Machine Learning (708.064)
Assignment 3 - Support Vector Machines and duality

May 29, 2019

Lecturers: G. Bellec and A. Subramoney
Tutors: L. Lindner (lydia.lindner@student.tugraz.at), D. Narnhofer (dominik.narnhofer@icg.tugraz.at)
Submission: Each group should submit a report and their code to the teach center. The students should use python 3.6 or higher and up to date versions of numpy and matplotlib libraries. It is forbidden to use any other libraries such as scipy or scikit-learn.
Deadline: **Thursday the 13^{th} of June**, 2019 at 23:55h.

1 Support Vector Regression: the primal problem (10 points)

For background information about SVM see Bishop 2006, for background information on convex optimization or accelerated gradient descent variants, see Chambolle and Pock 2016. In this first part of the assignment, we consider the linear regression formalized as follows

$$\underset{\boldsymbol{w}}{\text{minimize}} \quad C\sum_{n=1}^{N} E_\epsilon\left(y^{(n)} - t^{(n)}\right) + \frac{1}{2}||w||^2 \,, \tag{1}$$

where C and ϵ are positive constants, E_ϵ is defined by $E_\epsilon(e) = \max(0, |e| - \epsilon)$, $t^{(n)} \in \mathbb{R}$ is the target of the data point n, $y^{(n)} = \boldsymbol{w}^T\phi(\boldsymbol{x}^{(n)})$ is the linear combination of the feature vector $\boldsymbol{x}^{(n)}$ with the weights w. In this assignment we consider the swiss-roll dataset with feature vectors $\boldsymbol{x}^{(n)} \in \mathbb{R}^2$ and target $t^{(n)} \in \mathbb{R}$. The dataset can be generated from random variables $\psi^{(n)}$, $\nu_1^{(n)}$ and $\nu_2^{(n)}$ where $\psi^{(n)}$ is sampled uniformly in $[\pi, 5\pi]$ whereas $\nu_1^{(n)}$ and $\nu_2^{(n)}$ are sampled from Gaussian distributions of mean 0 and standard deviation $\sigma = 0.7$. The targets and the feature vectors and then given by

$$\begin{aligned} x_1^{(n)} &= \left(\psi^{(n)} + \nu_1^{(n)}\right)\cos\left(\psi^{(n)} + \nu_2^{(n)}\right) \\ x_2^{(n)} &= \left(\psi^{(n)} + \nu_1^{(n)}\right)\sin\left(\psi^{(n)} + \nu_2^{(n)}\right) \\ t^{(n)} &= \psi^{(n)} \,. \end{aligned}$$

To solve this problem, we use a non-linear expansion ϕ of the form $\phi(\boldsymbol{x}) = \left[1, x_1, x_2, \sqrt{x_1^2 + x_2^2}\right]^T$.

1. Sample a training set of 100 points from the swiss-roll distribution and solve the problem in equation (1) with gradient descent (GD) with $C = 1$ and $\epsilon = 0.1$. Show a plot of the dataset with x_1 and x_2 on the x and y-axis respectively and encode the target values t with colours. To visualize the precision of your solution, display a scatter plot with $t^{(n)}$ on the x-axis and $y^{(n)}$ on the y-axis for freshly generated samples that were not used during training. For a good solution the cloud of points should coincide with the line $y = x$.

2. Solve the same problem with the Accelerated Gradient Descent (AGD) algorithm described in pseudo code in algorithm 1. Show in your report the cost over iterations for GD and AGD, the y-axis should display the cost on a log scale.

Initialize $t_0 = 0$, $w_0 = w_{\text{-}1}$
for $k \geq 0$ **do**
  $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$
  $\boldsymbol{v}_k = \boldsymbol{w}_k + \frac{t_k - 1}{t_{k+1}} (\boldsymbol{w}_k - \boldsymbol{w}_{k-1})$
  $\boldsymbol{w}_{k+1} = \boldsymbol{v}_k - \tau \nabla f(\boldsymbol{v}_k)$
end

Algorithm 1: Pseudo code of the Accelerated Gradient Descent (AGD) algorithm with fixed step τ. This algorithm minimizes the cost function f.

