CM146, Fall 2017
Problem Set 02: Perceptron and Regression
Due Feb 6,2018

1 Problem 1

(a) Two possible lines: $y = X_1 + X_2 - 2.1$, $y = X_1 + X_2 - 2.2$.

(b) XOR is not going to be linearly separable. This is because the convex hulls of data points that map to 0 and to 1 intersect.

2 Problem 2

(a)

$$\begin{aligned}\frac{\partial J}{\partial \theta_j} &= -\sum_{n=1}^{N}\left[y_n \frac{x_j}{h_\theta(\mathbf{x}_n)} + (y_n - 1)\frac{x_j}{1 - h_\theta(\mathbf{x}_n)}\right] \\ &= \frac{y_n - h_\theta(\mathbf{x}_n)}{h_\theta(\mathbf{x}_n)[1 - h_\theta(\mathbf{x}_n)]} x_j\end{aligned}$$

3 Problem 3

(a)

$$\frac{\partial J}{\partial \theta_0} = \sum_{n=1}^{N} 2(\theta_0 + \theta_1 \mathbf{x}_{n,1} - y_n) w_n$$

$$\frac{\partial J}{\partial \theta_1} = \sum_{n=1}^{N} 2(\theta_0 + \theta_1 \mathbf{x}_{n,1} - y_n) w_n \mathbf{x}_{n,1}$$

(b) Define the following:

$$\mathbf{W} = \begin{bmatrix} w_1 & \dots & 0 \\ \vdots & \ddots & 0 \\ 0 & \dots & w_N \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} 1 & x_{1,1} \\ 1 & x_{2,1} \\ \vdots & \vdots \\ 1 & x_{N,1} \end{bmatrix}$$

$$\mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}$$

We know that (adding 1/2 for conveniance):

$$\begin{aligned} J(\theta) &= \frac{1}{2}(\mathbf{X}\theta - \mathbf{Y})^T \mathbf{W}(\mathbf{X}\theta - \mathbf{Y}). \\ \frac{\partial}{\partial \theta} J(\theta) &= \frac{1}{2}(\mathbf{X}\theta - \mathbf{Y})^T \mathbf{W}(\mathbf{X}\theta - \mathbf{Y}). \\ &= \frac{1}{2}\frac{\partial}{\partial \theta}(\theta^T \mathbf{X}^T \mathbf{W}\mathbf{X}\theta - \theta^T \mathbf{X}^T \mathbf{W}\mathbf{Y} - \mathbf{Y}^T \mathbf{W}\mathbf{X}\theta + \mathbf{Y}^T \mathbf{W}\mathbf{Y}) \\ &= \mathbf{X}^T \mathbf{W}\mathbf{X}\theta - \mathbf{X}^T \mathbf{W}\mathbf{Y} \end{aligned}$$

To minimize we need $\mathbf{X}^T \mathbf{W}\mathbf{X}\theta - \mathbf{X}^T \mathbf{W}\mathbf{Y} = 0$. This gives us

$$\theta = (\mathbf{X}^T \mathbf{W}\mathbf{X})^{-1} \mathbf{X}^T \mathbf{W}\mathbf{Y}.$$

4 Problem 4

(a) Since data is linearly separable, we know that there exists $\vec{w}$ such that for all $(\vec{x}_i, y_i)$:

$$y_i = \begin{cases} 1, & \vec{w}^T\vec{x}_i + \theta \geq 0 \\ -1, & \vec{w}^T\vec{x}_i + \theta < 0. \end{cases}$$

So, we know that $y_i(\vec{w}^T\vec{x}_i + \theta) > 0$. Let the minimum of all of these be for j and let $y_j(\vec{w}^T\vec{x}_j + \theta) = \eta > 0$. Now pick $\hat{w} = w\eta^{-1}$ and $\hat{\theta} = \theta\eta^{-1}$. Notice that $\hat{w}$ and $\hat{\eta}$ give $\delta = 0$.

(b) If there is an optimal solution $\vec{w}$, θ with $\delta = 0$ that implies that for all $(\vec{x}_i, y_i)$

$$y_i(\vec{w}^T\vec{x}_i + \theta) \geq 1$$

Which implies that, due to the fact that $\vec{w}^T\vec{x}_i + \theta$ and y_i have the same sign, for all $(\vec{x}_i, y_i)$

$$y_i = \begin{cases} 1, & \vec{w}^T\vec{x}_i + \theta \geq 0 \\ -1, & \vec{w}^T\vec{x}_i + \theta < 0. \end{cases}$$

which implies linear separability by definition.

