Home Assignment 1

Christophe Saad
May 22, 2020

1. (a) In high dimensions, feature vectors are sparse and overfitting could be very common since the pattern could be hardly distinguished from the noise. Regularization aims in solving this problem by adding a regularization term of the form $\lambda||\beta||$ to the loss function. It adds a penalty to the norm of the estimator in order to decrease its variance as well as the weights of irrelevant features. It sets a trade-off between bias (estimated risk) and variance (regularization parameter).

   (b) We run gradient descent on logistic regression with l1 and l2 regularization.

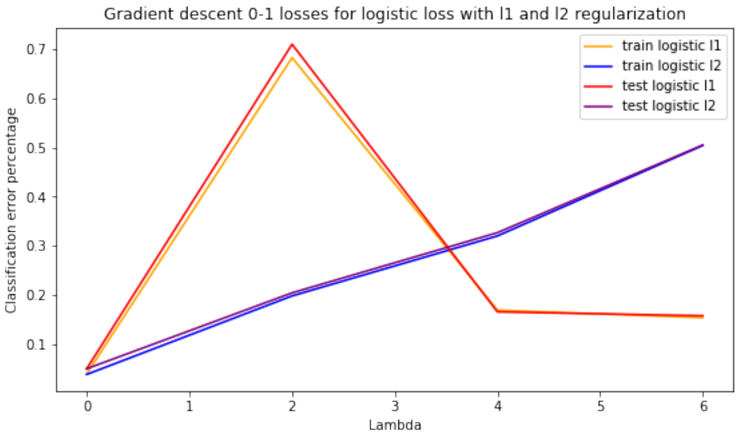

Figure 1: 01-loss for logistic loss with l1 and l2 regularization

   (c) We try to find the optimal value of λ to balance variance and bias in order to reduce overfitting. Since we add a regularization term to the empirical risk (which adds bias to reduce variance), the training error will be minimized for $\lambda = 0$. In order to find the optimal empirical trade-off, we look at the test error for several λ values and we find the value which minimizes it. We use K-fold cross-validation to be more accurate about our model's predictions performance.
   In my case, $\lambda = 0$ seems to be the best value.

   (d) Plots of the resulting estimators.

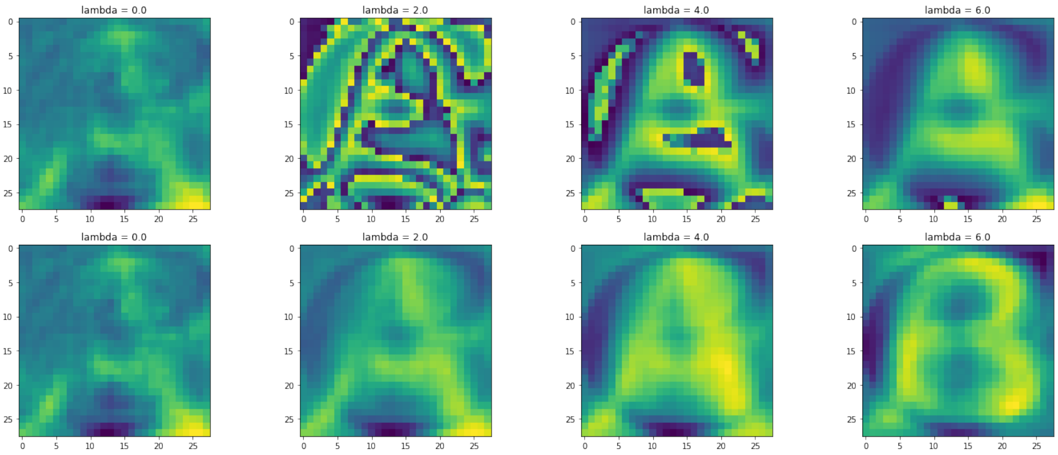
lambda = 0.0
lambda = 2.0
lambda = 4.0
lambda = 6.0
lambda = 0.0
lambda = 2.0
lambda = 4.0
lambda = 6.0

Figure 2: Logistic loss estimators with l1 (first row) and l2 (second row) regularization

(e) Answers in previous questions.

2. (a) From bayes' theorem, we have:

