Convex Optimization for Machine Learning and Computer Vision

Lecture: Dr. Tao Wu
Exercises: Emanuel Laude, Zhenzhang Ye
Summer Semester 2018

Computer Vision Group
Institut für Informatik
Technische Universität München

Weekly Exercises 5

Room: 02.09.023
Wednesday, 30.05.2018, 12:15-14:00
Submission deadline: Monday, 28.05.2018, 16:15, Room 02.09.023

Convex conjugate and prox (Due: 28.05) (8+4 Points)

Exercise 1 (4 points)**.** Consider following problems of convex conjugate:

- Let $f : \mathbb{R}^n \to \mathbb{R}$ be convex. Show that the convex conjugate of the perspective function $g : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R} \cup \{\infty\}$

$$g(x,t) = \begin{cases} tf(x/t), & \text{if } t > 0 \\ \infty, & \text{otherwise} \end{cases}$$

  is given by

$$g^*(y,s) = \begin{cases} 0, & \text{if } f^*(y) \leq s \\ \infty, & \text{otherwise} \end{cases}$$

- Show that the biconjugate of the persepective function g is given by

$$g^* * (x,t) = \begin{cases} tf(x/t), & \text{if } t > 0 \\ \sigma_{\text{dom}(f^*)}(x), & \text{if } t = 0 \\ \infty, & \text{if } t < 0 \end{cases}$$

  where $\sigma_{\text{dom}(f^*)}(x) = \sup_{y \in \text{dom}(f^*)} \langle x, y \rangle$ is the *support function* of $\text{dom}(f^*)$.

Exercise 2 (4 points)**.** Compute the proximity operator of the $1,2$-norm, i.e.

$$\text{prox}_{\tau ||X||_{1,2}},$$

where $X \in \mathbb{R}^{m \times n}$ is a matrix .

Exercise 3 (4 points)**.** Prove that the proximal operator of the nuclear norm is the proximal operator of the ℓ_1-norm applied to the singular values of the input argument. Formally, let $Y \in \mathbb{R}^{n \times n}$ and let $Y = U\Sigma V^\top$ be the singular value decomposition of Y. Prove that

$$\text{prox}_{\tau \|\cdot\|_{\text{nuc}}}(Y) = U\text{diag}(\{(\sigma_i - \tau)_+\})V^\top,$$

where $\text{diag}(\{\sigma_i - \tau\}_+) := \text{diag}(\{\max\{0, \sigma_i - \tau\}\}) = \text{prox}_{\tau\|\cdot\|_1}(\{\sigma_i\})$ is the shrinkage (or soft thresholding) operator applied to the singular values σ_i of Y.

Multinomial Logistic Regression(Due:28.05) (16 Points)

Exercise 4 (16 Points)**.** In this exercise you are asked to train a linear model for a multiclass classification task with Logistic regression. The idea is as follows: You are given a set of training samples $\mathcal{I} = \{1, \dots, N\}$ that are represented by their feature vectors $x_i \in \mathbb{R}^d$, for $i \in \mathcal{I}$. Each training sample i is associated with a class label $y_i \in \{1, \dots, C\}$. The aim is to estimate a linear classifier parameterized by $W^* \in \mathbb{R}^{d \times C}, b^* \in \mathbb{R}^C$ so that $y_i = \text{argmax}_{1 \le j \le C}\, x_i^\top W_j^* + b_j^*$ for most training samples i. Once you have obtained this "optimal" classifier the hope is, that you are able to classify new unseen and unlabeled samples $x \in \mathbb{R}^d$. In machine learning this is called generalization. For this task you may query your trained model via the classifier rule

$$y = \text{argmax}_{1 \le j \le C}\, x^\top W_j^* + b_j^* \tag{1}$$

and y probably is the true class label of x if your model generalizes well.

In order to estimate the model we solve an optimization problem of the form

$$\min_{W \in \mathbb{R}^{d \times C}, b \in \mathbb{R}^C} \frac{1}{N} \sum_{i=1}^{N} \ell(W, b, x_i, y_i) + \frac{\lambda_1}{2} \|W\|_2^2 + \frac{\lambda_2}{2} \|b\|_2^2, \tag{2}$$

where

$$\ell(W, b, x_i, y_i) = -\log\left(\frac{\exp(\langle W_{y_i}, x_i \rangle + b_{y_i})}{\sum_{j=1}^{C} \exp(\langle W_j, x_i \rangle + b_j)} \right) \tag{3}$$

is called the softmax loss. Note that the above problem is smooth and strongly convex and can be solved with gradient descent. In practice however, it may happen, that some features (i.e. components of the vector x_i) do not contain any information about the true class labels, i.e. components that are just noise. In order to filter out the useless features we modify the norm on W. So we have

$$\min_{W \in \mathbb{R}^{d \times C}, b \in \mathbb{R}^C} \frac{1}{N} \sum_{i=1}^{N} \ell(W, b, x_i, y_i) + \frac{\lambda_1}{2} \|W\|_{1,2} + \frac{\lambda_2}{2} \|b\|_2^2 \tag{4}$$

You are asked to do the following:

- Download the toy data template from the homepage
- Implement a proximal gradient descent algorithm to optimize above objective function (4) (Avoid for-loops)
- Make sure that your objective monotonically decreases. Plot the objective values. Stop your code if the difference of two successive iterates is less than 10^{-12}.

- In order to ensure that your derivative is computed correctly you may first optimize the fully differentiable model (2) with MATLABs *fminunc* with the options $'GradObj', 'On'$ and $'DerivativeCheck', 'On'$.

- Iteratively compute the test error in percent, i.e. how many test samples are not classified correctly via the rule (1).

- Play around with different parameter settings for λ_1, λ_2. Can you identify the useless features? Explain why the model generalizes better to unseen test data if you use $1,2$-norm on W^* (answer by comment at the end of your code).

- You may apply your code to the MNIST dataset `http://yann.lecun.com/exdb/mnist/` and see that your are now able to classify handwritten digits.