ECE595 / STAT598: Machine Learning I
Lecture 20 Support Vector Machine: Duality

Spring 2020

Stanley Chan

School of Electrical and Computer Engineering
Purdue University

Outline

Support Vector Machine

- Lecture 19 SVM 1: The Concept of Max-Margin
- Lecture 20 SVM 2: Dual SVM
- Lecture 21 SVM 3: Kernel SVM

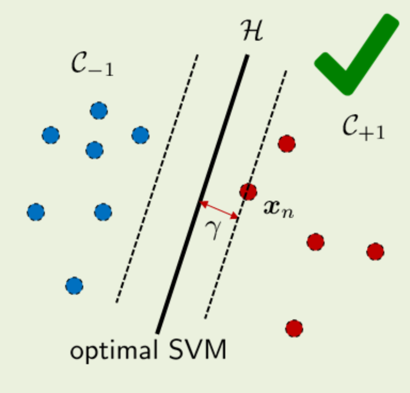
This lecture: Support Vector Machine: Duality

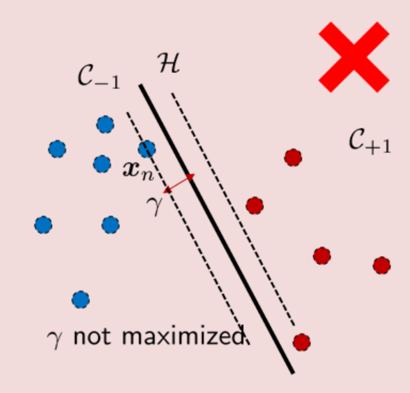
- Lagrange Duality
  - Maximize the dual variable
  - Minimax Problem
  - Toy Example
- Dual SVM
  - Formulation
  - Interpretation

Support Vector Machine

SVM is the solution of this optimization

$$\begin{aligned} \underset{\boldsymbol{w}, w_0}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{subject to} \quad & y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0) \geq 1, \quad j = 1, \ldots, N. \end{aligned}$$





Lagrange Function

- **Goal**: Construct the dual problem of

$$
\begin{aligned}
\underset{\boldsymbol{w}, w_0}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\
\text{subject to} \quad & y_j(\boldsymbol{w}^T\boldsymbol{x}_j + w_0) \geq 1, \quad j = 1, \ldots, N.
\end{aligned}
$$

- **Approach**: Consider the Lagrangian function

$$
\mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) \stackrel{\text{def}}{=} \underbrace{\frac{1}{2}\|\boldsymbol{w}\|_2^2}_{\text{objective}} + \sum_{j=1}^{N} \lambda_j \underbrace{\left[1 - y_j(\boldsymbol{w}^T\boldsymbol{x}_j + w_0)\right]}_{\text{constraint}},
$$

- Solution happens at the saddle point of $\mathcal{L}$:

$$
\nabla_{(\boldsymbol{w}, w_0)} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) = \boldsymbol{0}, \quad \text{and} \quad \nabla_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) = \boldsymbol{0}.
$$

Lagrangian function

- The **Lagrangian function** is

$$\mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) \overset{\text{def}}{=} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \sum_{j=1}^{N} \lambda_j \underbrace{\left[1 - y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0)\right]}_{\leq 0}$$

- **Complementarity Condition** says
  - $\lambda_j > 0$ and $\left[1 - y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0)\right] = 0$
  - $\lambda_j = 0$ and $\left[1 - y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0)\right] < 0$
- So, if we want $\nabla_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) = \boldsymbol{0}$, then must be one of the two cases:

$$\sum_{j=1}^{N} \lambda_j \left[1 - y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0)\right] \rightarrow \max \quad \text{or} \quad \min$$

- No saddle point because linear in λ.
- But $1 - y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0) \leq 0$. So unbounded minimum. So must go with maximum.

