IN4320 Machine Learning Assignment 1

Dilan Gecmen 4221168

February 28, 2018

Note: Worked with Jasper Hemmes

Some Optima & Geometry

1.

Assume $d = 1$. Also assume that we are in a situation where we know r_- is fixed to 1, so we only have to optimize for r_+. The only observations that we have for that + class are $x_1 = -1$ and $x_2 = 1$. The loss function L becomes:

$$
\begin{aligned}
L(1, r_+) &= \left(\sum_{i=1}^{2} \frac{1}{N_+} ||x_i - r_+||_2^2 \right) + \lambda ||1 - r_+||_1 \\
&= \frac{1}{2}(x_i - r_+)^2 + \frac{1}{2}(x_2 - r_+)^2 + \lambda |1 - r_+| \\
&= \frac{1}{2}(-1 - r_+)^2 + \frac{1}{2}(1 - r_+)^2 + \lambda |1 - r_+| \\
&= 1 + r_+^2 + \lambda |1 - r_+| \qquad\qquad = L(r_+)
\end{aligned}
$$

a. We use Matlab to draw the loss function $L(r_+)$ as a function of r_+ $\forall \lambda \in \{0, 1, 2, 3\}$, see Figure 1.

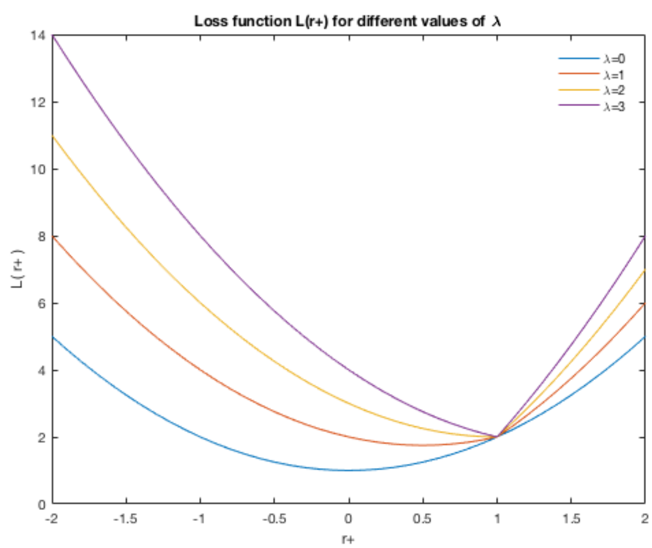


Figure 1: Loss function $L(r_+)$ for different values of λ

b. We can represent the loss function $L(r_+)$ as follows:

$$L(r_+) = 1 + r_+^2 + \lambda|1 - r_+| = \begin{cases} 1 + r_+^2 + \lambda(1 - r_+) & \text{if } r_+ < 1 \\ 1 + r_+^2 - \lambda(1 - r_+) & \text{if } r_+ > 1 \end{cases}$$

The derivative of the loss function is:

$$\frac{\partial L(r_+)}{\partial r_+} = \begin{cases} 2r_+ - \lambda & \text{if } r_+ < 1 \\ 2r_+ + \lambda & \text{if } r_+ > 1 \end{cases}$$

When $\lambda = 0$

$$\frac{\partial L(r_+)}{\partial r_+} = 0 \Longrightarrow r_+ = 0$$

The minimizer is $r_+ = 0$ and the minimum value of the loss function is $\min\{L(r_+)\} = \min\{L(0)\} = 1$.

When $\lambda = 1$

$$\frac{\partial L(r_+)}{\partial r_+} = 0 \Longrightarrow \begin{cases} r_+ = \frac{1}{2} & \text{if } r_+ < 1 \\ r_+ = -\frac{1}{2} & \text{if } r_+ > 1 \end{cases}$$

We see that $r_+ = \frac{1}{2}$ is the minimizer ($r_+ = -\frac{1}{2}$ for $r_+ > 1$ does not hold). The minimum value now becomes $\min\{L(r_+)\} = \min\{L(\frac{1}{2})\} = 1 + (\frac{1}{2})^2 + (1 - \frac{1}{2}) = \frac{7}{4}$

When $\lambda = 2$
There is no r_+ such that $\frac{\partial L(r_+)}{\partial r_+} = 0$. To find the minimizer we look at the loss function:

$L(r_+) = 1 + r_+^2 + 2|1 - r_+|$

We know that $|1 - r_+|$ is nonnegative $\forall r_+$. So the minimum value is attained when $r_+ = 1$. We see that this is indeed the case when we look at the plot of the function for $\lambda = 2$. Hence, the minimizer is $r_+ = 1$ and the minimum value of the loss function is $\min\{L(r_+)\} = \min\{L(1)\} = 2$

When $\lambda = 3$
There is no r_+ such that $\frac{\partial L(r_+)}{\partial r_+} = 0$. To find the minimizer we look at the loss function:

$L(r_+) = 1 + r_+^2 + 3|1 - r_+|$

We know that $|1 - r_+|$ is nonnegative $\forall r_+$. So the minimum value is attained when $r_+ = 1$. We see that this is indeed the case when we look at the plot of the function for $\lambda = 3$. Hence, the minimizer is $r_+ = 1$ and the minimum value of the loss function is $\min\{L(r_+)\} = \min\{L(1)\} = 2$

2.

