Assignment 2

Machine learning
708.064 19S

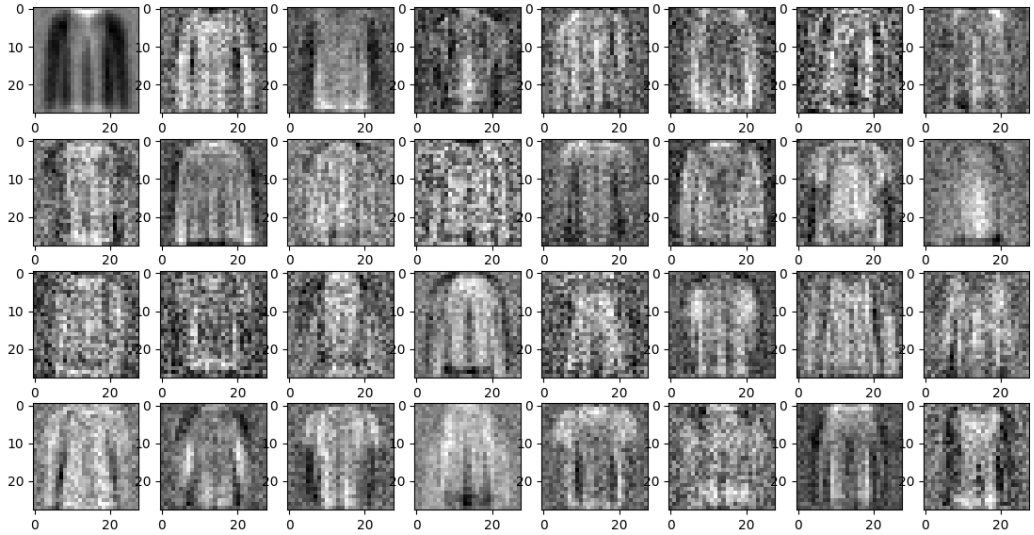
Last name	First name	Matr. Number
Konanur Ramanna	Keerthi Datta	01647641
Gajanovic	Stefan	01431869

1 Lasso and dictionary learning (15 points)

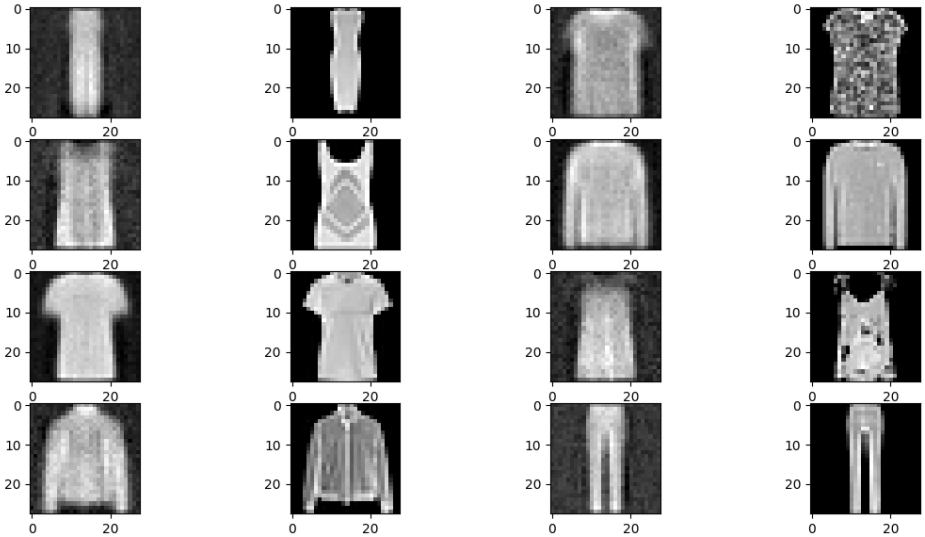
a)

Please find at the end of the assignment

b) D as 28x28 images. Lambda = 0.01



c) Reconstructed images.