Primal Problem

- Let $\boldsymbol{\lambda}^*$ as the maximizer

$$\boldsymbol{\lambda}^* \stackrel{\text{def}}{=} \underset{\boldsymbol{\lambda} \geq \mathbf{0}}{\operatorname{argmax}} \left\{ \sum_{j=1}^{N} \lambda_j \left[1 - y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0) \right] \right\}$$

- Then the **primal problem** is

$$\begin{aligned} & \underset{\boldsymbol{w}, w_0}{\text{minimize}} \quad \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}^*) \\ &= \underset{\boldsymbol{w}, w_0}{\text{minimize}} \left\{ \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \max_{\boldsymbol{\lambda} \geq \mathbf{0}} \left\{ \sum_{j=1}^{N} \lambda_j \left[1 - y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0) \right] \right\} \right\} \\ &= \underset{\boldsymbol{w}, w_0}{\text{minimize}} \left\{ \max_{\boldsymbol{\lambda} \geq 0} \; \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) \right\} \end{aligned}$$

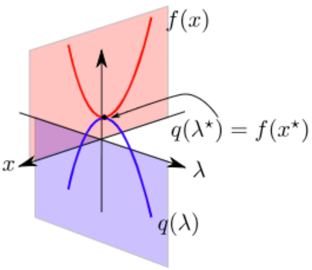
- This is a min-max problem:

$$\min_{\boldsymbol{w}, w_0} \; \max_{\boldsymbol{\lambda} \geq 0} \; \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda})$$

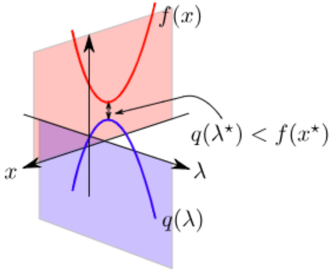
Strong Duality

- Recall that our problem is **quadratic programming** (QP).
- **Strong Duality** holds for QP:

$$\underbrace{\min_{\boldsymbol{w}, w_0} \max_{\boldsymbol{\lambda} \geq 0} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda})}_{\text{primal}} = \underbrace{\max_{\boldsymbol{\lambda} \geq 0} \min_{\boldsymbol{w}, w_0} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda})}_{\text{dual}}$$



strong duality



weak duality

http://www.onmyphd.com/?p=duality.theory

Toy Example

- The SVM problem

$$\begin{aligned} \underset{\boldsymbol{w}, w_0}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\ \text{subject to} \quad & y_j(\boldsymbol{w}^T \boldsymbol{x}_j + w_0) \geq 1, \quad j = 1, \ldots, N. \end{aligned}$$

is in the form of

$$\underset{\boldsymbol{u}}{\text{minimize}} \quad \|\boldsymbol{u}\|^2, \quad \text{subject to} \quad \boldsymbol{a}_j^T \boldsymbol{u} \geq b_j, \ \ j = 1, 2, \ldots, N$$

- **Example**:

$$\underset{u_1, u_2}{\text{minimize}} \quad u_1^2 + u_2^2, \quad \text{subject to} \quad \begin{bmatrix} 1 & 2 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \geq \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}$$

- Can we write down its dual problem?

Toy Example

- Lagrangian function is

$$\mathcal{L}(\boldsymbol{u}, \boldsymbol{\lambda}) \stackrel{\text{def}}{=} u_1^2 + u_2^2 + \lambda_1(2 - u_1 - 2u_2) - \lambda_2 u_1 - \lambda_3 u_3$$

- Minimize over $\boldsymbol{u}$:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial u_1} = 0 \quad &\Rightarrow \quad u_1 = \frac{\lambda_1 + \lambda_2}{2} \\ \frac{\partial \mathcal{L}}{\partial u_2} = 0 \quad &\Rightarrow \quad u_2 = \frac{2\lambda_1 + \lambda_3}{2}. \end{aligned}$$

- Plugging into the Lagrangian function yields

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{maximize}} \quad & \mathcal{L}(\boldsymbol{\lambda}) = -\frac{5}{4}\lambda_1^2 - \frac{1}{4}\lambda_2^2 - \frac{1}{4}\lambda_3^2 - \frac{1}{2}\lambda_1\lambda_2 - \lambda_1\lambda_3 + 2\lambda_1 \\ \text{subject to} \quad & \boldsymbol{\lambda} \geq 0. \end{aligned}$$

- Primal is QP. Dual is also QP.

Have We Gained Anything?