$$\begin{aligned}
\mathbb{P}(Y=1|X) &= \frac{\mathbb{P}(X|Y=1)\mathbb{P}(Y=1)}{\mathbb{P}(X|Y=0)\mathbb{P}(Y=0)+\mathbb{P}(X|Y=1)\mathbb{P}(Y=1)} \\
&= \frac{\frac{1}{\sqrt{(2\pi)^d|\Sigma|}}\pi e^{-\frac{1}{2}(X-\mu_1)^T\Sigma^{-1}(X-\mu_1)}}{\frac{1}{\sqrt{(2\pi)^d|\Sigma|}}\left[(1-\pi)e^{-\frac{1}{2}(X-\mu_{-1})^T\Sigma^{-1}(X-\mu_{-1})}+\pi e^{-\frac{1}{2}(X-\mu_1)^T\Sigma^{-1}(X-\mu_1)}\right]} \\
&= \frac{1}{1+\frac{(1-\pi)e^{-\frac{1}{2}(X-\mu_{-1})^T\Sigma^{-1}(X-\mu_{-1})}}{\pi e^{-\frac{1}{2}(X-\mu_1)^T\Sigma^{-1}(X-\mu_1)}}} \\
&= \frac{1}{1+e^{-\frac{1}{2}\log(\frac{1-\pi}{\pi})[(X-\mu_{-1})^T\Sigma^{-1}(X-\mu_{-1})-(X-\mu_1)^T\Sigma^{-1}(X-\mu_1)]}} \\
&= \frac{1}{1+e^{-\frac{1}{2}\log(\frac{1-\pi}{\pi})\left[\mu_{-1}^T\Sigma^{-1}\mu_{-1}-\mu_1^T\Sigma^{-1}\mu_1+2X^T\Sigma^{-1}(\mu_1-\mu_{-1})\right]}} \\
&= \frac{1}{1+e^{-\beta_0-\beta_1^TX}}
\end{aligned}$$

with $\beta_0 = \frac{1}{2}\log(\frac{1-\pi}{\pi})\left[\mu_{-1}^T\Sigma^{-1}\mu_{-1} - \mu_1^T\Sigma^{-1}\mu_1\right]$ *and* $\beta_1 = \log(\frac{1-\pi}{\pi})(\mu_1-\mu_{-1})^T\Sigma^{-1}$

(b) Taking $\beta_0 = 0$

$$\begin{aligned}
\mathbb{P}(Y=0|X) &= 1-\mathbb{P}(Y=1|X) \\
&= \frac{1}{1+e^{\beta_1^TX}}
\end{aligned}$$

We therefore have:

$$\mathbb{P}(Y|X) = \frac{1}{1+e^{-Y\beta_1^TX}}$$

The log-likelihood of β_1 for the centered LDA is:

$$\hat{\beta}_1 \in \arg\max_{\beta} \frac{1}{n}\sum_{i=1}^{n}\log(\frac{1}{1+e^{-Y_i\beta^TX_i}})$$

$$\hat{\beta}_1 \in \arg\min_{\beta} \frac{1}{n}\sum_{i=1}^{n}\log(1+e^{-Y_i\beta^TX_i})$$

The maximum likelihood of β_1 is the same as the solution of the logistic regression with $\lambda = 0$ since it reduces in solving the same optimization problem.

(c)

(d) A false positive is a classification error in which we wrongfully report the presence of a condition. A false negative is also a classification error in which we falsly report the abscence of a condition.
A confusion matrix splits the classification error into these two categories, giving more details about the behaviour of our model. It is better than just looking at the classification error as it allows us to locate the defect in case of bad classification performance.

3. (a) We implement a PCA algorithm and check its performance by plotting the resulting explained variance of each new feature.

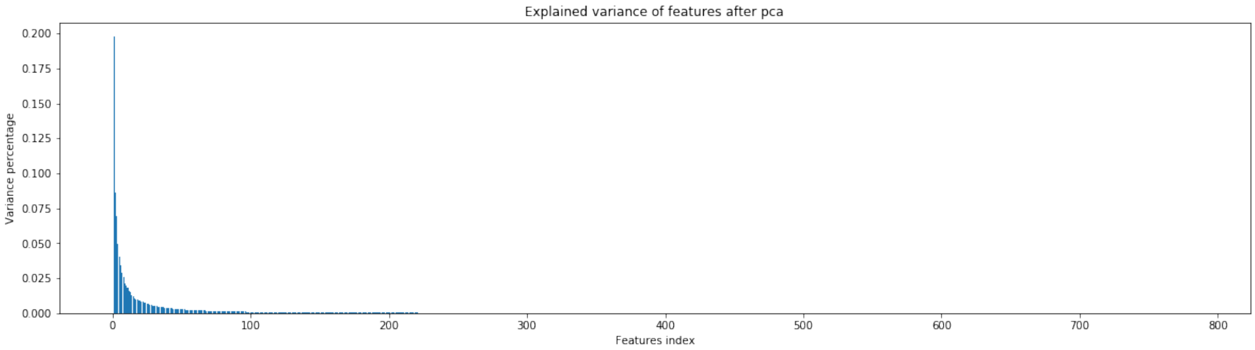

Figure 3: Explained variance

We run a K-means algorithm and plot the projected features and the centers onto their two principal components.

