**Deep Learning - 236606 - HW1**

Ido Glanz - 302568936 Matan Weksler - 302955372

1. **Distance of point to hyperplane**
   1. Given a hyperplane and a point ;

We will write the constrained optimization problem whose solution is the distance from to

* 1. We will use the Lagrange multipliers method to solve the optimization problem:

We than derivate the equation w.r.t x and and equate to zero:

Substituting (1) in to (2):

Substituting back:

We now have the (due to f being monotone). Calculating the distance between the point we found and we obtain:

And we got the formula obtained in class

1. **Perceptron**
   1. Prove where:

Proof:

* 1. We now prove that if a linear separator with margin exsits for S, then the margin perceptron algorithm, will make at most mistakes on S.

We will first show :

Notice;

We saw in class that: due to x being on the unit sphere, and on x. Hence taking the root of the above we obtain:

Using Taylor expansion, we can substitute :

We will now see what happens when :

We saw in class , now after M iteration we obtain (similar to what was done in the lecture):

and for :

1. **Ridge Regression**

Given the ERM training strategy:

we will first rearrange the equation above:

we denote: as b is a scalar, and as the samples are centered hence it’s average times b = 0.

We now have:

* 1. We will now compute :
  2. And for the derivative w.r.t w we get:
  3. Assuming , infer the solution obtained by equating the derivatives from 3(A) and 3(B) to 0:

In the next section, we will show why is positive definite and hence invertible. Also, being positive definite there is a single solution and it is the global minimum of the function.

* 1. In the last section we obtained . Assuming , we will prove M is invertible due to being positive definite:

is a gram matrix and is hence positive semidefinite (meaning is it ), , as is greater than 0 and so M is positive-definite.

* 1. The advantages of the Ridge Regression method is as we’ve seen above, as long as , M is positive-definite and hence invertible, leading to a single and global minimum for our problem.

1. **Binary Classification using Ridge Regression:**

Training Set:

------------- Lambda = 1e-05-------------

Analytic: 0/1 Loss = 0.10506000000000001, Square loss = 0.3848223593949477

Gradient Decent: 0/1 Loss = 0.15026, Square loss = 0.4770427522576504

Validation Set:

------------- Lambda = 1e-05-------------

Analytic: 0/1 Loss = 0.09620000000000001, Square loss = 0.3681495912984633

Gradient Decent: 0/1 Loss = 0.138, Square loss = 0.455714139091732

Training Set:

------------- Lambda = 0.0001-------------

Analytic: 0/1 Loss = 0.10534, Square loss = 0.3852902716901684

Gradient Decent: 0/1 Loss = 0.15026, Square loss = 0.4770489997987863

Validation Set:

------------- Lambda = 0.0001-------------

Analytic: 0/1 Loss = 0.09670000000000001, Square loss = 0.3680395483842602

Gradient Decent: 0/1 Loss = 0.138, Square loss = 0.45572133256623004

Training Set:

------------- Lambda = 0.001-------------

Analytic: 0/1 Loss = 0.10674000000000002, Square loss = 0.38702297617025144

Gradient Decent: 0/1 Loss = 0.15026, Square loss = 0.47711155190041343

Validation Set:

------------- Lambda = 0.001-------------

Analytic: 0/1 Loss = 0.09680000000000001, Square loss = 0.3673293410755911

Gradient Decent: 0/1 Loss = 0.13820000000000002, Square loss = 0.4557933419050195

Training Set:

------------- Lambda = 0.01-------------

Analytic: 0/1 Loss = 0.11402000000000001, Square loss = 0.3984509927523093

Gradient Decent: 0/1 Loss = 0.15038, Square loss = 0.4777446490293447

Validation Set:

------------- Lambda = 0.01-------------

Analytic: 0/1 Loss = 0.1037, Square loss = 0.3742663699065845

Gradient Decent: 0/1 Loss = 0.1383, Square loss = 0.4565208008240802

Training Set:

------------- Lambda = 0.1-------------

Analytic: 0/1 Loss = 0.13224000000000002, Square loss = 0.4384801661382786

Gradient Decent: 0/1 Loss = 0.15152000000000002, Square loss = 0.48474661759407917

Validation Set:

------------- Lambda = 0.1-------------

Analytic: 0/1 Loss = 0.1188, Square loss = 0.4144005145227963

Gradient Decent: 0/1 Loss = 0.1398, Square loss = 0.4644445220180887

Training Set:

------------- Lambda = 1-------------

Analytic: 0/1 Loss = 0.16038000000000002, Square loss = 0.5705970170330118

Gradient Decent: 0/1 Loss = 0.1615, Square loss = 0.5746047505802111

Validation Set:

------------- Lambda = 1-------------

Analytic: 0/1 Loss = 0.1516, Square loss = 0.5570021917188126

Gradient Decent: 0/1 Loss = 0.1526, Square loss = 0.5612373901061849

Training Set:

