

# Leren: Programmeeropdracht 1

Week 1 - Semester 1 2014/15

**Deadline: maandag 3 november 12.00 uur**

**Deze opdracht moet individueel gemaakt worden**

Implementeer univariate lineaire regressie met gradient descent in MATLAB. Ontwerp eerst het programma op papier in pseudocode. Bedenk datastructuren voor de trainingsdata. Denk hierbij ook aan  $\theta_0$ . Maak het programma zo dat er een bepaald aantal iteraties gedaan wordt. Dit aantal wordt aan het programma meegegeven net als de learning rate  $\alpha$ . Output van het programma moet de regressiefunctie zijn, dus de waarden van de parameters  $\theta_0$  en  $\theta_1$ . Maak een apart programma dat deze regressiefunctie toepast op data die de gebruiker opgeeft. Gebruik de data van de data van de geschreven opdracht, dus

```
X  5  5  3
Y  6  6 10
```

en de data op de file housesRegr.csv als voorbeeld. HousesRegr.csv bevat de volgende kolommen: MLS = id van de verkoop van een huis, Price = prijs van het huis, Bedrooms = aantal slaapkamers, Bathrooms: aantal badkamers, Size = aantal vierkante voeten (!) oppervlak. Doel is om de prijs te voorspellen uit aantal slaapkamers, aantal badkamers en oppervlak. De data op deze files bevatten meer dan 1 variabele. Pas univariate lineaire regressie met gradient descent toe op de losse variabelen.

Bouw de implementatie op uit de volgende componenten:

1. Inlezen van de data van een file. (Maak zelf een datafile in een handig formaat heeft. Zonodig converteer je andere data met de hand naar dit formaat, bv. met excel of met een editor. )
2. Plotten van de data van (1 variabele!) X en Y; hiervoor kan je bestaande MATLAB code gebruiken
3. Berekenen van de gradient als functie van de parameters. Maak een versie die is geformuleerd in termen van iteraties en een versie in termen van vectoren.
4. Updaten van de parameters
5. Berekenen van de cost  $J(\theta)$ . Maak een versie die is geformuleerd in termen van iteraties en een versie in termen van vectoren.
6. Plotten van de regressie functie; hiervoor kan je bestaande code gebruiken
7. Ga na welke variabele het best voorspelt.
8. Belangrijke details van het algorithm zijn de learning rate en de terminatie. Ga uit van een vast aantal iteraties als basis voor terminatie. Varieer de learning rate en het aantal iteraties om de optimale waarden te vinden voor deze data.

*Inleveren: code voor gradient, cost function, parameter updating, voorspeller, plot van data, plot van regressiefunctie, resultaat van experimenten (waarden van  $\theta$  en cost.*

*Voor deze opdracht zijn 5 punten beschikbaar. Er zijn 8 deelvragen, dus ca. 0.6 punt per deelvraag.*