same.

Alternative derivation of (10)

st idea Select the classifier that has maximum margin on the training set, by the alternative definition of margin.
Formally, define $\min_{i=1:n} y^i f(x^i)$ be the **margin of classifier f on $\mathcal{D}$**. Let $f(x) = w^T x + b$, and choose w, b that

$$\text{maximize}_{w \in \mathbb{R}^n, b \in \mathbb{R}} \min_{i=1:n} y^i (w^T x^i + b) \; s.t. \; \hat{L}(w, b) = 0$$

- Remarks
  - (if data is linearly separable), there exist classifiers with margins > 0
  - one can arbitrarily increase the margin of such a classifier by multiplying w and b by a positive constant.
  - Hence, we need to "normalize" the set of candidate classifiers by requiring instead

$$\text{maximize} \min_{i=1:n} d(x, H_{w,b}), \; \text{s.t. } y^i (w^T x^i + b) \geq 1 \text{ for } i = 1:n, \tag{11}$$

    where $d()$ denotes the Euclidean distance and $H_{w,b} = \{ x \mid w^T x + b = 0 \}$ is the decision boundary of the linear classifier.
  - Under the conditions of (11), because there are points for which $|w^T x + b| = 1$, maximizing $d(x, H_{w,b})$ over w, b for such a point is the same as

$$\max_{w,b} \frac{1}{||w||}, \; \text{s.t.} \min_i y_i (w^T x + b) = 1 \tag{12}$$

Second idea

The **Second idea** is to formulate (10) as a **quadratic** optimization problem.

$$\min_{w,b} \frac{1}{2}||w||^2 \text{ s.t } y^i(w^T x^i + b) \geq 1 \text{ for all } i = 1 : n \tag{13}$$

This is the **Linear SVM (primal) optimization problem**

- This problem has a strongly convex **objective** $||w||^2$, and **constraints** $y^i(w^T x^i + b)$ linear in (w, b).
- Hence this is a convex problem, and can be studied with the tools of convex optimization.

The distance of a point x to a hyperplane $H_{w,b}$

$$d(x, H_{w,b}) \quad = \quad \frac{|w^T x + b|}{||w||} \tag{14}$$

Intuition: denote

$$\tilde{w} \;=\; \frac{w}{||w||},\; \tilde{b} \;=\; \frac{b}{||w||},\; x' \;=\; \tilde{w}^T x. \tag{15}$$

Obviously $H_{w,b} = H_{\tilde{w},\tilde{b}}$, and x' is the length of the projection of point x on the direction of w. The distance is measured along the normal through x to H; note that if $x' = -\tilde{b}$ then $x \in H_{w,b}$ and $d(x, H_{w,b}) = 0$; in general, the distance along this line will be $|x' - (-\tilde{b})|$.

Optimization with Lagrange multipliers

[2] The **Lagrangean** of (13) is

$$L(w, b, \alpha) \;=\; \frac{1}{2}||w||^2 - \sum_i \alpha_i [y^i(w^T x^i + b) - 1]. \qquad (16)$$

[KKT conditions]

At the optimum of (13)

$$w \;=\; \sum_i \alpha_i y^i x^i \quad \text{with } \alpha_i \geq 0 \qquad (17)$$

and $b = y^i - w^T x^i$ for any i with $\alpha_i > 0$.

- **Support vector** is a data point x^i such that $\alpha_i > 0$.
- According to (17), the final decision boundary is determined by the support vectors (i.e. does not depend explicitly on any data point that is not a support vector).

[2] The derivations of these results are in the Appendix

Dual SVM optimization problem

- Any convex optimization problem has a **dual** problem. In SVM, it is both illuminating and practical to solve the dual problem.
- The dual to problem (13) is

$$\max_{\alpha_{1:n}} \sum_i \alpha_i - \frac{1}{2}\sum_i \alpha_i\alpha_j y^i y^j {x^i}^T x_j \text{ s.t } \alpha_i \geq 0 \text{ for all } i \text{ and } \sum_i \alpha_i y^i = 0. \quad (18)$$

