From Basic Machine Learning models to Advanced Kernel Learning

Zineb Et-tarraf

November 2022

1 Question 1

- The inputs are gray-scale images of size $(28, 28, 1)$. Theses images are of depth one where each pixel represents a single value in $[\![0\ ;\ 255]\!]$ .So the input space can be represented by $\mathcal{X} = [\![0\ ;\ 255]\!]^{28\times 28}$
It is of dimension $28 \times 28 \times 1 = 784$
- The output is $Y_i = 1$ if image i is "A" and -1 otherwise (if the image is "B" or "C").So the output space is : $\mathcal{Y} = \{-1, 1\}$
It is of dimension 2.
- The training data set is a set of 6000 labeled images $\mathcal{S} = (X_i, Y_i)_{1\leq i\leq 6000}$
It is of dimension $784 \times 6000 \times 2$

2 Question 2

2.1 Question a

The true risk associated with the 0-1 loss is :

$$R(f) = E[1_{f_\theta(X_i)\neq Y_i}]$$

The empirical risk over S is the unbiased estimator of the true risk :

$$\hat{R}(f_\theta, S) = \frac{1}{n}\sum_{i=1}^{n} 1_{f_\theta(X_i)\neq Y_i}$$

The 0-1 loss function $1_{f_\theta(X_i)\neq Y_i}$ is not continuous (and thus also non-convex) and leads to difficult optimization problems.
For example, gradient methods cannot be applied because the function to be minimized has no minimum and is not differential. Hence the complexity of minimizing the empirical risk in this case.

2.2 Question b

To assess the performance of a trained model, it is essential to evaluate it on unseen data provided by a test set. This is to check if the algorithm has been trained effectively or not.

2.3 Question c

The optimisation problem is to find predictor's parameters by minimising the empirical risk .This leads :

- In the case of the linear least square regression algorithm, the optimization problem can be written :

  $$\theta_{optimal} = argmin_{\theta \in R^{m+1}} \frac{1}{n} \sum_{i=1}^{n} (\theta^T \tilde{X}_i - Y_i)^2$$

  where :

  $$\tilde{X}_i = (1, X_{i,1}, ..., X_{i,m})$$

- For the logistic regression , the optimization problem can be written :

  $$\theta_{optimal} = argmin_{\theta \in R^{p+1}} \frac{1}{n} \sum_{i=1}^{n} log(1 + e^{-Y_i \cdot (\theta^T \tilde{X}_i)})$$

  where :

  $$\tilde{X}_i = (1, X_{i,1}, ..., X_{i,m})$$

2.4 Question d

The Logistic Regression model does not make any assumption except that :

$$\ln(\frac{P(Y=1|x)}{P(Y=0|x)}) = f_\theta(x)$$

Thus :

$$\frac{P(Y=0|x)}{P(Y=1|x)} = e^{-f_\theta(x)} = \frac{1 - P(Y=1|x)}{P(Y=1|x)}$$

By extracting $P(Y=1|x)$ from this equation , we get :

$$P(Y=1|x) = \frac{1}{1 + e^{-f_\theta(x)}}$$

2.5 Question e

The log-likelihood is defined by :

$$LL = \ln(\prod_i p(y_i|x_i)) = \sum_i \ln(p(y_i|x_i))$$

Since y is in 0,1 So :

$$p(y|x) = p(y=1|x)^y \times (1 - p(y=1|x))^{1-y}$$

From the previous question we have :

$$p(y=1|x) = \sigma(f_\theta(x))$$

Therefore the log-likelihood can be written as :

$$LL = \sum_i y \ln(\sigma(f_\theta(x))) \times (1-y)(1-\sigma(f_\theta(x)))$$

If we define $l(y,z) = y \ln(\sigma(z) \times (1-y)(1-\sigma(z))$

$$argmax_{\theta \in R^{m+1}} LL = argmin_{\theta \in R^{m+1}} \sum_i l(yi, f_\theta(x_i))$$

Hence the logistic regression estimator is therefore the maximum likelihood .

Likelihood assumes that the data is drawn from a certain probability distribution and provides a way to measure the goodness of fit of a model.