We have the following loss function L:

$$L(r_-, r_+) := \left(\sum_{i=1}^{N} \frac{1}{N_{y_i}} ||x_i - r_{y_i}||_2^2 \right) + \lambda ||r_- - r_+||_1$$

The regularizer enforces that the representors r_- and r_+ are brought closer together. If λ gets larger and larger the regularizer term becomes dominant in the loss function and the representors get closer and closer. If $\lambda \to \infty$ then $r_- \to r_+$. When the representors are at the same position the error becomes $\epsilon = \frac{1}{2}$.

3.

We now consider the setting in which both representors have to be determined through a minimization of the loss. Still, $d = 1$, so we have $L : \mathbb{R}^2 \to \mathbb{R}$

a. When we are trying to find two 1-dimensional representors r_- and r_+ the contour lines are a concatenation of two basic geometric shapes, which are ellipses and lines in the (r_-, r_+)-plane.

When λ is very large for which $r_+ = r_-$ the contour lines are concatenations of ellipsoids, which are concatenated on the same line.

One could also express the regularizer as a constraint in the minimization problem. The contour lines of the objective function are ellipsoids (residual sum of squares have elliptical contours) and the constraint becomes a square. The contours are now ellipsoids concatenated on a line of the square.

b. The loss function becomes

$$L(r_-, r_+) = \frac{1}{2}(-1-r_+)^2+\frac{1}{2}(1-r_+)^2+\frac{1}{2}(3-r_-)^2+\frac{1}{2}(-1-r_-)^2+\lambda|r_- - r_+|$$

For large λ the regularizer in the loss function becomes the dominant term. This means that the solution of the loss function lies on the line $r_+ = r_-$. For large enough λ we can substitute $r_+ = r_-$ in the loss function:

$$\begin{aligned} L(r_+) &= \frac{1}{2}(-1 - r_+)^2 + \frac{1}{2}(1 - r_+)^2 + \frac{1}{2}(3 - r_+)^2 + \frac{1}{2}(-1 - r_+)^2 \\ &= 2r_+^2 + 6 - 2r_+ \end{aligned}$$

We set the derivative of L with respect to r_+ equal to zero to find the minimum of the parabola.

$$\frac{\partial L(r_+)}{\partial r_+} = 0 \Longrightarrow r_+ = \frac{1}{2}$$

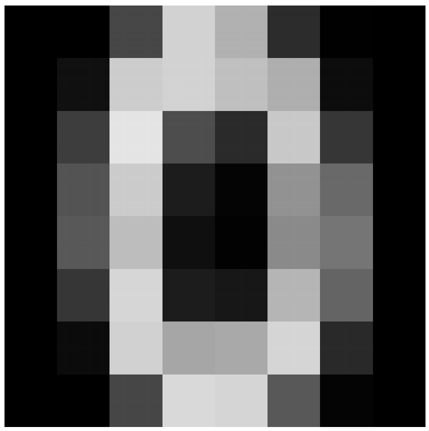
$(r_-, r_+) = (\frac{1}{2}, \frac{1}{2})$ is the minimizer of the loss function with minimum value $\min\{L(r_+)\} = 5.5$ For large enough λ the optimal solution becomes $(r_-, r_+) = (\frac{1}{2}, \frac{1}{2})$

4.