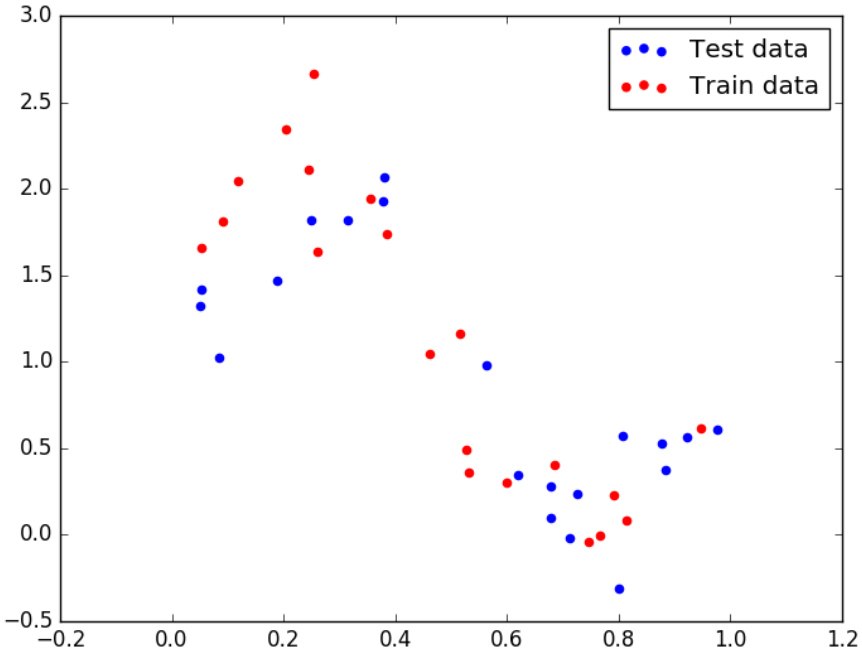
(c) Depends for which value of δ the hyperplane generates. If $\delta < 1$, i.e. $1 - \delta > 0$, the problem boils down to the above case and the data is linearly separable. However, if $\delta > 1$ that means that $1 - \delta < 0$, which means there is no gurantee that $\vec{w}^T\vec{x}_i + \theta$ and y_i will have the same sign, which implies we cannot infer anything about linear separability and that the given plane does not separate the data.

(d) Take $\theta = 0$, $w^T = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$, and we will have $\delta = 0$, which is optimal due to the constraint $\delta \geq 0$.

(e) Take $\theta = 0$, $w^T = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$.

5 Problem 5

(a) The test data seems to intuitively match the best fit polynomial of the training data. The training data seems to be fitted best by a polynomial of degree three. Both the training and test data have a relatively high amount variance, but these two variances are similar to eachother. Particularly, the data seems to deviate from a degree three polynomial with a normal distribution.



(b)

(c)

(d)

Learning rate	Iterations	Coeficients	Train Cost	Test Cost
10^{-2}	764	[2.4464, -2.8163]	3.9126	7.047
10^{-3}	7020	[2.4464, -2.8163]	3.9126	7.047
10^{-4}	10000	[2.27044, -2.4606]	4.086	5.841
0.0407	10000	[-9×10^{18}, -4×10^{18}]	2×10^{39}	2×10^{39}

(e) A closed-form solution in this context means that there is an explicit way to exactly calculate the optimal value (assuming mean-square error) of weights by just multiplying and adding known values.

The train, test costs and the coefficients match exactly to the gradient descent fit where $\eta = 0.01$ (probably not exactly, but close enough for me not to be able to see the difference in python's float representation).

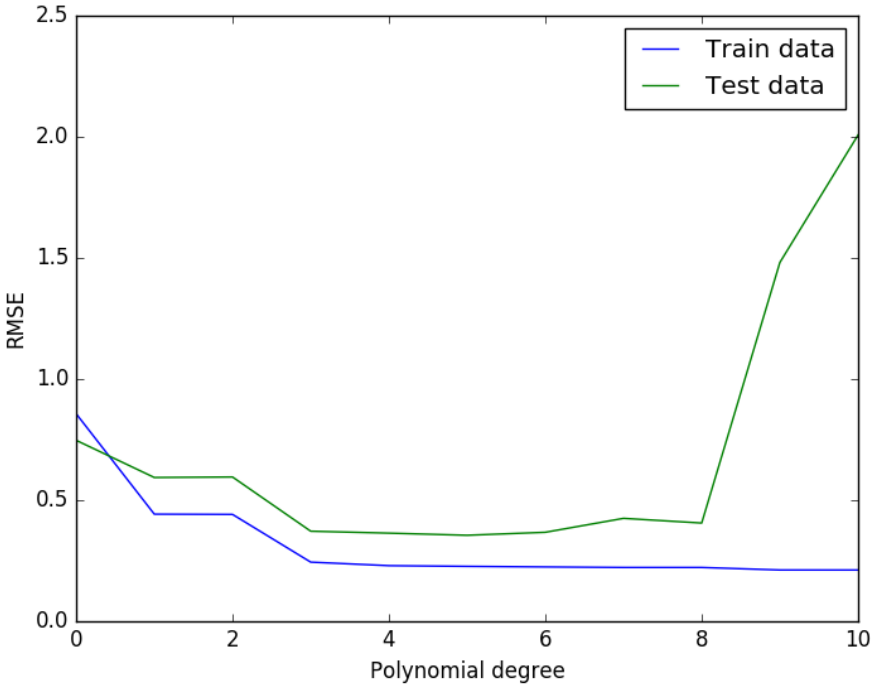
(f)

Iterations	Coeficients	Train Cost	Test Cost
1356	[2.4464, -2.81635]	3.9126	7.047

(g)

(h) RMSE is a better metric here because the error will have the same dimensionality as the actual distance between predicted and true values. Therefore, RMSE will provide a more linear comparison of quality. This means that if RMSE is twice as bad, we can say that the model sort of predicts twice as bad, while saying this for $J(\theta)$ would certainly be incorrect.

(i) The best polynomial is probably of degree 3, since it provides the lowest RMS error, while also being less complex than other models that have similar test data performance (4, 5, 6, 7). There is evidence of overfitting for $m > 8$. Here we see that the training data is being fitted better and better, yet the test data becomes very bad very fast. Also, the data is underfitted for $m = 0$, since the test data performs better than the training data for this case.


Train data
Test data
RMSE
Polynomial degree
0.0
0.5
1.0
1.5
2.0
2.5
0
2
4
6
8
10