3 Question 3

-As the degree of the polynomials increases, the complexity of the model increases.
- When performing least-squares polynomial regression , the test error goes down at first because the model is not complex enough and it's still making too much misclassifications since the model's parameters are not updated enough.When the degree of our polynomials increases the model becomes more and more expressive but at some point its complexity is too lage (overfitting) and the result model fits exactly the seen data but it's making too misclassifications for the unseen data , thus the test error goes up.

4 Question 4

By computing the gradient of the empirical risk of each algorithm we conclude the following update rules for the gradient descent :

The linear least-squares regression

$$\theta \leftarrow \theta - 2 \times \eta \times (f_\theta(X_i) - Y_i)\tilde{X}_i$$

The logistic regression

$$\theta \leftarrow \theta + \eta \times Y_i \times (1 - sigmoid(Y_i f_\theta(X_i)))\tilde{X}_i$$

The perceptron algorithm

if $f_\theta(X_i) \neq Y_i$

$$\theta \leftarrow \theta + \eta \times Y_i \times \tilde{X}_i$$

5 Question 5

This question is implemented on the notebook .We bring here some resulting figures (the comments are one the notebook).

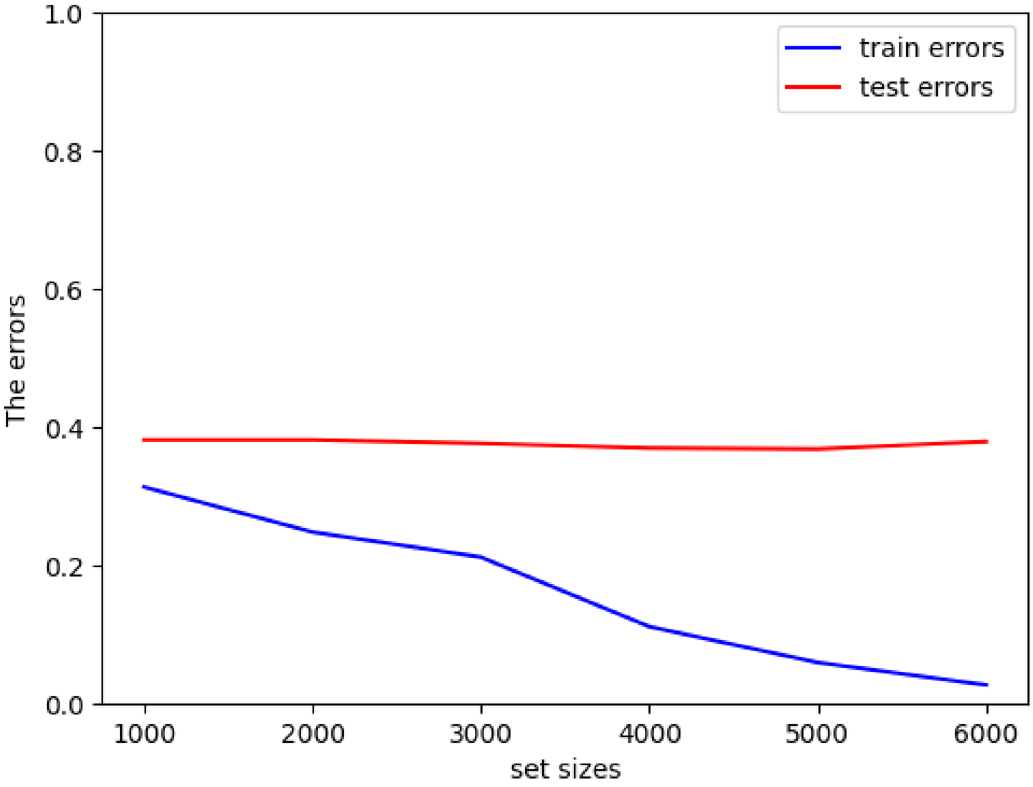

Figure 1: Training and test error on mutiple training set of different sizes

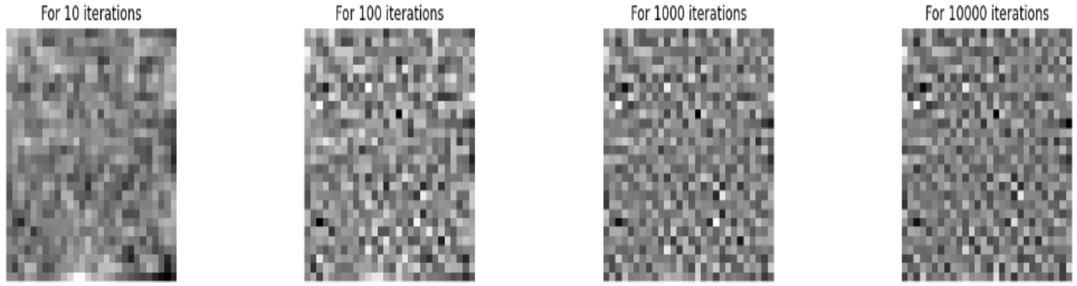

Figure 2: Plotting estimators as images for four logistic regression models trained on 10 ,100,1000,1000 iterations

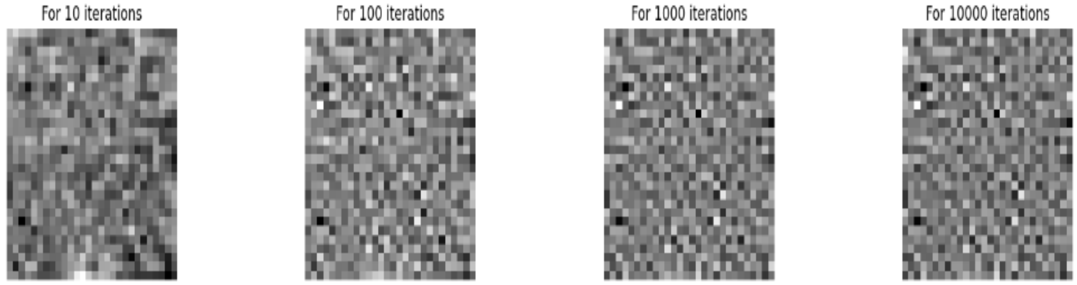

Figure 3: Plotting estimators as images for four OLS models trained on 10 ,100,1000,1000 iterations

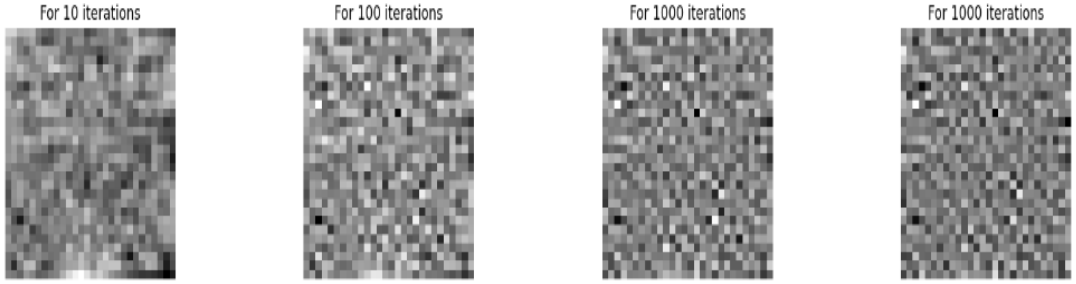

Figure 4: Plotting estimators as images for four perceptron models trained on 10 ,100,1000,1000 iterations