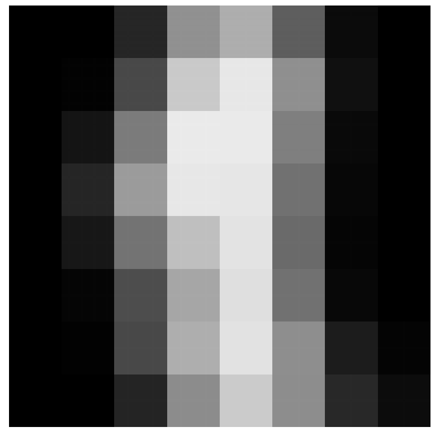
a. We solve for minimizers (r_+, r_-) minimizing the loss function. In this program we will use MatLab function fminunc to minimize, this because we are dealing with an unconstrained problem and the fast convergence due to the quasi-euler minimization, an iterative gradient descent algorithm. The algorithm stops iterating when:

- The stepsize is below a treshold
- The gradient of the function is below a treshold

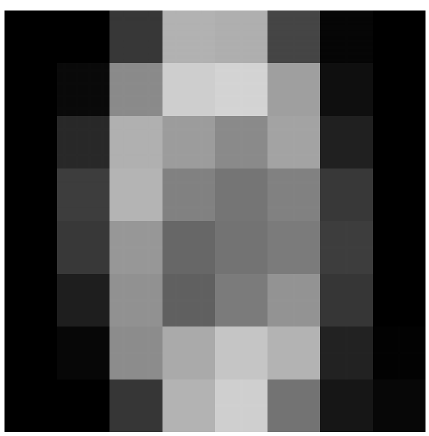
b. When we look at Figure 2. The representors are as we would expect. The unregularized representors become the average zero and one respectively. When $\lambda \to \infty$ then $r_0 \to r_1$, which we can clearly see as both images look the same and the representors have become an average of a zero and a one.



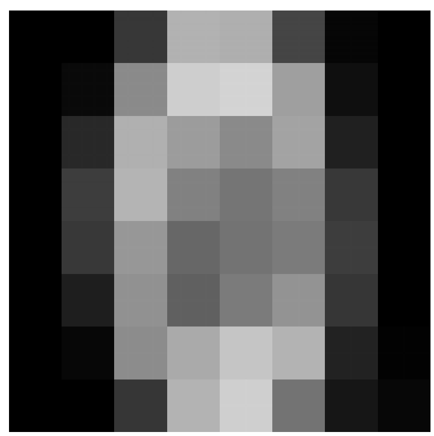
(a) Plot of r_0 for $\lambda = 0$



(b) Plot of r_1 for $\lambda = 0$



(c) Plot of r_0 for $\lambda = 1000$



(d) Plot of r_1 for $\lambda = 1000$

Figure 2: Plots of the representors for $\lambda = 0$ & $\lambda = 1000$

c. When we look at Figure 3 we can see that for low λ is low the results are as we expected them to be. When λ the apparent error is equal to 0, as we trained the model with a single sample, and test with this same sample. Hence, this will give us no error. On the contrary, the true error is nonzero, as we trained the model to recognize a 1 or 0 with a single sample. This works in most cases, but still errors are made. Now when λ is increased (to λ=100) we see in Figure 3 that the true error decreases and the apparent error still remains equal to zero, the true error decreases because the optimal λ lies around $\lambda = 100$. When we look at a further increase of λ, let us say $\lambda = 10^3$we see that the apparent error and the true error increase. This because of the representors for both classes are very close to each other. In this case the algorithm makes a lot of errors when classifying the test samples. Note that we choose in 4b $\lambda = 1000$ such that the solution does not change. But if we look at Figure 3 we see that the error is not around $\epsilon = \frac{1}{2}$, which is what we expect when the representors are similar. When we look at Figure 2 we see that for $\lambda = 1000$ the images seem the same. The algorithm still makes an error of around 30%. If we increase the λ then the error indeed goes to $\epsilon = \frac{1}{2}$

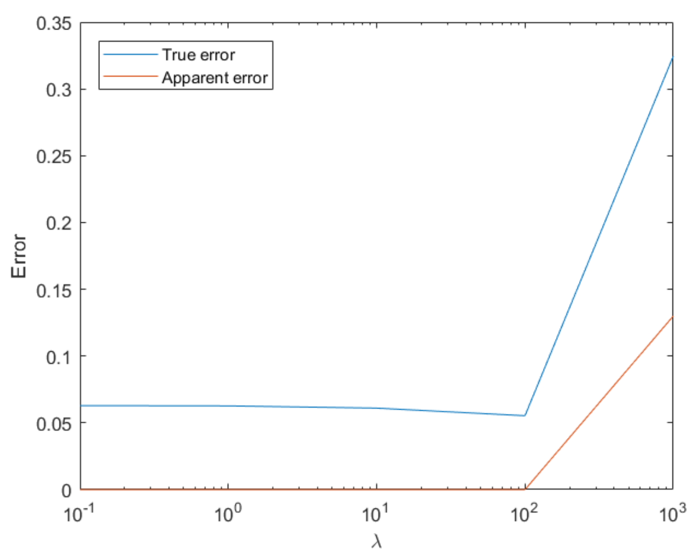


Figure 3: True & apparent error