Machine Learning (708.064)
Assignment 2 - Dictionary Learning and Logistic Regression

April 12, 2019

Lecturers: Anand Subramoney and Guillaume Bellec

Tutors: Lydia Lindner (lydia.lindner@student.tugraz.at)
Dominik Narnhofer (narnhofer@student.tugraz.at)

Submission: Each group should submit on the Teach Center:

1. a report in pdf (hand-written or tex file) **(please do not put the pdf into a zip file)**,

2. the code producing the figures used in the report.

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, except for plotting.

Deadline: May 17th, 2019 at 23:55h.

In this assignment, you will learn a sparse dictionary over an image dataset using the 'Lasso' approach. Then, you will use the learnt dictionary and the image-specific coefficients to classify the images in the dataset using logistic regression.

In all cases, you are expected to do the entire implementation of the algorithms themselves using only Python and NumPy, including the gradient calculations. You are free to use any libraries for creating the plots.

The dataset you will use for this assignment is the "Fashion-MNIST" dataset available at `https://github.com/zalandoresearch/fashion-mnist`. The url has links to download the dataset, as well as documentation on how to load it into a NumPy array. This dataset consists of a training set of 60,000 examples and a test set of 10,000 examples. Use only the training set for all questions unless specified otherwise. Each example is a 28×28 greyscale image, associated with one of 10 classes of apparel. Since training can take too long with the full dataset, reduce the size of the dataset by choosing just the first 5 classes (class labels 0 to 4), and all the corresponding images for these 5 classes as your dataset. Before learning the dictionary, scale and center the images – divide each pixel by the max possible value of the pixel; then subtract, from each pixel, the mean value calculated for each image.

1 Lasso and dictionary learning (15 points)

Many real world signals can be modelled via sparse representation in a suitable basis or 'dictionary'.

Given input vectors $\mathbf{x}_j \in \mathbb{R}^L, j = 1 \dots N$ which form the columns of the design matrix $X \in \mathbb{R}^{L \times N}$, a sparse representation in the dictionary $D \in \mathbb{R}^{L \times M}$ can be found by solving the following optimization problem:

$$\min_{W,D} \lambda \left(\sum_{i=2}^{M} ||W_i||_1 \right) + \frac{1}{2} ||D\, W - X||_F^2, \tag{1}$$

with the constraints that, the j-th column of the dictionary D satisfies

$$||D_j||_2 \; \leq \; 1; \qquad j = 1 \dots M \tag{2}$$

$$\mathbf{1}^T D_j \; = \; 0; \qquad j = 2 \dots M \tag{3}$$

where M is the number of dictionary atoms, N is the number of images in the dataset, and L is the number of pixels in each image, W_i is the i-th row vector of the unknown coefficient matrix $W \in \mathbb{R}^{M \times N}$, $\mathbf{1}$ denotes the 1-vector of the appropriate dimension (vector of all ones), $||.||_p$ denotes the p-th vector norm, and $||A||_F^2 = \operatorname{tr}(A^T A)$ denotes the Frobenius norm of the matrix A, and $\operatorname{tr}(.)$ denotes the trace of a matrix. The dictionary contains one unconstrained column $D_{j=1}$ that captures the noise in the observed signal, with the corresponding row vector $W_{i=1}$. Note that some constraints are not applied to these vectors when learning W and D using 'PALM'.

This formulation is called the *least absolute shrinkage and selection operator* (*Lasso*) problem. The *Lasso* approach can be interpreted as a model that tries to reconstruct the given image $\mathbf{x}_j$ using only a small number (M) of basis atoms learnt in the dictionary D.

We define the reconstruction loss as:

$$\mathcal{L}(D, W) = \frac{1}{2} ||D\, W - X||_F^2, \tag{4}$$

Your task is to solve the above Lasso problem for your chosen subset of the Fashion-MNIST dataset to find the dictionary D and the coefficient vectors W. The column vectors W_j of W are the coefficient vectors associated with each image. Use the *Proximal Alternating Linearization Method (PALM)* algorithm described in Algorithm 1 to fit these values. Use 32 dictionary atoms ($M = 32$). Use a value of λ that gives you the lowest reconstruction loss.

In your report:

(a) Derive and show the analytical form of the gradients $\nabla_W \mathcal{L}$ and $\nabla_D \mathcal{L}$ used in the PALM algorithm.

(b) Once you've solved for the dictionary D, plot the columns of D as 28×28 images.

(c) Choose a few random images from the your dataset and calculate the reconstructions of these images using the sparse dictionary as $D\, W_j$. Plot the reconstructed images alongside the original images.

(d) Discuss your observations from the plots in (b) and (c).

(e) What effect does using low (10^{-4}) or high (10^{-1}) values of λ have on the dictionary that is learnt and classification performance in the next question? Why?

2 Logistic Regression (10 points)

Train a logistic regression classifier to classify the images, but instead of using the raw images as the input features, use the learnt coefficient vectors W_j for each image, where each W_j is the j-th column vector corresponding to each image. The labels for classification would be the true labels from the dataset.

Use a one-vs-all scheme, where you have 5 different classifiers, each of which output whether the image belongs to its respective class or not. In the one-vs-all scheme, the overall prediction is the label corresponding to the classifier which has the maximum output probability.