- This is a quadratic problem with n variables on a convex domain.
- Dual problem in matrix form
  - Denote $\alpha = [\alpha_i]_{i=1:n}$, $y = [y^i]_{i=1:n}$, $G_{ij} = {x^i}^T x_j$, $\bar{G}_{ij} = y^i y^j {x^i}^T x_j$, $G = [G_{ij}] \in \mathbb{R}^{n\times n}$, $\bar{G} = [\bar{G}_{ij}] \in \mathbb{R}^{n\times n}$.

$$\max_{\alpha\in\mathbb{R}^n} 1^T\alpha - \frac{1}{2}\alpha^T\bar{G}\alpha \quad \text{s.t } \alpha \succeq 0 \text{ and } y^T\alpha = 0. \quad (19)$$

- $g(\alpha) = 1^T\alpha - \frac{1}{2}\alpha^T\bar{G}\alpha$ is the **dual objective function**
- G is called the **Gram matrix** of the data. Note that $\bar{G} = \mathrm{diag}\{y^{1:n}\}^T G \mathrm{diag}\{y^{1:n}\}$.

- At the dual optimum
  - $\alpha_i > 0$ for constraints that are satisfied with equality, i.e. **tight**
  - $\alpha_i = 0$ for the **slack** constraints

Non-linearly separable problems and their duals

The **C-SVM**

$$\text{minimize}_{w,b,\xi} \quad \frac{1}{2}||w||^2 + C\sum_i \xi_i \tag{20}$$

$$\text{s.t.} \quad y^i(w^T x^i + b) \ \geq \ 1 - \xi_i$$

$$\xi_i \ \geq \ 0$$

In the above, ξ_i are the slack variables. Dual[3]:

$$\text{maximize}_\alpha \quad \sum_i \alpha_i - \frac{1}{2}\sum_i \alpha_i \alpha_j y^i y_j x^{iT} x_j \tag{21}$$

$$\text{s.t.} \quad C \ \geq \ \alpha_i \geq 0 \text{ for all } i$$

$$\sum_i \alpha_i y^i \ = \ 0$$

$\Rightarrow$ two types of SV

- $\alpha_i < C$ data point x^i is "on the margin" $\Leftrightarrow$ $y^i(w^T x^i + b) = 1$ (original SV)
- $\alpha_i = C$ data point x^i cannot be classified with margin 1 (**margin error)**
  $\Leftrightarrow$ $y^i(w^T x^i + b) < 1$

[3]Lagrangean $L(w, b, \xi, \alpha, \mu) \ = \ \frac{1}{2}||w||^2 + C\sum_i \xi_i - \sum_i \alpha_i[y^i(w^T x^i + b) - 1 + \xi_i] - \sum_i \mu_i \xi_i$ with $\alpha_i \geq 0, \ \xi_i \geq 0, \ \mu_i \geq 0$

The ν-SVM

$$\text{minimize}_{w,b,\xi,\rho} \quad \frac{1}{2}||w||^2 - \nu\rho + \frac{1}{n}\sum_i \xi_i \tag{22}$$
$$\text{s.t.} \quad y^i(w^T x^i + b) \ \geq \ \rho - \xi_i \tag{23}$$
$$\xi_i \ \geq \ 0 \tag{24}$$
$$\rho \ \geq \ 0 \tag{25}$$

where $\nu \in [0, 1]$ is a parameter.
Dual[4]:

$$\text{maximize}_\alpha \quad -\frac{1}{2}\sum_i \alpha_i \alpha_j y^i y^j x^{iT} x^j \tag{26}$$
$$\text{s.t.} \quad \frac{1}{n} \ \geq \ \alpha_i \geq 0 \text{ for all } i \tag{27}$$
$$\sum_i \alpha_i y^i \ = \ 0 \tag{28}$$
$$\sum_i \alpha_i \ \geq \ \nu \tag{29}$$

Properties If $\rho > 0$ then:

- ν is an upper bound on #margin errors/n (if $\sum_i \alpha_i = \nu$)
- ν is a lower bound on #(original support vectors + margin errors)/n
- ν-SVM leads to the same w, b as C-SVM with $C = 1/\nu$

[4] Lagrangean $L(w, b, \xi, \rho, \alpha, \mu, \delta) \ = \ \frac{1}{2}||w||^2 - \nu\rho + \frac{1}{n}\sum_i \xi_i - \sum_i \alpha_i [y^i(w^T x^i + b) - \rho + \xi_i] - \sum_i \mu_i \xi_i - \delta\rho$ with $\alpha_i \geq 0, \ \delta \geq 0, \ \mu_i \geq 0$

A simple error bound

$$L_{01}(f_n) \;\leq\; E\left[\frac{\#\text{support vectors of } f_{n+1}}{n+1}\right] \tag{30}$$

where f_n denotes the SVM trained on a sample of size n.

