Business Analytics
Exercise Sheet 9

Martin Wistuba (wistuba@ismll.de)
Information Systems and Machine Learning Lab (ISMLL)
Universität Hildesheim

1 July 2014
Submission until 8 July 2014 23:59

Exercise 23: PCA - Stochastic Gradient Descent (5 points)

Principal Component Analysis (PCA) is a dimensionality reduction technique which aims at projecting a dataset $X \in \mathbb{R}^{N \times M}$ via latent principal components $V \in \mathbb{R}^{K \times M}$ for $K << M$. The procedure aims at learning both the latent components and a linear combinations of the components via weights $Z \in \mathbb{R}^{N \times K}$, such that the original data is approximated via the following loss L:

$$\operatorname*{argmin}_{Z,V} L = ||X - Z \cdot V||^2 = \sum_{i=1}^{N} \sum_{j=1}^{M} \left(X_{i,j} - \sum_{k=1}^{K} Z_{i,k} V_{k,j} \right)^2 \tag{1}$$

(a) Explain the relation between the PCA definition above and the truncated SVD dimensionality reduction? (**1 point**)

(b) One method used to compute the PCA of a dataset is called Stochastic Gradient Descent and is shown in Algorithm 1. The values of Z, V are updated to decrease the loss L in the negative direction of the gradient by step η. The procedure is conducted for every cell of X for a total of E many epochs. Each update rule learns the error with respect to one cell (i, j) at a time, hence via the gradient $L_{i,j}$ defoned in line 3.

Algorithm 1 Compute PCA through Stochastic Gradient Descent

Require: Original Data $X \in \mathbb{R}^{N \times M}$, Number of latent dimensions K, Learning Rate η, Number of epochs E
Ensure: Low-rank data $Z \in \mathbb{R}^{N \times K}$, Principal components $V \in \mathbb{R}^{K \times M}$
1: **for** $1, \ldots, E$ **do**
2:   **for** $i = 1, \ldots, N$ $j = 1, \ldots, M, k = 1, \ldots, K$ **do**
3:     $L_{i,j} \leftarrow \left(X_{i,j} - \sum_{k=1}^{K} Z_{i,k} V_{k,j} \right)^2$
4:     $V_{k,j} \leftarrow V_{k,j} - \eta \frac{\partial L_{i,j}}{\partial V_{k,j}}$
5:     $Z_{i,k} \leftarrow Z_{i,k} - \eta \frac{\partial L_{i,j}}{\partial Z_{i,k}}$
6:   **end for**
7: **end for**
8: **return** Z, V

Derive the update rule gradients $\frac{\partial L_{i,j}}{\partial V_{k,j}} =?$, $\frac{\partial L_{i,j}}{\partial Z_{i,k}} =?$ (**3 points**)

(c) Argue how can we find a good value for E? Does it depend on η? (**1 points**)

Exercise 24: PCA - Gradient Descent (5 points)

(a) What is the difference between a full gradient descent and stochastic gradient descent? (**1 point**)

(b) Another method to compute the PCA of a dataset is through gradient descent, where the latent data Z and the principal components Z are updated via computing the full gradient over L, as shown in Algorithm 2.

Algorithm 2 Compute PCA through Gradient Descent

Require: Original Data $X \in \mathbb{R}^{N\times M}$, Number of latent dimensions K, Learning Rate η, Number of epochs E
Ensure: Low-rank data $Z \in \mathbb{R}^{N\times K}$, Principal components $V \in \mathbb{R}^{K\times M}$

```
for 1, ..., E do
    for i = 1, ..., N j = 1, ..., M, k = 1, ..., K do
        V_{k,j} ← V_{k,j} − η ∂L/∂V_{k,j}
        Z_{i,k} ← Z_{i,k} − η ∂L/∂Z_{i,k}
    end for
end for
return Z, V
```

Derive the update rule gradients $\frac{\partial L}{\partial V_{k,j}} =?$, $\frac{\partial L}{\partial Z_{i,k}} =?$ (**3 points**)

(c) Argument on the advantages and disadvantages of both learning algorithms. (**1 points**)

Submission

- Electronically to wistuba@ismll.de. Text submitted as pdf, code submitted as source files. No archives.