------------- Lambda = 10-------------

Analytic: 0/1 Loss = 0.19602000000000003, Square loss = 0.8678484668338203

Gradient Decent: 0/1 Loss = 0.19282000000000002, Square loss = 0.8678568706896871

Validation Set:

------------- Lambda = 10-------------

Analytic: 0/1 Loss = 0.19240000000000002, Square loss = 0.8656610325813602

Gradient Decent: 0/1 Loss = 0.1889, Square loss = 0.8656336480669078

Training Set:

------------- Lambda = 100-------------

Analytic: 0/1 Loss = 0.39388, Square loss = 0.9831052092037058

Gradient Decent: 0/1 Loss = 0.67034, Square loss = 4.608377068721547

Validation Set:

------------- Lambda = 100-------------

Analytic: 0/1 Loss = 0.3956, Square loss = 0.9830812896333874

Gradient Decent: 0/1 Loss = 0.6659, Square loss = 4.485072791010506

**Best Lambda for analytic: 1e-05**

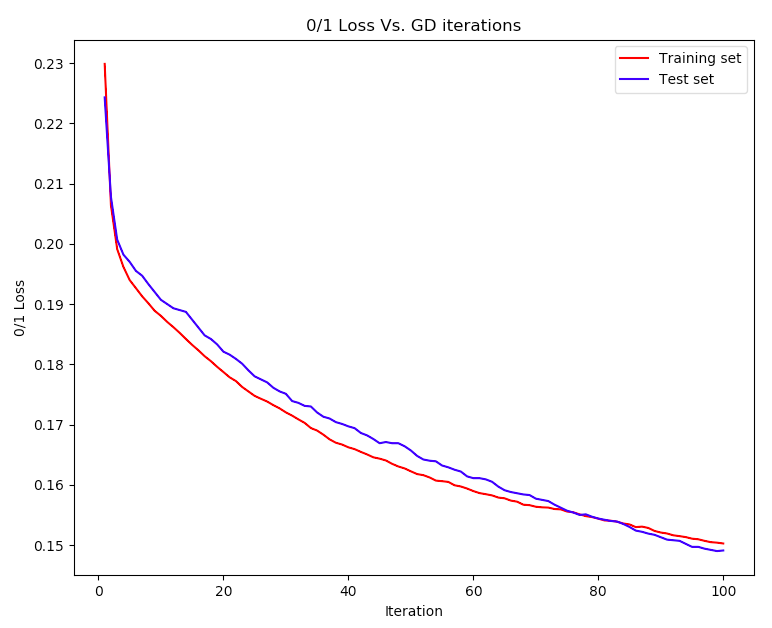
**Best Lambda for GD: 1e-05**

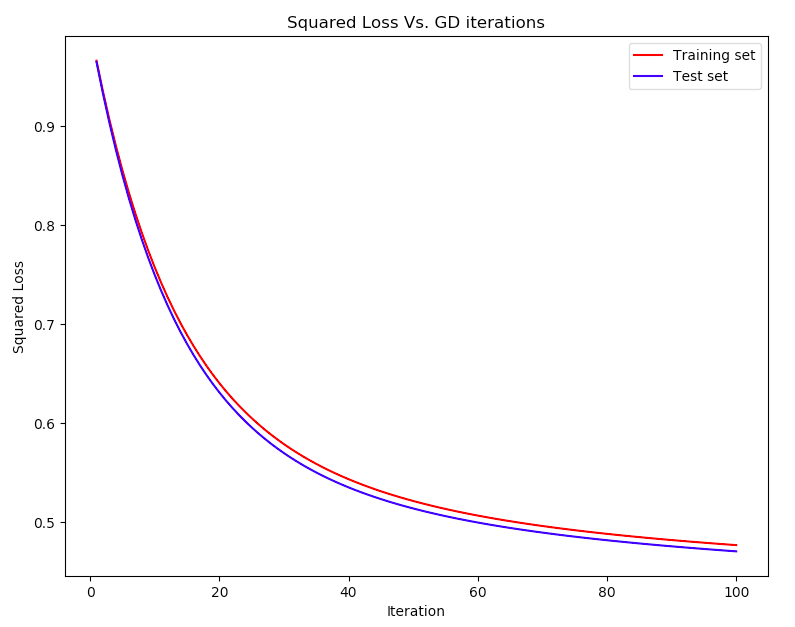
**Test Set:**

**------------- For the best Lambda found (different for each scheme) -------------**

**Analytic: Lambda = 1e-05, 0/1 Loss = 0.10590000000000001, Square loss = 0.3884744552245516**

**Gradient Decent: Lambda = 1e-05, 0/1 Loss = 0.1491, Square loss = 0.470738027690171**





A simple case where the linear regression is a bad idea would be in the XOR example as we saw in class. As this function is not linearly separable, it could not output a good classifier. Another simple example would be any problem not linearly separable and hence not suitable for this kind of classification.