Exercise Use the Homework 6 to prove this result.

Non-linear SVM

How to use linear classifier on data that is not linearly separable?

An old trick

1. Map the data $x^{1:n}$ to a higher dimensional space

$$x \to z = \phi(x) \in \mathcal{H}, \text{with dim } \mathcal{H} >> n.$$

2. Construct a linear classifier $w^T z + b$ for the data in $\mathcal{H}$

   In other words, we are implementing the non-linear classifier

$$f(x) \;=\; w^T\phi(x) + b \;=\; w_1\phi_1(x) + w_2\phi_2(x) + \ldots + w_m\phi_m(x) + b \tag{31}$$

Example

- Data $\{(x, y)\}$ below are not linearly separable

x		y	z		
-1	-1	1	-1	-1	1
-1	1	-1	-1	1	-1
1	-1	-1	1	-1	-1
1	1	1	1	1	1

- We map them to 3 dimensions by

$$z = \phi(x) = [x_1 \; x_2 \; x_1 x_2].$$

- Now the classes can be separated by the hypeplane $z_3 = 0$ (which happens to be the maximum margin hyperplane). Hence,
  - $w = [\,0\;0\;1\,]$ (a vector in $\mathcal{H}$)
  - $b = 0$
  - and the classification rule is $f(\phi(x)) = w^T\phi(x) + b$.
- If we write f as a function of the original x we get

$$f(x) \;=\; x_1 x_2$$

a quadratic classifier.

Non-linear SV problem

- Primal problem minimize $\frac{1}{2}||w||^2$ s.t $y^i(w^T\phi(x^i)+b)-1\geq 0$ for all i.
- Dual problem

$$\max_{\alpha_{1:n}} \sum_i \alpha_i - \frac{1}{2}\sum_i \alpha_i\alpha_j \underbrace{y^i y_j \phi(x^i)^T\phi(x_j)}_{\bar{G}_{ij}} \text{ s.t. } \alpha_i \geq 0 \text{ for all } i \text{ and } \sum_i y^i\alpha_i = 0 \quad (32)$$

$$G_{ij} = \phi(x^i)^T\phi(x^j) \quad \text{and} \quad \bar{G} = \operatorname{diag}\{y^{1:n}\}^T G \operatorname{diag}\{y^{1:n}\}. \quad (33)$$

- $\bar{G}_{ij}$ has been redefined in terms of ϕ
- Dual problem

$$\max_{\alpha} 1^T\alpha - \frac{1}{2}\alpha^T\bar{G}\alpha \quad \text{s.t. } \alpha_i \geq 0,\ y^T\alpha = 0 \quad (34)$$

- Same as (19)!

The "Kernel Trick"

d idea The result (34) is the celebrated **kernel trick** of the SVM literature. We can make the following remarks.

1. The ϕ vectors enter the SVM optimization problem only trough the Gram matrix, thus only as the scalar products $\phi(x^i)^T\phi(x_j)$. We denote by $K(x, x')$ the function

$$K(x, x') = K(x', x) = \phi(x)^T\phi(x') \tag{35}$$

   K is called the **kernel** function. If K can be computed efficiently, then the Gram matrix G can also be computed efficiently. This is exactly what one does in practice: we choose ϕ implicitly by choosing a kernel K. Hereby we also ensure that K can be computed efficiently.
2. Once G is obtained, the SVM optimization is independent of the dimension of x and of the dimension of $z = \phi(x)$. The complexity of the SVM optimization depends only on n the number of examples. This means that we can choose a very high dimensional ϕ without any penalty on the optimization cost.
3. Classifying a new point x. As we know, the SVM classification rule is

$$f(x) = w^T\phi(x) + b = \sum_{i=1}^{n}\alpha_i y^i \phi(x^i)^T\phi(x) = \sum_{i=1}^{n}\alpha_i y^i K(x^i, x) \tag{36}$$

   Hence, the classification rule is expressed in terms of the support vectors and the kernel only. No operations other than scalar product are performed in the high dimensional space H.