- Here is the dual problem:

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{maximize}} \quad & \mathcal{L}(\boldsymbol{\lambda}) = -\frac{5}{4}\lambda_1^2 - \frac{1}{4}\lambda_2^2 - \frac{1}{4}\lambda_3^2 - \frac{1}{2}\lambda_1\lambda_2 - \lambda_1\lambda_3 + 2\lambda_1 \\ \text{subject to} \quad & \boldsymbol{\lambda} \geq 0. \end{aligned}$$

- These terms are all negative! So we must have $\lambda_2 = \lambda_3 = 0$.
- This gives

$$\underset{\lambda_1 \geq 0}{\text{maximize}} \quad -\frac{5}{4}\lambda_1^2 + 2\lambda_1.$$

  which is maximized at $\lambda_1 = \frac{4}{5}$.
- Plugging into the primal yields

$$u_1 = \frac{\lambda_1 + \lambda_2}{2} = \frac{2}{5}, \quad \text{and} \quad u_2 = \frac{2\lambda_1 + \lambda_3}{2} = \frac{4}{5}.$$

Outline

Support Vector Machine

- Lecture 19 SVM 1: The Concept of Max-Margin
- Lecture 20 SVM 2: Dual SVM
- Lecture 21 SVM 3: Kernel SVM

This lecture: Support Vector Machine: Duality

- Lagrange Duality
  - Maximize the dual variable
  - Minimax Problem
  - Toy Example
- Dual SVM
  - Formulation
  - Interpretation

Dual of SVM

- We want to find the dual problem of

$$
\begin{aligned}
\underset{\boldsymbol{w}, w_0}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 \\
\text{subject to} \quad & y_j(\boldsymbol{w}^T\boldsymbol{x}_j + w_0) \geq 1, \quad j = 1, \ldots, N.
\end{aligned}
$$

- We start with the Lagrangian function

$$
\mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) \overset{\text{def}}{=} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \sum_{j=1}^{N} \lambda_j \left[1 - y_j(\boldsymbol{w}^T\boldsymbol{x}_j + w_0)\right].
$$

- Let us **minimize** over $(\boldsymbol{w}, w_0)$:

$$
\nabla_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) = \boldsymbol{w} - \sum_{j=1}^{N} \lambda_j y_j \boldsymbol{x}_j = \boldsymbol{0}
$$

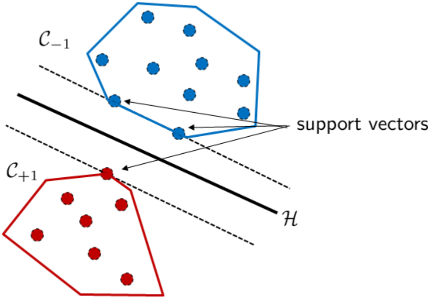
$$
\nabla_{w_0} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) = \sum_{j=1}^{N} \lambda_j y_j = 0.
$$

Interpreting $\nabla_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) = 0$

- The first result suggests that

$$\boldsymbol{w} = \sum_{j=1}^{N} \lambda_j y_j \boldsymbol{x}_j.$$

- This is support vector: λ_j is either $\lambda_j = 0$ or $\lambda_j > 0$.



Interpreting $\nabla_{\boldsymbol{w}} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda}) = 0$

- The complementarity condition states that

$$\lambda_j^* \left[1 - y_j({\boldsymbol{w}^*}^T \boldsymbol{x}_j + w_0^*)\right] = 0, \quad \text{for} \quad j = 1, \ldots, N.$$

- If $1 - y_j({\boldsymbol{w}^*}^T \boldsymbol{x}_j + w_0^*) > 0$, then $\lambda_j^* = 0$
- If $\lambda_j^* > 0$, then $1 - y_j({\boldsymbol{w}^*}^T \boldsymbol{x}_j + w_0^*) = 0$
- So you can define the **support vector set**:

$$\mathcal{V} \overset{\text{def}}{=} \{j \mid \lambda_j^* > 0\}.$$

- So the optimal weight is

$$\boldsymbol{w}^* = \sum_{j \in \mathcal{V}} \lambda_j^* y_j \boldsymbol{x}_j.$$

Back to Duality

The Lagrangian function is

$$
\begin{aligned}
\mathcal{L}(\boldsymbol{w}^*, w_0^*, \boldsymbol{\lambda}) &= \frac{1}{2}\|\boldsymbol{w}^*\|_2^2 + \sum_{j=1}^{N} \lambda_j \left[1 - y_j((\boldsymbol{w}^*)^T \boldsymbol{x}_j + w_0)\right] \\
&= \underbrace{\frac{1}{2}\left\|\sum_{j=1}^{N} \lambda_j y_j \boldsymbol{x}_j\right\|_2^2}_{A} \\
&\quad + \underbrace{\sum_{j=1}^{N} \lambda_j \left[1 - y_j \left(\left(\sum_{i=1}^{n} \lambda_i y_i \boldsymbol{x}_i\right)^T \boldsymbol{x}_j + w_0\right)\right]}_{B}
\end{aligned}
$$

