Lehrstuhl Computergrafik
Prof. Dr.-Ing. Hendrik P.A. Lensch
Prof. Dr.-Ing. Andreas Geiger

Eberhard Karls Universität Tübingen

Machine Learning in Graphics & Vision

Exercise 2

Release date: Wed, 6. May 2020 - **Deadline for Homework: Wed, 20. May 2020 - 21:00**

Excercises

Please **do not** use jupyter-notebooks for these exercises.
Each submission should consists of a **report (pdf)** and the **code (py)** files.
Answer **all questions**, write **all derivations** and put **all figures** in your report!

2.1 Logistic Regression (2+3+6+2+3+2+2 Points)

In this exercise, we apply logistic regression for binary classification. Considering a binary classification problem with two classes c_1 and c_2, our goal is to learn a predictive model that predicts the class posterior $p(c_k|\mathbf{x})$ given an observation $\mathbf{x} \in \mathbb{R}^D$ for $k = 1, 2$. For example, $\{c_1, c_2\}$ can be {"Face", "Not Face"} for an image $\mathbf{x}$.
Logistic regression models the class posterior $p(c_1|\mathbf{x})$ by applying the sigmoid function on a linear function of $\mathbf{x}$

$$p(c_1|\mathbf{x}) = \frac{1}{1 + e^{-\mathbf{w}^T\mathbf{x}}}, \tag{1}$$

with weights $\mathbf{w} \in \mathbb{R}^D$ affecting the decision. Note that $p(c_2|\mathbf{x})$ is given by $1 - p(c_1|\mathbf{x})$. For simplicity, let $f_\mathbf{w}(\mathbf{x}) = p(c_1|\mathbf{x})$ denote a function of $\mathbf{x}$ parametrized by $\mathbf{w}$.
In Machine Learning we are interested in learning the weights $\mathbf{w}$ from a representative training dataset $\mathcal{X} = \{(\mathbf{x}_n, t_n)\}_{n=1}^N$, where $t_n \in \{0, 1\}$ and $t_n = 1$ denotes that $\mathbf{x}_n$ belongs to class c_1. Optimizing $\mathbf{w}$ can be done by minimizing the negative log-likelihood (cross-entropy loss function)

$$L(\mathbf{x}, t, \mathbf{w}) = \frac{1}{N} \sum_{n=1}^{N} \left[-t_n \log(f_\mathbf{w}(\mathbf{x}_n)) - (1 - t_n)\log(1 - f_\mathbf{w}(\mathbf{x}_n))\right]. \tag{2}$$

Let's consider the Fashion-MNIST dataset introduced in the last exercise. In this exercise we consider only the two classes "Pullover" and "Coat" of Fashion-MNIST and would like to learn a logistic regression model that predicts the probability of an image to be assign to the class "Pullover". For simplicity, we flatten all pixels of the images to a single column and consider these as feature vector $\mathbf{x}_n$ instead of using more complex features like SIFT.

a) Implement the function $f_\mathbf{w}(\mathbf{x})$ that computes the probability of class "Pullover" for features $\mathbf{x}$. Insert your code in function "predict_proba". What is the classification accuracy of the initialized model on the test dataset?

b) Implement the loss function $L(\mathbf{x}, t, \mathbf{w})$ to optimize the weights $\mathbf{w}$ of the logistic regression model. Insert your code in function "compute_loss". What is the loss of the initialized model?
Note, you need to remove the first "quit()" in the main program before running the code.

c) A very popular method for optimization is gradient descent which minimizes the loss function by following the gradients. Derive the gradients of loss function $L(\mathbf{x}, t, \mathbf{w})$ with respect to $\mathbf{w}$ to use gradient descent. Write the full derivation in your report and try to simplify the derivate as much as possible!

Implement the derived gradients of the loss in the code. Insert your code in function "update_weights". What is the loss and accuracy of the model after 1000 iterations?
Note, you need to remove the second "quit()" in the main program before running the code.

d) Plot the loss function L for $f_{\mathbf{w}} \in [0, 1]$ considering the two cases $t = 0$ and $t = 1$ and describe both plots with respect to the minimization problem.

e) Plot the learning curve (training losses and test accuracy for each iteration) for different learning rates (1e-4,1e-3,1e-2,1e-1,1e-0) and describe the differences. What happens with the smallest and highest learning rate?

f) Set the learning rate as 1e-2 to optimize the weights $\mathbf{w}$ for 100 iterations. Plot the optimized weights $\mathbf{w}$ as a 28×28 image. Draw 10 samples of test images $\mathbf{x}$ (5 "Pullover" and 5 "Coat") and plot the corresponding $\mathbf{w} \odot \mathbf{x}$ (also reshape to 28×28) on the same image, where $\odot$ denotes element-wise product. Based on the plots name at least one major difference between "Pullover" and "Coat" that the logistic regression model learns to distinguish them.

g) Name at least two advantages of a learning approach over nearest neighbors search as discussed in the previous exercise?