In the 1st and 3rd column we see the reconstructed images from dictionary D. Columns 2 and 4 represent corresponding original images.

d) First of all, our dictionary D (computed with lambda = 0.01) gave us 32 “images” each with 784 pixels. In fact, each pixel is actually an atom of dictionary D that has 28 x 28 cells (784) and depth of 32 on each cell. In the representation of every layer from the task 1.b) we can see that they already resemble some cloth pieces (classes), but the images are blurry, and sometimes they seem to be overlapped as is the case in the top-left picture in 1.b) (mixture of jeans and a sweater).

Reconstructed images seem to resemble the original ones precisely. The reconstruction is however, not perfect but that was expected. At least the shape (silhouette) of the clothes is precise, when compared to the original ones.

e) When using the low lambda value (0.001), we observed that the D matrix has more noisy pictures, but the reconstructed images seem to have more sharpness than in examples when using lambda values of 0.01 or 0.1, meaning that the reconstruction is better.
On the other hand, while using the lambda value of 0.1 (high) we observed that the D matrix contained images less blurry and looking more like original pictures, but the reconstruction performance was noticeably worse.
When it comes to the Task 2, performance was also better with smaller lambda.
The reason for that is probably the fact that with the smaller lambda, the entropy loss in smaller, thus the better performance.

2 Logistic Regression (10 points)

a)

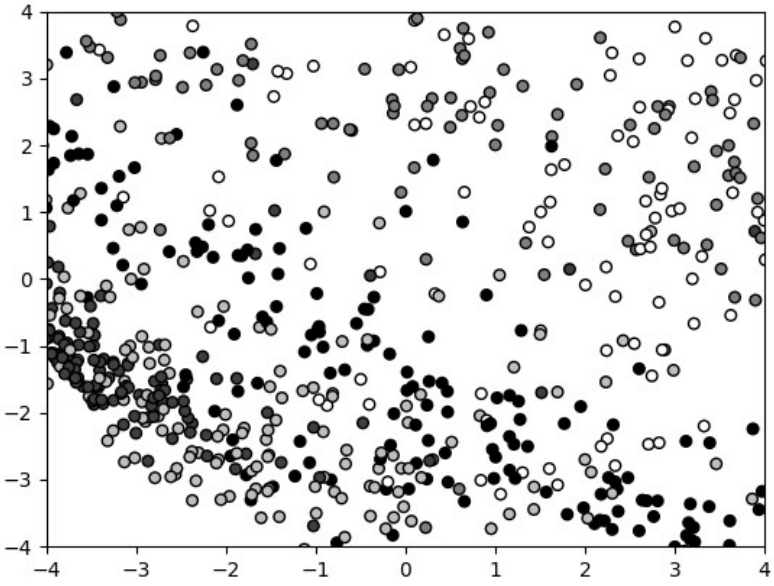
Please find at the end of the assignment

b)
Loss: 1.4207
Accuracy: 0.8465
This performance was achieved with the SGD and only 10 iterations, for each class (0 to 4).

c)
Loss: 0.5658
Accuracy: 0.7544
Regularization term did not improve our performance on the choosen data-set, only decreased cross-entropy loss. But that was expected, as regularization term only improves generalization performance (performance of the new data).

d)
When compared to the performance of Logistic Regression trained of W matrix, we did not see much of a difference regarding the accuracy. The only thing we noticed is that the training time with scaled and centered images is much longer compared with training with W. So we conclude that using W matrix is more efficient.

e)



1) a)

$$L(D,w) = \frac{1}{2}\|Dw - x\|_F^2$$

$$= \frac{1}{2}(Dw-x)^T(Dw-x)$$

$$= \frac{1}{2}(w^TD^TDw - 2\,w^TD^Tx - x^TDw + x^Tx)$$

$$L(D,w) = \frac{1}{2}(w^TD^TDw - 2\,w^TD^Tx + x^Tx) \quad -(1)$$

Taking derivative of $L(D,w) \to (1)$ w.r.t w

$$\nabla_w L(D,w) = \frac{1}{2}(2D^TDw - 2D^Tw)$$

$$\boxed{\nabla_w L(D,w) = D^T(Dw - w)}$$

Taking derivative (1) w.r.t D.

