Labs
Machine Learning Course
Fall 2019

EPFL
School of Computer and Communication Sciences
Martin Jaggi & Rüdiger Urbanke
www.epfl.ch/labs/mlo/machine-learning-cs-433

Problem Set 7, Oct 31, 2019 (Theory Questions Part)

2. Support Vector Machines using Coordinate Descent

1. The dual objective function that we have to optimize is the following :

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{maximize}} \quad & f(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^\top \mathbf{1} - \tfrac{1}{2\lambda}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} \\ \text{subject to} \quad & \boldsymbol{\alpha} \in [0,1]^N \end{aligned}$$

where $\boldsymbol{Q} := \text{diag}(\boldsymbol{y})\boldsymbol{X}\boldsymbol{X}^\top\text{diag}(\boldsymbol{y})$. For computing coordinate update for one coordinate n, consider the following one variable sub-problem:

$$\begin{aligned} \underset{\gamma\in\mathbb{R}}{\text{maximize}} \quad & f(\boldsymbol{\alpha} + \gamma \boldsymbol{e}_n) \\ \text{subject to} \quad & 0 \le \alpha_n + \gamma \le 1 \end{aligned}$$

where $\boldsymbol{e}_n = [0, \cdots, 1, \cdots, 0]^\top$ (all zero vector except at the n^{th} position). Note that $\gamma = 0$ (no need to update α_n) is an optimum iff $\nabla_n^P f(\boldsymbol{\alpha}) = 0$, where ∇_n denotes the n^{th} component of the gradient, and $\nabla_n^P f(\boldsymbol{\alpha})$ is the projected gradient of f, projected onto the box or interval constraint. $\nabla_n^P f(\boldsymbol{\alpha})$ can be computed as

$$\nabla_n^P f(\boldsymbol{\alpha}) = \begin{cases} \nabla_n f(\boldsymbol{\alpha}) & \text{if } 0 < \alpha_n < 1 \\ \min\{0, \nabla_n f(\boldsymbol{\alpha})\} & \text{if } \alpha_n = 0 \\ \max\{0, \nabla_n f(\boldsymbol{\alpha})\} & \text{if } \alpha_n = 1 \end{cases}$$

Note that, $\nabla f(\boldsymbol{\alpha}) = \mathbf{1} - \frac{1}{2\lambda}(\boldsymbol{Q} + \boldsymbol{Q}^\top)\boldsymbol{\alpha} = \mathbf{1} - \frac{1}{\lambda}\boldsymbol{Q}\boldsymbol{\alpha}$ ($\boldsymbol{Q}$ is symmetric) and thus $\nabla_n f(\boldsymbol{\alpha}) = 1 - \frac{1}{\lambda}\boldsymbol{e}_n^\top \boldsymbol{Q}\boldsymbol{\alpha}$. Simplifying $f(\boldsymbol{\alpha} + \gamma\boldsymbol{e}_n)$, we get

$$\begin{aligned} f(\boldsymbol{\alpha} + \gamma\boldsymbol{e}_n) &= (\boldsymbol{\alpha} + \gamma\boldsymbol{e}_n)^\top \mathbf{1} - \tfrac{1}{2\lambda}(\boldsymbol{\alpha} + \gamma\boldsymbol{e}_n)^\top \boldsymbol{Q}(\boldsymbol{\alpha} + \gamma\boldsymbol{e}_n) \\ &= \boldsymbol{\alpha}^\top\mathbf{1} - \tfrac{1}{2\lambda}\boldsymbol{\alpha}^\top\boldsymbol{Q}\boldsymbol{\alpha} + \gamma - \tfrac{1}{2\lambda}(\gamma^2\boldsymbol{e}_n^\top\boldsymbol{Q}\boldsymbol{e}_n + \gamma\boldsymbol{\alpha}^\top\boldsymbol{Q}\boldsymbol{e}_n + \gamma\boldsymbol{e}_n^\top\boldsymbol{Q}\boldsymbol{\alpha}) \\ &= f(\boldsymbol{\alpha}) - \tfrac{\gamma^2}{2\lambda}\boldsymbol{Q}_{nn} + \gamma(1 - \tfrac{1}{\lambda}\boldsymbol{\alpha}^\top\boldsymbol{Q}\boldsymbol{e}_n) \end{aligned}$$

Differentiating with respect to γ and equating to 0, we get :

$$\begin{aligned} -\tfrac{\gamma^\star}{\lambda}\boldsymbol{Q}_{nn} + (1 - \tfrac{1}{\lambda}\boldsymbol{\alpha}^\top\boldsymbol{Q}\boldsymbol{e}_n) &= 0 \\ \gamma^\star &= \frac{\lambda}{\boldsymbol{Q}_{nn}}(1 - \tfrac{1}{\lambda}\boldsymbol{\alpha}^\top\boldsymbol{Q}\boldsymbol{e}_n) \end{aligned}$$

Note that $\boldsymbol{Q}_{nn} = \boldsymbol{x}_n^\top\boldsymbol{x}_n y_n^2 = \boldsymbol{x}_n^\top\boldsymbol{x}_n$ and $\boldsymbol{\alpha}^\top\boldsymbol{Q}\boldsymbol{e}_n = \sum_{i=1}^N \alpha_i\boldsymbol{Q}_{i,n} = \sum_{i=1}^N \alpha_i y_i \boldsymbol{x}_i^\top\boldsymbol{x}_n y_n$. Using $\boldsymbol{w}(\boldsymbol{\alpha}) = \frac{1}{\lambda}\sum_{i=1}^N \alpha_i y_i \boldsymbol{x}_i$, we get $\boldsymbol{\alpha}^\top\boldsymbol{Q}\boldsymbol{e}_n = \lambda y_n \boldsymbol{w}^\top\boldsymbol{x}_n$ and thus

$$\gamma^\star = \frac{\lambda}{\boldsymbol{x}_n^\top\boldsymbol{x}_n}(1 - y_n\boldsymbol{w}^\top\boldsymbol{x}_n)$$

We conclude

$$\begin{aligned} \alpha_n^{\text{new}} &= \alpha_n^{\text{old}} + \gamma^\star \\ &= \alpha_n^{\text{old}} + \frac{\lambda}{\boldsymbol{x}_n^\top\boldsymbol{x}_n}(1 - y_n\boldsymbol{w}^\top\boldsymbol{x}_n) \end{aligned}$$

Since $\boldsymbol{\alpha} \in [0,1]^N$, we project α_n as

$$\alpha_n^{\text{new}} := \min\left\{ \max\left\{ \alpha_n^{\text{old}} + \frac{\lambda}{\boldsymbol{x}_n^\top \boldsymbol{x}_n}(1 - y_n \boldsymbol{w}^\top \boldsymbol{x}_n), 0 \right\}, 1 \right\}$$

3. Kernels

1. Note that every term in the resulting polynomial is a product of kernels and that all these terms have positive coefficients. Hence the result follows by the two statements proved in class: namely that the positive sum of valid kernels is a valid kernel and that the product of valid kernels is a valid kernel.

2. We have $\exp(x) = \sum_{i \geq 0} \frac{x^i}{i!}$. We can hence apply the previous result concerning polynomials with positive coefficients and apply the limit.