Back to Duality

- We can show that

$$A = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j$$

$$B = \sum_{j=1}^{N}\lambda_j - \sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j - \underbrace{\left(\sum_{j=1}^{N}\lambda_j y_j\right)}_{=0} w_0$$

- and we can show that

$$A + B = \sum_{j=1}^{N}\lambda_j + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j$$

Back to Duality

- Therefore, the dual problem is

$$\begin{aligned} \underset{\boldsymbol{\lambda}\geq 0}{\text{maximize}} \quad & -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_{j=1}^{N}\lambda_j \\ \text{subject to} \quad & \sum_{j=1}^{N}\lambda_j y_j = 0. \end{aligned}$$

- If you prefer matrix-vector:

$$\begin{aligned} \underset{\boldsymbol{\lambda}\geq 0}{\text{maximize}} \quad & -\frac{1}{2}\boldsymbol{\lambda}^T\boldsymbol{Q}\boldsymbol{\lambda} + \boldsymbol{1}^T\boldsymbol{\lambda} \\ \text{subject to} \quad & \boldsymbol{y}^T\boldsymbol{\lambda} = \boldsymbol{0}. \end{aligned}$$

- We can combine the constraints $\boldsymbol{\lambda} \geq 0$ and $\boldsymbol{y}^T\boldsymbol{\lambda} = \boldsymbol{0}$ as

$$\boldsymbol{A}\boldsymbol{\lambda} \geq \boldsymbol{0}.$$

Back to Duality

- $\boldsymbol{y}^T\boldsymbol{\lambda} = 0$ means

$$\boldsymbol{y}^T\boldsymbol{\lambda} \geq 0, \quad \text{and} \quad \boldsymbol{y}^T\boldsymbol{\lambda} \leq 0.$$

- Thus, we can write $\boldsymbol{y}^T\boldsymbol{\lambda} = 0$ as

$$\begin{bmatrix} \boldsymbol{y}^T \\ -\boldsymbol{y}^T \end{bmatrix} \boldsymbol{\lambda} \geq \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

- Therefore, the matrices $\boldsymbol{Q}$ and $\boldsymbol{A}$ are

$$\boldsymbol{Q} = \begin{bmatrix} y_1 y_1 \boldsymbol{x}_1^T \boldsymbol{x}_1 & \dots & y_1 y_N \boldsymbol{x}_1^T \boldsymbol{x}_N \\ y_2 y_1 \boldsymbol{x}_2^T \boldsymbol{x}_1 & \dots & y_2 y_N \boldsymbol{x}_2^T \boldsymbol{x}_N \\ \vdots & \vdots & \vdots \\ y_N y_1 \boldsymbol{x}_N^T \boldsymbol{x}_1 & \dots & y_N y_N \boldsymbol{x}_N^T \boldsymbol{x}_N \end{bmatrix} \quad \text{and} \quad \boldsymbol{A} = \begin{bmatrix} \boldsymbol{y}^T \\ -\boldsymbol{y}^T \\ \boldsymbol{I} \end{bmatrix}$$

So How to Solve the SVM Problem?

- You look at the dual problem

$$\begin{aligned} \underset{\boldsymbol{\lambda}}{\text{maximize}} \quad & -\frac{1}{2}\boldsymbol{\lambda}^T \boldsymbol{Q}\boldsymbol{\lambda} + \mathbf{1}^T\boldsymbol{\lambda} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{\lambda} \geq \mathbf{0}. \end{aligned}$$

- You get the solution $\boldsymbol{\lambda}^*$.
- Then compute $\boldsymbol{w}^*$:

$$\boldsymbol{w}^* = \sum_{j \in \mathcal{V}} \lambda_j^* y_j \boldsymbol{x}_j.$$

- $\mathcal{V}$ is the set of support vectors: $\lambda_j > 0$.

Are We Done Yet?