$$\nabla_D L(D,w) = \frac{1}{2}(2Dww^T - 2xw^T)$$

$$\boxed{\nabla_D L(D,w) = (Dw - x)w^T.}$$

2) a)

$$E = -\sum_{j=1}^{N} y_j \log p(C_1 \mid \omega_j) + (1-y_j) \log p(C_2 \mid \omega_j)$$

$$= -\sum_{j=1}^{N} y_j \log(\sigma(\omega_j^T \theta)) + (1-y_j) \log(1-\sigma(\omega_j^T \theta))$$

$$E = -\left(\sum_{j=1}^{N} y_j \log\left(\frac{1}{1+e^{-\omega_j^T \theta}}\right) + (1-y_j) \log\left(1 - \frac{1}{1+e^{-\omega_j^T \theta}}\right)\right)$$

$\nabla_\theta E = ??$ To solve this, we split the sum into two parts.

Taking derivative of $\sum_{j=1}^{N} y_j \log\left(\frac{1}{1+e^{-\omega_j^T \theta}}\right)$

$$= \sum_{j=1}^{N} \frac{y_j}{\frac{1}{1+e^{-\omega_j^T \theta}}} \cdot \frac{(-1)}{(1+e^{-\omega_j^T \theta})^2} \cdot -e^{-\omega_j^T \theta} \cdot \omega_j^T$$

$$= \sum_{j=1}^{N} \frac{y_j \cdot e^{-\omega_j^T \theta} \cdot \omega_j^T}{(1+e^{-\omega_j^T \theta})} \quad \text{— (1)}$$

Taking derivative of $\sum_{j=1}^{N} (1-y_j) \log\left(1 - \frac{1}{1+e^{-\omega_j^T \theta}}\right)$

First of all, solving or simplifying above expression,

$$\Rightarrow \sum_{j=1}^{N} (1-y_j) \log\left(\frac{e^{-\omega_j^T \theta}}{1+e^{-\omega_j^T \theta}}\right) \quad \text{— (2)}$$

Expanding log & taking derivative,

$$\log\left(\frac{e^{-\omega_j^T \theta}}{1+e^{-\omega_j^T \theta}}\right) \Rightarrow \log(e^{-\omega_j^T \theta}) + \log\left(\frac{1}{1+e^{-\omega_j^T \theta}}\right)$$

$$\Rightarrow -w_j^T\theta + \log\left(\frac{1}{1+e^{-w_j^T\theta}}\right)$$

Now taking derivative w.r.t. θ, also from (1)

$$\Rightarrow -w_j^T + \frac{e^{-w_j^T\theta}\cdot w_j^T}{1+e^{-w_j^T\theta}}$$

pluging this to equation (2) & Subsitute

putting it all together →

$$\nabla_\theta E = -\sum_{j=1}^{N}\left(y_j\cdot\frac{e^{-w_j^T\theta}\cdot w_j^T}{1+e^{-w_j^T\theta}} - w_j^T + w_j^T y_j - y_j\frac{e^{-w_j^T\theta}\cdot w_j^T}{1+e^{-w_j^T\theta}} + \frac{e^{-w_j^T\theta}\cdot w_j^T}{1+e^{-w_j^T\theta}}\right)$$

$$\nabla_\theta E = -\sum_{j=1}^{N}\left(-w_j^T + w_j^T y_j + \frac{e^{-w_j^T\theta}\cdot w_j^T}{1+e^{-w_j^T\theta}}\right)$$

$$= \sum_{j=1}^{N}\left(-y_j - \frac{e^{-w_j^T\theta}}{1+e^{-w_j^T\theta}} + 1\right) w_j^T$$

$$= \sum_{j=1}^{N}\left(-y_j + \frac{1}{1+e^{-w_j^T\theta}}\right) w_j^T$$

$$\boxed{\nabla_\theta E = \sum_{j=1}^{N}\left(a(w_j^T\theta) - y_j\right) w_j^T}$$

where $a(w_j^T\theta) = \frac{1}{1+e^{-w_j^T\theta}}$.