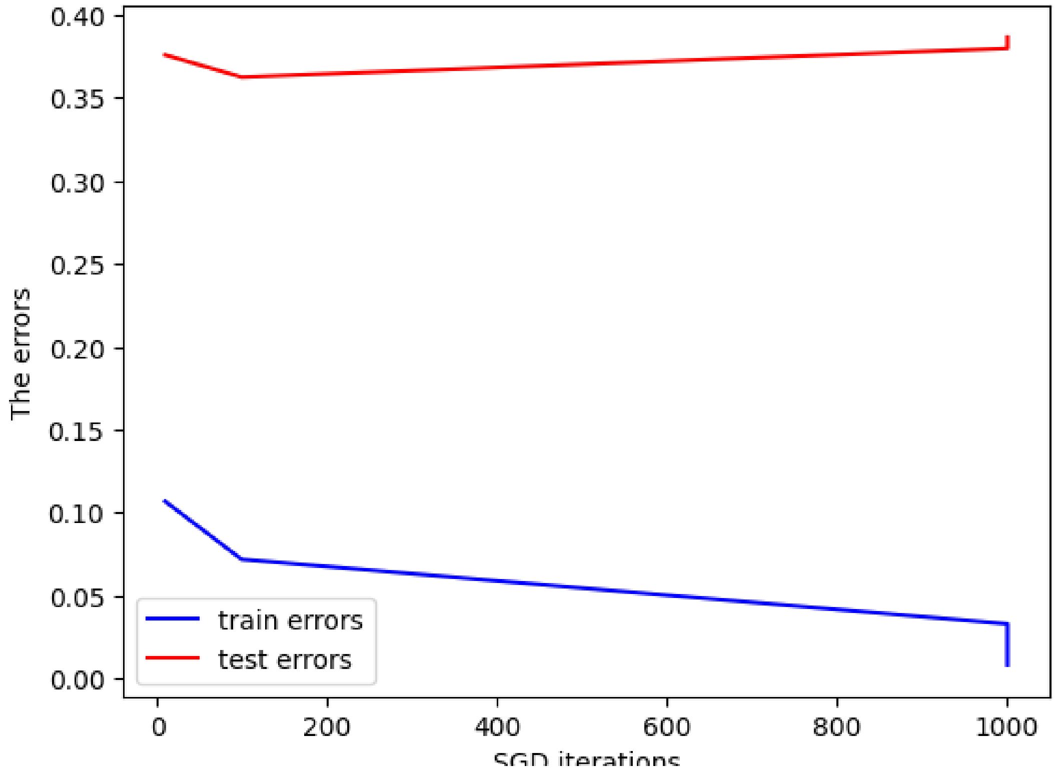

Figure 5: train and test error of the SGD of perceptron algorithm over 10,100,1000,10000 iterations

6 Question 6

6.1 Question a

let (X_i, Y_i) a sample from our training set and we denote by $\mathcal{N}$ $((X_i, Y_i))$ the set of k sample (k closest examples in the training data set).In our case we can use the euclidean distance as a metric for the k-nearest neighbors classification rule.
In the case of a 2 class classification the output of the KNN algorithm is :

$$y_{pred} \quad = \quad sign(\frac{1}{k} \sum_{Y_j \in \mathcal{N}(\mathcal{X}_\rangle, \mathcal{Y}_\rangle)} Y_j)$$

Which is the case since we want to classify the data into two clusters class A and class not A .

6.2 Question b

In the notebook

6.3 Question c

In the notebook

7 Question 7

The number of parameters need to be trained is :$(28 \times 28) \times 32 + 32 \times 3$
he code is on the notebook

8 Question 8

	logistic regression	OLS	Perceptron	MLP
Empirical error (0-1 loss)	0.011	0.0453.	0.0043	0.0283
Test error (0-1 loss)	0.0546	0.0526	0.0753	0.086

We can conclude that the logistic regression has a higher accuracy than the OLS and the Perceptron.

9 Question 9

By adding the regularization term we recompute the gradient of the empirical risk of each algorithm .We conclude the following update rules for the gradient descent :

The linear least-squares regression

$$\theta \leftarrow \theta - 2 \times \eta \times (f_\theta(X_i) - Y_i)\tilde{X}_i - 2\lambda\eta\theta$$

The logistic regression

$$\theta \leftarrow \theta + \eta \times Y_i \times (1 - \frac{1}{1 + e^{-Y_i f_\theta(X_i)}})\tilde{X}_i - 2\lambda\eta\theta$$

The perceptron algorithm
if $f_\theta(X_i) \neq Y_i$

$$\theta \leftarrow \theta + \eta \times 0.5 \times (Y_i - f_\theta(X_i)) \times \tilde{X}_i - 2\lambda\eta\theta$$

Indeed the regularization is added to calibrate models in order to minimize the adjusted loss function and prevent overfitting or underfitting.