- Not quite.
- We still need to find out w_0^*.
- Pick any **support vector** $\boldsymbol{x}^+ \in \mathcal{C}_+$ and $\boldsymbol{x}^- \in \mathcal{C}_-$.
- Then we have

$$\boldsymbol{w}^T\boldsymbol{x}^+ + w_0 = +1, \quad \text{and} \quad \boldsymbol{w}^T\boldsymbol{x}^- + w_0 = -1.$$

- Sum them, we have $\boldsymbol{w}^T(\boldsymbol{x}^+ + \boldsymbol{x}^-) + 2w_0 = 0$, which means

$$w_0^* = -\frac{(\boldsymbol{x}^+ + \boldsymbol{x}^-)^T\boldsymbol{w}^*}{2}$$

Summary of Dual SVM

- Primal

$$\underset{\boldsymbol{w}, w_0}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2$$
$$\text{subject to} \quad y_j(\boldsymbol{w}^T\boldsymbol{x}_j + w_0) \geq 1, \quad j = 1, \ldots, N.$$

- Strong Duality

$$\underbrace{\min_{\boldsymbol{w}, w_0} \max_{\boldsymbol{\lambda}\geq 0} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda})}_{\text{primal}} = \underbrace{\max_{\boldsymbol{\lambda}\geq 0} \min_{\boldsymbol{w}, w_0} \mathcal{L}(\boldsymbol{w}, w_0, \boldsymbol{\lambda})}_{\text{dual}}$$

- Dual

$$\underset{\boldsymbol{\lambda}\geq 0}{\text{maximize}} \quad -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_j y_i y_j \boldsymbol{x}_i^T\boldsymbol{x}_j + \sum_{j=1}^{N}\lambda_j$$
$$\text{subject to} \quad \sum_{j=1}^{N}\lambda_j y_j = 0.$$

Summary of Dual SVM

- The weights are computed as

$$\boldsymbol{w}^* = \sum_{j=1}^{N} \lambda_j^* y_j \boldsymbol{x}_j.$$

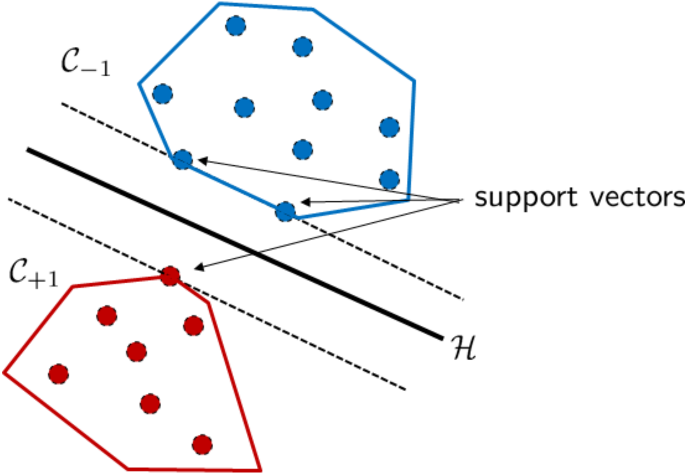
- This is support vector: λ_j is either $\lambda_j = 0$ or $\lambda_j > 0$.
- Pick any **support vector** $\boldsymbol{x}^+ \in \mathcal{C}_+$ and $\boldsymbol{x}^- \in \mathcal{C}_-$.
- Then we have

$$\boldsymbol{w}^T \boldsymbol{x}^+ + w_0 = +1, \quad \text{and} \quad \boldsymbol{w}^T \boldsymbol{x}^- + w_0 = -1.$$

- Sum them, we have $\boldsymbol{w}^T(\boldsymbol{x}^+ + \boldsymbol{x}^-) + 2w_0 = 0$, which means

$$w_0^* = -\frac{(\boldsymbol{x}^+ + \boldsymbol{x}^-)^T \boldsymbol{w}^*}{2}$$

Summary of Dual SVM



Reading List

Support Vector Machine

- Mustafa, *Learning from Data*, e-Chapter
- Duda-Hart-Stork, *Pattern Classification*, Chapter 5.5
- Chris Bishop, *Pattern Recognition*, Chapter 7.1
- UCSD Statistical Learning
  http://www.svcl.ucsd.edu/courses/ece271B-F09/

Appendix

Inequality Constrained Optimization

Inequality constrained optimization:

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathbb{R}^n}{\text{minimize}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & g_i(\boldsymbol{x}) \geq 0, \qquad i = 1, \ldots, m \\ & h_j(\boldsymbol{x}) = 0, \qquad j = 1, \ldots, k. \end{aligned}$$

Requires a function: Lagrangian function

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} f(\boldsymbol{x}) - \sum_{i=1}^{m} \mu_i g_i(\boldsymbol{x}) - \sum_{j=1}^{k} \nu_j h_j(\boldsymbol{x}).$$

$\boldsymbol{\mu} \in \mathbb{R}^m$ and $\boldsymbol{\nu} \in \mathbb{R}^k$ are called the **Lagrange multipliers** or the **dual variables**.

