

# Leren: Opdracht 2

Semester 1 2014/15

Deadline: 6 November 2014

**MAG IN TEAMS VAN MAXIMAAL 2 GEMAAKT WORDEN**

NB: Als je deze opgave in een team van 2 maakt, lever hem dan 1 keer en maar vermeld duidelijk wie de auteurs zijn.

## 1. *Multipele Lineaire Regressie met de hand*

Doe 1 iteratie van gradient descent voor multipele regressie met onderstaande data. Gebruik een learning rate van 0.1 en initialiseer  $\theta_0$ ,  $\theta_1$  en  $\theta_2$  allemaal met 0.5. Bereken de cost voor en na deze iteratie.

<b>X1</b>	2	4	4
<b>X2</b>	3	5	3
<b>Y</b>	6	6	10

*Dit gaat op dezelfde manier als univariate regressie. We hebben nu drie parameters. De updateregel wordt nu toegepast voor alle parameters. De resultaten zijn:*

Cost voor: 7.7  
gradient  $\theta_0$  -10/3  
gradient  $\theta_1$  -34/3  
gradient  $\theta_2$  -32/3  
nieuwe  $\theta_0$  0.83  
nieuwe  $\theta_1$  1.63  
nieuwe  $\theta_2$  1.57  
cost na 16.1

Kosten worden groter door te grote alfa.

## 2. *Terminatie*

Het gradient descent algorithm heeft een terminatie conditie nodig. We kunnen dit opgeven als (a) het aantal iteraties, of als (b) de maximale grootte van de laatste verandering in parameters of (c) van de laatste verandering in cost. Geef voor en nadelen van deze verschillende mogelijkheden.

*Als het aantal iteraties te klein wordt gekozen dan bereikt hij het minimum niet. Als dit te groot wordt genomen kost het rekentijd. Als we het gaan optimaliseren kost dat tijd. Als de andere twee te groot worden gekozen dan benaderen ze het minimum mogelijk niet goed genoeg (overshoot) en als het te klein is dan kost dit rekentijd omdat het lang kan duren voor de parameter onder de drempel komt, terwijl hij er lang vlakbij heen en weer gaat. Er zit weinig anders op dan wat experimenteren en daarbij het verloop van de error te gebruiken als basis.*

## 3. *Logistische Regressie met de hand*

Doe 1 iteratie van gradient descent voor logistische regressie met onderstaande data. Gebruik een learning rate van 0.1 en initialiseer  $\theta_0$ ,  $\theta_1$  en  $\theta_2$  allemaal met 0.5. Bereken de cost voor en na deze iteratie. Plot de boundary ( $h(x_1, x_2) = 0.5$ ) voor en na de iteratie.

<b>X1</b>	5	5	3	2
<b>X2</b>	3	5	3	4
<b>Y</b>	0	0	1	1

*We krijgen de volgende resultaten:*

Cost voor:	2.52
gradient $\theta_0$	0.5
gradient $\theta_1$	2.5
gradient $\theta_2$	0.2
nieuwe $\theta_0$	0.45
nieuwe $\theta_1$	0.26
nieuwe $\theta_2$	0.31
cost na	1.55

#### 4. Cost Logistische Regressie

Twee maten voor de "cost" van een classifier zijn accuracy, ofwel de proportie goede klassificaties (als we het over kost hebben misschien beter: 1- accuracy) en de cost function die Andrew gebruikt. Voor wat voor data maakt het verschil welke functie gebruikt wordt? Kan je een (kleine) dataset bedenken waarin de class boundary verschillend is voor deze twee maten voor de cost?

*De cost functie die Andrew gebruikt is gevoelig voor het verschil tussen de waarde van  $y$  (de klasse) en de voorspelde waarde. Een groter verschil geeft een grotere fout dan een klein verschil. Daardoor zijn de gradient en de update regel gevoelig voor hoe goed de voorspelling is. Als we alleen naar de accuracy kijken is er geen verschil tussen goede en foute voorspellingen. Dit heeft gevolgen als we extreme datapunten hebben. Een outlier die helemaal verkeerd wordt geklassificeerd zal een grote invloed hebben op de cost. Dit kan de "boundary" zodanig verschuiven dat een voorbeeld net fout wordt geklassificeerd, maar met een kleine "cost", net naast 0.5.*