Kernels

The previous section shows why SVMs are often called **kernel machines**. If we choose a kernel, we have all the benefits of a mapping in high dimensions, without ever carrying on any operations in that high dimensional space. The most usual kernel functions are

$K(x, x') = (1 + x^T x')^p$	the polynomial kernel of degree p
$K(x, x') = \tanh(\sigma x^T x' - \beta)$	the "neural network" kernel
$K(x, x') = e^{-\frac{\|\|x-x'\|\|^2}{\sigma^2}}$	the Gaussian or **radial basis function** (RBF) kernel it's ϕ is ∞-dimensional

The Mercer condition

- How do we verify that a chosen K is is a valid kernel, i.e that there exists a ϕ so that $K(x, x') = \phi(x)^T \phi(x')$?
- This property is ensured by a positivity condition known as the Mercer condition.

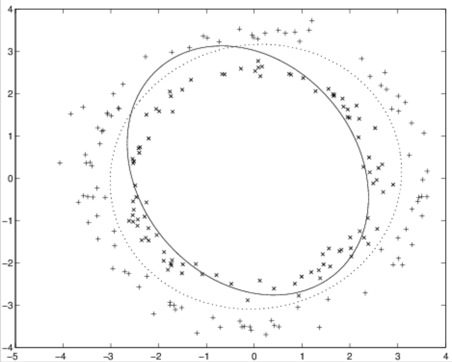
Mercer condition

Let $(\mathcal{X}, \mu)$ be a finite measure space. A symmetric function $K : \mathcal{X} \times \mathcal{X}$, can be written in the form $K(x, x') = \phi(x)^T \phi(x')$ for some $\phi : \mathcal{X} \to \mathcal{H} \subset \mathbb{R}^m$ iff

$$\int_{\mathcal{X}^2} K(x, x') g(x) g(x') d\mu(x) d\mu(x') \geq 0 \quad \text{for all } g \text{ such that } ||g(x)||_{L_2} < \infty \tag{37}$$

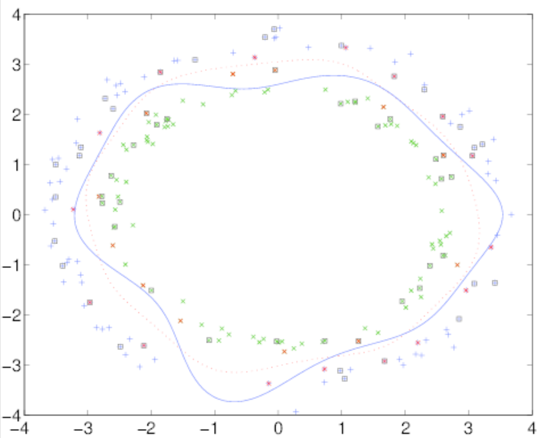
- In other words, K must be a positive semidefinite operator on L_2.
- If K satisfies the Mercer condition, there is no guarantee that the corresponding ϕ is unique, or that it is finite-dimensional.

Quadratic kernel



- C-SVM, polynomial degree 2 kernel, $n = 200$, $C = 10000$
- The two ellipses show that a constant shift to the data ($x^i \leftarrow x^i + v$, $v \in \mathbb{R}^n$) can affect non-linear kernel classifiers.

RBF kernel and Support Vectors



Prediction with SVM

- Estimating b
  - For any i support vector, $w^T x^i + b = y^i$ because the classification is tight
  - Alternatively, if there are slack variables, $w^T x^i + b = y^i(1 - \xi_i)$
  - Hence, $b = y^i(1 - \xi_i) - w^T x^i$
  - For non-linear SVM, where w is not known explicitly, $w = \sum_j \alpha_j y^j \phi(x_j)$. Hence, $b = y^i(1 - \xi_i) - \sum_{j=1}^n \alpha_j y^j K(x^i, x^j)$ for any i support vector
- Given new x

$$\hat{y}(x) = \mathrm{sgn}(w^T x + b) = \mathrm{sgn}\left(\sum_{i=1}^{n} \alpha_i y^i K(x^i, x) + b\right). \quad (38)$$