Karush-Kahn-Tucker Conditions

If $(\boldsymbol{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*)$ is the solution to the constrained optimization, then all the following conditions should hold:

(i) $\nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\nu}^*) = \mathbf{0}$.
- **Stationarity.**
- The primal variables should be stationary.

(ii) $g_i(\boldsymbol{x}^*) \geq 0$ and $h_j(\boldsymbol{x}^*) = 0$ for all i and j.
- **Primal Feasibility.**
- Ensures that constraints are satisfied.

(iii) $\mu_i^* \geq 0$ for all i and j.
- **Dual Feasibility.**
- Require $\mu_i^* \geq 0$; but no constraint on ν_i^*.

(iv) $\mu_i^* g_i(\boldsymbol{x}^*) = 0$ for all i and j.
- **Complementary Slackness**
- Either $\mu_i^* = 0$ or $g_i(\boldsymbol{x}^*) = 0$ (or both).

KKT Condition is a first order **necessary** condition.

Example: ℓ_2-minimization with two constraints

Solve the following least squares over positive quadrant problem.

$$\begin{aligned} &\underset{\boldsymbol{x}\in\mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{b}\|^2, \\ &\text{subject to} \quad \boldsymbol{x}^T\mathbf{1} = 1, \quad \text{and} \quad \boldsymbol{x} \geq \mathbf{0}. \end{aligned} \tag{1}$$

```
%MATLAB code: Use CVX to solve min ||x-b|| s.t. sum(x) = 1, x >= 0.
cvx_begin
   variable x(n)
   minimize( norm(x-b, 2) )
   subject to
        sum(x) == 1;
        x      >= 0;
cvx_end
```

Analytic Solution

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \gamma) = \frac{1}{2}\|\boldsymbol{x} - \boldsymbol{y}\|^2 - \boldsymbol{\lambda}^T\boldsymbol{x} - \gamma(1 - \boldsymbol{x}^T\mathbf{1}).$$

Stationarity suggests that:

$$\nabla_{\boldsymbol{x}}\mathcal{L}(\boldsymbol{x}, \boldsymbol{\lambda}, \gamma) = \boldsymbol{x} - \boldsymbol{b} - \boldsymbol{\lambda} + \gamma\mathbf{1} = \mathbf{0}$$

This means

$$\boldsymbol{x} = \boldsymbol{b} + \boldsymbol{\lambda} - \gamma\mathbf{1} \quad \text{or} \quad x_i = b_i + \lambda_i - \gamma.$$

The **complementary slackness** implies $\lambda_i x_i = 0$.

- Case 1: If $\lambda_i = 0$, then
  - $x_i = b_i + \cancelto{0}{\lambda_i} - \gamma = b_i - \gamma$.
  - Since constraint requires $x_i \geq 0$, this means $b_i \geq \gamma$.
- Case 2: If $\lambda_i > 0$, then $x_i = 0$.
  - $\cancelto{0}{x_i} = b_i + \lambda_i - \gamma$.
  - This implies $b_i + \lambda_i = \gamma$.
  - Since $\lambda_i > 0$, this implies $b_i < \gamma$.

These three cases can be re-written as:

- If $b_i > \gamma$, then $x_i = b_i - \gamma$;
- If $b_i = \gamma$, then $x_i = 0$;
- If $b_i < \gamma$, then $x_i = 0$.

Compactly written as

$$\boldsymbol{x} = \max(\boldsymbol{b} - \gamma \mathbf{1}, 0).$$

Primal feasibility implies that

$$\boldsymbol{x}^T \mathbf{1} = 1, \qquad \Leftrightarrow \qquad \sum_{i=1}^{n} x_i = 1.$$

Therefore, γ needs to satisfy the equation

$$\sum_{i=1}^{n} \max(b_i - \gamma, 0) = 1,$$

which can be found by doing a root-finding of

$$g(\gamma) = \sum_{i=1}^{n} \max(b_i - \gamma, 0) - 1.$$

Non-CVX Implementation

```
%MATLAB code: solve min ||x-b|| s.t. sum(x) = 1, x >= 0.
n = 10;
b = randn(n,1);
fun = @(gamma) myfun(gamma,b);
gamma = fzero(fun,0);
x = max(b-gamma,0);
```

where the function `myfun` is defined as

```
function y = myfun(gamma,b)
y = sum(max(b-gamma,0))-1;
```