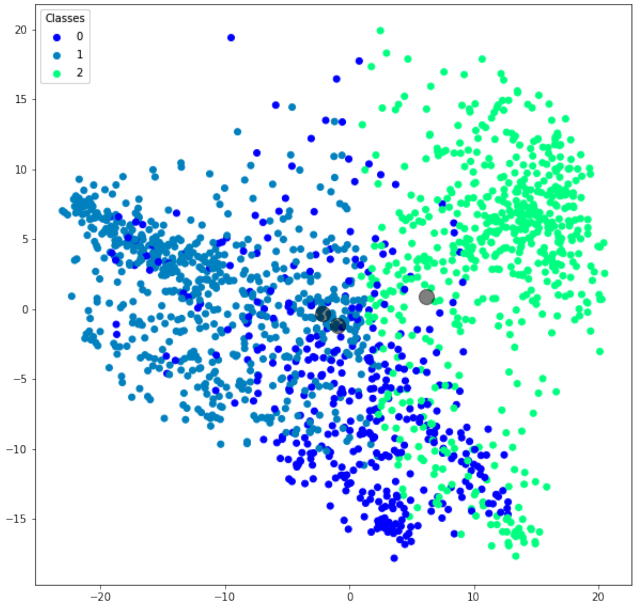

Figure 4: K-means classes

(b) In order to assign a label to our clusters, we assume that we manage to predict the majority of the cluster, i.e. we make more correct predictions than false ones. We assign to each cluster the class which is most frequent. It is the same assumption we make for computing the purity of clusters (evaluation measure). Let $c \in \{A, B, C\}$ be our labels and l_c our classes. Cluster k_i will be assigned

$$\hat{c}_i = \arg\max_c |k_i \cap l_c|$$

We therefore assign cluster (0) to letter B, cluster (1) to A and cluster (2) to C.
We plot the confusion matrix.

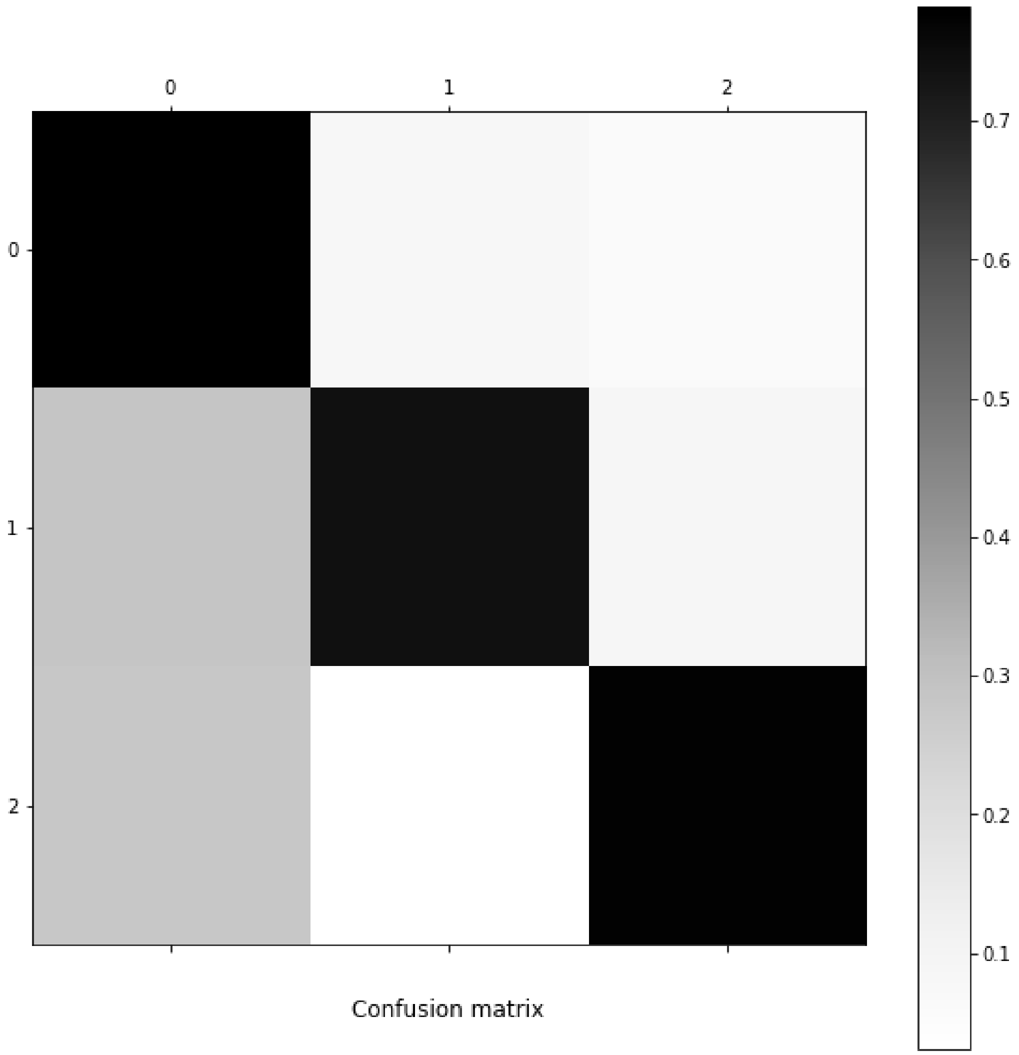

Figure 5: Confusion matrix