L1-SVM

- If the regularization $||w||^2$, based on l_2 norm, is replaced with the l_1 norm $||w||_1$, we obtain what is known as the **Linear L1-SVM**

$$\min_{w,b} ||w||_1 + C\sum_i \xi_i \quad \text{s.t}\, y^i(w^Tx^i + b) \geq 1 - \xi_i,\ \xi_i \geq 0 \text{ for all } i = 1:n \tag{39}$$

- The use of the l_1 norm promotes sparsity in the entries of w

- The **Non-linear L1-SVM** is

$$f(x) = \sum_i (\alpha_i^+ + \alpha_i^-) y^i K(x_i, x) + b \quad \text{classifier} \tag{40}$$

$$\min_{\alpha_\pm, b} \sum_i (\alpha_i^+ + \alpha_i^-) + C\sum_i \xi_i \quad \text{s.t}\, y^i f(x^i) \geq 1 - \xi_i,\ \xi_i, \alpha_i^\pm \geq 0 \text{ for all } i = 1:n \tag{41}$$

- This formulation enforces $\alpha_i^+ = 0$ or $\alpha_i^- = 0$ for all i. If we set $w_i = \alpha_i^+ - \alpha_i^-$, we can write $f(x) = \sum_i w_i y^i K(x^i, x) + b$, a linear classifier in the non-linear features $K(x^i, x)$.
- The L1-SVM problems are Linear Programs
- The dual L1-SVM problems are also linear programs
- The L1-SVM is no longer a Maximum Margin classifier

Multi-class and One class SVM

Multiclass SVM

For a problem with K possible classes, we construct K separating hyperplanes $w_r^T x + b_r = 0$.

$$\text{minimize} \quad \frac{1}{2}\sum_{r=1}^{K}||w_r||^2 + \frac{C}{n}\sum_{i,r}\xi_{i,r} \tag{42}$$

$$\text{s.t.} \quad w_{y^i}^T x^i + b_{y^i} \geq w_r^T x^i + b_r + 1 - \xi_{i,r} \quad \text{for all } i = 1:n,\ r \neq y^i \tag{43}$$

$$\xi_{i,r} \geq 0 \tag{44}$$

One-class SVM This SVM finds the “support regions” of the data, by separating the data from the origin by a hyperplane. It’s mostly used with the Gaussian kernel, that projects the data on the unit sphere. The formulation below is identical to the ν-SVM where all points have label 1.

$$\text{minimize} \quad \frac{1}{2}||w||^2 - \nu\rho + \frac{1}{n}\sum_{i}\xi_i \tag{45}$$

$$\text{s.t.} \quad w^T x^i + b \geq \rho - \xi_i \tag{46}$$

$$\xi_i \geq 0 \tag{47}$$

$$\rho \geq 0 \tag{48}$$

SV Regression

The idea is to construct a "tolerance interval" of $\pm\epsilon$ around the regressor f and to penalize data points for being outside this tolerance margin. In words, we try to construct the smoothest function that goes within ϵ of the data points.

$$\text{minimize} \quad \frac{1}{2}||w||^2 + C\sum_i(\xi_i^+ + \xi_i^-) \tag{49}$$

$$\text{s.t.} \quad \epsilon + \xi_i^+ \;\geq\; w^T x^i + b - y^i \;\geq\; -\epsilon - \xi_i^- \tag{50}$$

$$\xi_i^\pm \;\geq\; 0 \tag{51}$$

$$\rho \;\geq\; 0 \tag{52}$$

The above problem is a linear regression, but with the kernel trick we obtain a kernel regressor of the form $f(x) = \sum_i(\alpha_i^- - \alpha_i^+)K(x^i, x) + b$

Convex optimization in a nutshell

A set $D \subseteq \mathbb{R}^n$ is **convex** iff for every two points $x^1, x^2 \in D$ the line segment defined by $x = tx^1 + (1-t)x^2,\ t \in [0,1]$ is also in D. A function $f : D \to R$ is **convex** iff, for any $x^1, x^2 \in D$ and for any $t \in [0,1]$ for which $tx^1 + (1-t)x^2 \in D$ the following inequality holds

$$f(tx^1 + (1-t)x^2) \ \leq \ tf(x^1) + (1-t)f(x^2) \tag{53}$$

If f is convex, then the set $\{\, x \,|\, f(x) \leq \ c \,\}$ is convex for any value of c. Convex functions defined on convex sets have very interesting properties which have engendered the field called **convex optimization**.