2 Support Vector Regression: the dual problem (10 points)

The dual formulation of the same problem can be written as

$$\begin{aligned} \underset{\boldsymbol{a},\boldsymbol{b}}{\text{maximize}} \quad & -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}(a_n - b_n)K(\boldsymbol{x}^{(n)}, \boldsymbol{x}^{(m)})(a_m - b_m) - \epsilon\sum_{n=1}^{N}(a_n + b_n) + \sum_{n=1}^{N} t_n(a_n - b_n) \\ \text{subject to} \quad & 0 \leq a_n \leq C \\ & 0 \leq b_n \leq C, \text{ for } n, \in \{1, \ldots, N\} \end{aligned} \tag{2}$$

where the parameters a_n and b_n are sometimes called Lagrange multipliers. The kernel $K(\boldsymbol{x}^{(n)}, \boldsymbol{x}^{(m)})$ is an $N \times N$ matrix where each elements is given by $K(\boldsymbol{x}^{(n)}, \boldsymbol{x}^{(m)}) = \phi(\boldsymbol{x}^{(n)})^T \phi(\boldsymbol{x}^{(m)})$.

1. Solve the dual problem for the same swiss roll dataset and the same hyper-parameters as before. To solve the constrained optimization problem in (2) we use the projected gradient descent algorithm (see algorithm (2)). Given that the predictions are given by $y(\boldsymbol{x}) = \sum_{n=1}^{N}(a_n - b_n)K(\boldsymbol{x}_n, \boldsymbol{x})$, show a scatter plot with $y^{(n)}$ on the y-axis and $t^{(n)}$ on the x-axis to visualize the precision of the regression on a freshly generated samples that were not used during training.

for $k \geq 0$ **do**
  $\boldsymbol{v}_k = \boldsymbol{w}_k - \tau \nabla f(\boldsymbol{w}_k)$
  $\boldsymbol{w}_{k+1} = \Pi(\boldsymbol{v}_k)$
end

Algorithm 2: Pseudo code of the projected gradient descent (PG) algorithm with fixed step τ, and projection operator Π. This algorithm minimizes the cost function f under the constraint $\mathcal{C}$ if the projection operator Π projects the current solution $\boldsymbol{v}_k$ onto the constraint $\mathcal{C}$.

2. Solve the same problem with the FISTA algorithm described in pseudo code in algorithm 3. Show in your report the cost over iterations for PG and FISTA, the y-axis should display the cost on a log scale.

3 Investigations (5 points)

1. In the dual problem, replace the kernel with a radial basis function (RBF) kernel $K(\boldsymbol{x}, \boldsymbol{x}_j) = \exp\left(-\frac{||\boldsymbol{x}-\boldsymbol{x}_j||^2}{2\sigma_K^2}\right)$. By changing the number of points N, the parameter C and the parameter σ_K,

Initialize $t_0 = 0$, $w_0 = w_{\text{-}1}$
for $k \geq 0$ **do**
  $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$
  $\boldsymbol{v}_k = \boldsymbol{w}_k + \frac{t_k - 1}{t_{k+1}} (\boldsymbol{w}_k - \boldsymbol{w}_{k-1})$
  $\boldsymbol{w}_{k+1} = \Pi(\boldsymbol{v}_k - \tau \nabla f(\boldsymbol{v}_k))$
end

Algorithm 3: Pseudo code of the FISTA algorithm with fixed step τ, and projection operator Π. This algorithm minimizes the cost function f under the constraint $\mathcal{C}$ if the projection operator Π projects the current solution $\boldsymbol{v}_k$ onto the constraint $\mathcal{C}$.

discuss in which condition this kernel works well on this problem. Your answer should contain maximum 5 lines and the highest grades requires to be supported with numerical results and/or one Figure of your choice.

2. What is the difference between the solution of the primal formulation of SVM regression in comparison to a standard regularized mean square error? Your answer should contain maximum 5 lines and the highest grades requires to be supported with numerical results and/or one Figure of your choice.

References

Chambolle, Antonin, and Thomas Pock. "An introduction to continuous optimization for imaging." Acta Numerica 25 (2016): 161-319.

Bishop, Christopher M. Pattern recognition and machine learning. springer, 2006. Chapter 7, section 7.2.1 on SVMs regression, page 339.