Use standard stochastic gradient descent on the binary cross-entropy loss function for each of these classifiers to find the parameters of the logistic regression model.

Recall that the class prediction of the logistic regression model is given by $p(C_1 \,|\, \tilde{W}_j) = \sigma(\tilde{W}_j^T \, \theta)$, where $\tilde{W}_j \in \mathbb{R}^{M+1}$ is the vector W_j augmented with the constant 1 for the bias term of the parameter vector $\theta \in \mathbb{R}^{M+1}$, $\sigma(x) = \frac{1}{1+e^{-x}}$ is the logistic function, $p(C_1 \,|\, \tilde{W}_j)$ denotes the probability that the given feature vector $\tilde{W}_j$ (and by extension the image $\mathbf{x}_j$) belongs to class 1. The equivalent probability for class 2 is $Pr(C_2 \,|\, \tilde{W}_j) = 1 - Pr(C_1 \,|\, \tilde{W}_j)$.

The cross entropy loss is:

$$E = -\sum_{j=1}^{N} y_j \, \log p(C_1 \,|\, \tilde{W}_j) \; + \; (1 - y_j) \log p(C_2 \,|\, \tilde{W}_j) \tag{5}$$

where $y_j \in \{0, 1\}$ are the target classes.

In your report:

(a) Derive and show the analytical form of the gradients used for logistic regression.

(b) Report the classification accuracy and the cross-entropy loss on your chosen dataset.

(c) Compare the performance of logistic regression with and without quadratic regularization (i.e. with an additional $\gamma \, ||\theta||_2^2$ in the loss function E). Which one performs better? Why?

(d) Compare the performance of this classifier with that of a logistic regression classifier trained on scaled and centered raw image pixels.

(e) Plot and interpret the images as points in a 2-D plot colored depending on which classes they belong to. Since the input features are multi-dimensional, use the first two principal components of each input feature, calculated using Principal Component Analysis (PCA). You can use the function `PCA_plot` in the provided code to perform PCA – the docstring in the function explains what arguments to pass in.

3 Bonus Question: Calculating test error (5* points)

So far, you have been using only the images from the training set of "Fashion-MNIST" dataset to learn both the dictionary and the features to use for classification. In this task, learn the coefficient vectors W_j^{test} for each image in the test set, but using the dictionary D learnt in Task 1. This can be done by using just the "Update coefficients W" part of the 'PALM' algorithm, while keeping D fixed without updating it. Use these learnt coefficient vectors W_j^{test} as feature inputs for classification using logistic regression as in Task 2.

In your report:

(a) Report the test classification performance (accuracy and cross-entropy loss) achieved using these features on the images in the test set.

(b) Compare your results with the reported performance of various models at this url. Discuss how this model compare to other models.

(c) Choose a few random images from the test set and calculate the reconstructions of these images using the sparse dictionary as $D \, W_j^{\text{test}}$. Plot the reconstructed images alongside the original images.

PALM algorithm

Algorithm 1 'PALM' algorithm (Bolte et al. 2014)

```
Initialize $W$ and $D$ with random values.
while not converged do
    For iteration $k$
    Update dictionary D
        $\tau_D \leftarrow \sigma_{\max}(W^{(k)})^{-2}$    ▷ Adapt step size $\tau_D$
        $D^{(k+1,1)} \leftarrow D^{(k)} - \tau_D \nabla_D \mathcal{L}(W^{(k)}, D^{(k)})$    ▷ Gradient Descent on $D$
        $D_j^{(k+1,2)} \leftarrow D_j^{(k+1,1)} - \frac{\mathbf{1}^T D_j^{(k+1,1)}}{L}$ for $j = 2 \dots M$    ▷ Apply constraint from Eq. 3
        $D_j^{(k+1)} \leftarrow \frac{D_j^{(k+1,2)}}{\max(1, ||D_j^{(k+1,2)}||_2)}$ for $j = 1 \dots M$    ▷ Apply constraint from Eq. 2
    Update coefficients W
        $\tau_W \leftarrow \sigma_{\max}(D^{(k+1)})^{-2}$    ▷ Adapt step size $\tau_W$
        $W^{(k+1,1)} \leftarrow W^{(k)} - \tau_W \nabla_W \mathcal{L}(W^{(k)}, D^{(k+1)})$    ▷ Gradient Descent on $W$
        $W_i^{(k+1)} \leftarrow \max(\mathbf{0}, |W_i^{(k+1,1)}| - \lambda) \, . \, \mathrm{sign}(W_i^{(k+1,1)})$ for $i = 2 \dots M$    ▷ Shrinkage operator
end while
```

where $\nabla_D \mathcal{L}$ and $\nabla_W \mathcal{L}$ are the gradients of $\mathcal{L}$ with respect to D and W respectively, W_i is the i-th row of W, D_j is the j-th column of D, $\mathbf{0}$ is the zero vector, max is the component-wise maximum, $|.|$ denotes component-wise absolute value, '.' is the component-wise product, $||.||_p$ denotes the p-th vector norm, $\sigma_{\max}(.)$ denotes maximum singular value of the matrix, for each component x_i in vector x, $\mathrm{sign}(x)_i = x_i/|x_i|$, $D^{(k,t)}$ denotes the t-th sub-step of the k-th iteration.