The optimization problem

$$\begin{aligned} &\min_x \ f_0(x) \\ &\text{s.t. } f_i(x) \ \leq \ 0 \ \text{ for } i = 1, \ldots m \end{aligned} \tag{54}$$

is a **convex optimization problem** if all the functions f, f_i are convex. Note that in this case the **feasible domain** $A = \bigcap_i \{\, x \,|\, f_i(x) \leq \ 0 \,\}$ is a convex set.

It is known that if A has a non empty interior then the convex optimization problem has at most one optimum x^*. If A is also bounded, x^* always exists.
Assuming that x^* exists, there are two possible cases: (1) The **unconstrained minimum** of f_0 lies in A. In this case, the optimum can be found by solving the equations $\frac{\partial f_0}{\partial x} = 0$. (2) The unconstrained minimum of f_0 lies outside A. Figure 1 depicts what happens at the optimum x^* in this case.

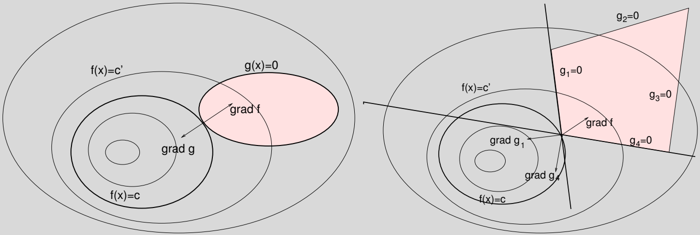


Figure: (a) One constraint optimization. (b) Four constraint optimization. At the optimum only constraints g_1, g_4 are active. f denotes the objective (f_0 in text) and g denote the constraints (f_i in text).

Assume there is only one constraint f_1. The domain A is the inside of the curve $f_1(x) = 0$. The optimum x^* is the point where a level curve $f_0(x) = c$ is tangent to $f_1 = 0$ from the outside. In this point, the gradients of two curves lie along the same line, pointing in opposite directions. Therefore, we can write $\frac{\partial f_0}{\partial x} = -\alpha \frac{\partial f_1}{\partial x}$. Equivalently, we have that at x^*, $\frac{\partial f_0}{\partial x} + \alpha \frac{\partial f_1}{\partial x} = 0$. Note that this is a necessary but not a sufficient condition. The above set of equations represents the **Karush-Kuhn-Tucker optimality conditions (KKT)**.

With more than one constraint, the KKT conditions are equivalent to requiring that the gradient of f_0 lies in the subspace spanned by the gradients of the constraints.

$$\frac{\partial f_0}{\partial x} \;=\; -\sum_i \alpha_i \frac{\partial f_i}{\partial x} \;\text{ with } \alpha_i \geq 0 \;\text{ for all } i \tag{55}$$

Note that if a certain constraint f_i does not participate in the boundary of D at x^*, i.e if the constraint is not **active**, the coefficient α_i should be 0. Equation (55) can be rewritten as

$$\frac{\partial}{\partial x}\underbrace{[f_0(x) + \sum_i \alpha_i f_i(x)]}_{L(x,\alpha)} \;=\; 0 \;\text{ for some } \alpha_i \geq 0 \text{ for } i = 1, \ldots m \tag{56}$$

The optimum x^* has to satisfy the equation above. The new function $L(x, \alpha)$ is the **Lagrangean** of the problem and the variables α_i are called **Lagrange multipliers**. The Lagrangean is convex in x and **affine** (i.e linear + constant) in α.

The dual problem Define the function

$$g(\alpha) \;=\; \inf_x \; L(x,\alpha) \;\; \alpha \;=\; (\alpha_i)_i,\; \alpha_i \geq 0 \tag{57}$$

In the above, the infimum is over all the values of x for which f_0, f_i are defined, not just A (but everything still holds if the infimum is only taken over A). Two facts are important about g

- $g(\alpha) \leq L(x,\alpha) \leq f(x)$ for any $x \in A,\; \alpha \geq 0$, i.e g is a lower bound for f_0, and implicitly for the optimal value $f_0(x^*)$, for any value of $\alpha \geq 0$.
- $g(\alpha)$ is concave (i.e $-g(\alpha)$ is convex).

We also can derive from (56) that if x^* exists then for an appropriate value α^* we have

$$g(\alpha^*) \;=\; L(x^*,\alpha^*) \;=\; f_0(x^*) + 0 \tag{58}$$

and therefore $g(\alpha^*)$ must be the unique maximum of $g(\alpha)$. The second term in L above is zero because x^* is on the boundary of A; hence for the active constraints $f_i(x^*) = 0$ and for the inactive constraints $\alpha_i^* = 0$.

This surprising relationship shows that by solving the **dual problem**

$$\max g(\alpha) \quad \text{s.t } \alpha \geq 0 \tag{59}$$

we can obtain the values α^* that plugged into (55 will allow us to find the solution x^* to our original (**primal**) problem. The constraints of the dual are simpler than the constraints of the primal. In practice, it is surprisingly often possible to compute the function $g(\alpha)$ explicitly. Below we give a simple example thereof. This is also the case of the SVM optimization problem, which will be discussed in section 5.

A simple optimization example

Take as an example the convex optimization problem

$$\min \frac{1}{2}x^2 \quad \text{s.t} \;\; x+1 \leq 0 \tag{60}$$

By inspection the solution is $x^* = -1$.
Let us now apply to it the convex optimization machinery. We have

$$L(x,\alpha) \;=\; \frac{1}{2}x^2 + \alpha(x+1) \tag{61}$$

defined for $x \in R$ and $\alpha \geq 0$.

$$\begin{aligned} g(\alpha) &= \inf_x \left[\frac{1}{2}x^2 + \alpha(x+1)\right] && (62)\\ &= \inf_x \left[\frac{1}{2}(x+\alpha)^2 - \frac{1}{2}\alpha^2 + \alpha\right] && (63)\\ &= -\frac{1}{2}\alpha^2 + \alpha && (64)\\ &= \frac{1}{2}\alpha(2-\alpha) \quad \text{attained for } x \;=\; -\alpha && (65)\end{aligned}$$

The dual problem is

$$\max \frac{1}{2}\alpha(2-\alpha) \;\; \text{s.t} \; \alpha \geq 0 \tag{66}$$

and its solution is $\alpha = 1$ which, using equation (65) leads to $x = -1$.
From the KKT condition

$$\frac{\partial L}{\partial x} \;=\; x + \alpha \;=\; 0 \tag{67}$$

we also obtain $x^* = -\alpha^* = -1$.

Figure 2 depicts the function L. Note that L is convex in x (a parabola) and that along the α axis the graph of L consists of lines. The areas of L that fall outside the admissible domain $x \leq -1,\ \alpha \geq 0$ are in flat (green) color. The crossection $L(x, \alpha = 0)$ represents the plot of f. The constrained minimum of f is at $x = -1$, the unconstrained one is at $x = 0$ outside the admissible domain. Note that $g(\alpha) = L(-\alpha, \alpha)$ is concave, and that in the admissible domain it is always below the graph of f. The (red) dot is the optimum (x^*, α^*), which represents a **saddle point** for h. The line $L(x = -1, \alpha)$ is horizontal (because $f_1 = x + 1 = 0$) and thus $L(x^*, \alpha^*) \ = \ L(x^*,) \ = \ f(x^*)$.

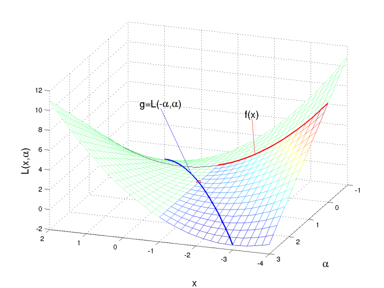


Figure: The surface $L(x, \alpha)$ for the problem min $\frac{1}{2}x^2$ s.t $x + 1 \leq 0$.

The SVM solution by convex optimization

The SVM optimization problem

$$\min_{w} \frac{1}{2}||w||^2 \quad \text{s.t. } y^i(w^T x^i + b) \geq 1 \text{ for all } i \tag{68}$$

is a convex (quadratic) optimizaton problem where

$$f_0(w, b) = \frac{1}{2}||w||^2 \tag{69}$$

$$f_i(w, b) = -y^i w^T x^i + 1 - y^i b \tag{70}$$

Hence,

$$L(w, b, \alpha) = \frac{1}{2}||w||^2 + \sum_i \alpha_i [1 - y^i b - y^i x^{iT} w] \tag{71}$$

Equating the partial derivatives of h w.r.t w, b with 0 we get

$$\frac{\partial L}{\partial w} = w - \sum_i \alpha_i y^i x^i \tag{72}$$

$$\frac{\partial L}{\partial b} = \sum_i \alpha_i y^i \tag{73}$$

or, equivalently

$$w = \sum_i \alpha_i y^i x^i \quad 0 = \sum_i \alpha_i y^i \tag{74}$$

Hence, the normal w to the optimal separating hyperplane is a linear combination of data points.

Sparsity of solution Moreover, we know that only those α_i corresponding to active constraints will be non-zero. In the case of SVM, these represent points that are classified with $yi(w^T x^i + b) = 1$. We call these points **support points** or **support vectors**. The solution of the SVM problem does not depend on all the data points, it depends only on the support vectors and therefore is **sparse**.

Computing the solution. SVM solvers use the dual problem to compute the solution. Below we derive the dual for the SVM problem. $g(\alpha)$ is computed explicitly by replacing equation (74) in (71). After a simple calculation we obtain

$$g(\alpha) \;=\; \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} y^i y_j {x^i}^T x_j \alpha_i \alpha_j \tag{75}$$

or, in vector/matrix notation

$$g(\alpha) \;=\; 1^T \alpha - \frac{1}{2} \alpha^T G \alpha \tag{76}$$

where $G \;=\; [G_{ij}]_{ij} \;=\; [y^i y_j {x^i}^T x_j]_{ij}$.

A simple SVM problem

Data: 4 vectors in the plane and their labels

$$\begin{aligned} x_1 &= (-2,-2) & y_1 &= +1 \\ x_2 &= (-1,1) & y_2 &= +1 \\ x_3 &= (1,1) & y_3 &= -1 \\ x_4 &= (2,-2) & y_4 &= -1 \end{aligned}$$

The Gramm matrix $G = [x^{i\,T} x_j]_{i,j=1:l}$

$$G = \begin{bmatrix} 8 & 0 & -4 & 0 \\ 0 & 2 & 0 & -4 \\ -4 & 0 & 2 & 0 \\ 0 & -4 & 0 & 8 \end{bmatrix}$$

The dual function to be maximized (subject to $\alpha_i \geq 0$) is

$$\begin{aligned} g(\alpha) &= \sum_i \alpha_i - \frac{1}{2} \sum_i \alpha_i \alpha_j y^i y_j x^{i\,T} x_j \\ &= \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 - 4\alpha_1^2 - \alpha_2^2 - \alpha_3^2 - 4\alpha_4^2 - 4\alpha_1\alpha_3 - 4\alpha_2\alpha_4 \\ &= (2\alpha_1 + \alpha_3) - (2\alpha_1 + \alpha_3)^2 - \alpha_1 \\ &\quad +(\alpha_2 + 2\alpha_4) - (\alpha_2 + 2\alpha_4)^2 - \alpha_4 \end{aligned}$$

The parts depending on α_1, α_3 and α_2, α_4 can be maximized separately.

After some short calculations we obtain:

$$\alpha_1 = 0 \qquad \alpha_4 = 0$$
$$\alpha_2 = \frac{1}{2} \qquad \alpha_3 = \frac{1}{2}$$

Hence, the support vectors are x_2 and x_3. From these, we obtain

$$w = \sum_i \alpha_i y^i x^i = \frac{1}{2}(x_2 - x_3) = (-1, 0)$$
$$b = y_2 - w^T x_2 = 0$$

The results are depicted in the figure below: