# Coursera week 1

Monday, 19 November 2018 12:42

**Wat is machine learning**
A computer program is said to learn from experience E with respect to some task T and some performance measure P if its performance on T, as measured by P, improves with experience E

**Supervised** = given dataset en already know the output
**Unsupervised =** approach problems with little or no idea what the result look like

**Regressie** = predict continuous valued output (price)
**Classificatie =** discrete valued output (0 or 1)

T:spam filtering
P:%of spam filtert out
E:e-mail labelled as spam

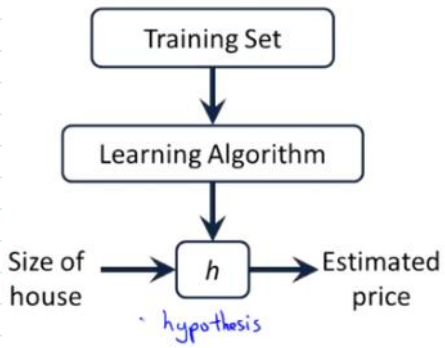
Target variabele = estimate
Independent variabels = features

Lineair regresion = supervised machine learning algorithm

**Naamgeving onderdelen**
Dataset = training set
**m**=number of training examples (aantal rijen)
**x's** = input / features
**y's** = output / target
**(x,y)** = single training example
**(x^i, y^i)**= i aanduiding welk training example er bedoeld wordt

**Hypothesis** $h_\theta(x)$ Kun je op veel verschillende manier beschrijven hieronder staan de meest genoemde in coursera:

- certain function that is the most similar to the dataset
- Hypothesis krijgt input (size house) poept hier de output (prijs) uit.
- $h_\theta$ maps from x's to y's




$$h_\theta(x) = \theta_0 + \theta_1 x$$

*Linear regression with one variable.*
*Univariate linear regression*

$\theta_i's = Parameters$
$x = features\ /\ input$
$h_\theta(x) = Voorspelling\ /output\ /\ target$

**Hoe kies je θ ?**
Choose $\theta_0, \theta_1$ so that $h_\theta(x)$ is close to *y* for our training example (*x*,*y*)

- Hiervoor is de costfunction

**Sum of least squares**

**Cost function:**

- How to fit the best line to our dataset
- Vind de minimale waarde voor $\theta_1, \theta_2$

$$J(\theta_1, \theta_2) = \frac{1}{2\,m} * \sum_{i=1}^{m} (h_\theta(x^i) - y^i)^2$$

**Goal**: minimize $J(\theta_1, \theta_2)$
$\theta_1, \theta_2$

Costfunction = loss / error = Squared error function

**Grandient descent**

- Minimizing costfunction J (op zoek naar het laagste punt)
- Niet alleen linear regressie maar ook bij andere algorihmes
- Bedoelt voor meerdere parameters

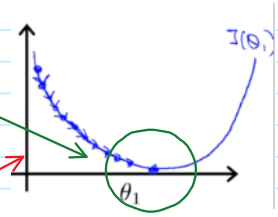
$$\min J(\theta_0, \dots, \theta_n)$$

$$\{\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_n)\}$$

*Repeat until convergence*

:= assignment

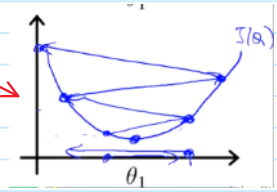
$\alpha$ learning rate




**Learning rate**

- If α is too small, gradient descent can be slow
- If α is too big grandient descent can overshoot minimum (it may fail to converge, or even diverge)

As we approuch a local minimum, gradient descent will automatically take smaller steps. So, no need to decrease α over time




**Matrix**

Data wordt vaak opgeslagen als een matrix

Matrix = m (number of row) * n (number of columns)

Vector = n * 1 matrix

**Optellen/aftrekken**

Alleen als ze de zelfde dementie hebben.

**Vermenigvuldigen**

[m * n matrix] * [n * 1 matrix] = m dimensionale vector

Prediction = datamatrix * parameters

$$\begin{bmatrix} 1 & 0 & 2 \\ -1 & 3 & 1 \end{bmatrix} \cdot \begin{bmatrix} 3 & 1 \\ 2 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1\times3+0\times2+2\times1 & 1\times1+0\times1+2\times0 \\ -1\times3+3\times2+1\times1 & -1\times1+3\times1+1\times0 \end{bmatrix} = \begin{bmatrix} 5 & 1 \\ 4 & 2 \end{bmatrix}$$

**Inverse:**

- Komt altijd 1 uit ( $3(3^{-1})=1$)
- $A\,A^{-1} = A^{-1}\,A = I$
- Matrixen die geen inverse hebben = singulor of degenerate

**Transpose:**

- $A^T$
- De rijen worden kolommen en de kolommen worden rijen

**Indentity matrix**

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

**Cost function vs Gradient decent**

A cost function is something you want to minimize. For example, your cost function might be the sum of squared errors over your training set. Gradient descent is a method for finding the minimum of a function of multiple variables. So you can use gradient descent to minimize your cost function. If your cost is a function of K variables, then the gradient is the length-K

vector that defines the direction in which the cost is increasing most rapidly. So in gradient descent, you follow the negative of the gradient to the point where the cost is a minimum. If someone is talking about gradient descent in a machine learning context, the cost function is probably implied (it is the function to which you are applying the gradient descent algorithm).

Van <https://stackoverflow.com/questions/13623113/can-someone-explain-to-me-the-difference-between-a-cost-function-and-the-gradien>

# Coursera week 2

21 November 2018 14:22

**Gradient descent for multiple variables**

$x_n$ = number of features

$x^{(i)}$=input (features) of the $i^{th}$ training example

**Hypothesis**

- (met enkele variabele) $h_\theta(x) = \theta_0 + \theta_1 x_1$
- (met meerdere variabele) $h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$

*Alle x en θ kunnen als volg geschreven worden:*

$$x = \begin{bmatrix} x_0 \\ \vdots \\ x_n \end{bmatrix} \qquad \theta = \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_n \end{bmatrix}$$

Hierbij is $x_0 = 1$.

*Dit geeft:*

$$h_\theta(x) = \theta^T x$$

**Gradienct Descent for multiple variables**

Gaat op dezelfde als gradienct bij de forige keer alleen hierbij is de costfunction er bijgevoegd:

Repeat
{

$$\theta_j := \theta_j - \alpha \frac{1}{m} * \sum_{i=1}^{m} (h_\theta(x^i) - y^i) x_j^{(i)}$$

}

**Feature selection**

Strategies for feature selection:

- Use correlation coefficients to find dependencies
- Forward selection: add new features
- Backwards elimination: remove feature that improves effectiveness most

Advantages of feature selection:

- Faster training
- Reduces complexity
- Easier to interpret
- Improves accuracy
- Reduces overfitting
- Improves generalization

In general it helps to create more data, and focus less on specific values.

**Feature scaling (data pre-processing)**

Het is belangrijk om ervoor te zorgen dat de features allemaal op dezelfde schaal zijn. Hierdoor is het moeilijker om gradient descent toe te passen. (moeilijk om optimaal minimum te vinden).

Rood = ei = fout

Groen = bol = goed

**Range**

Bij feature scaling is het altijd belangrijk om een $x_n$ waarde in een kleine range te gebruiken. *(hierdoor onstaat ook die mooie bolvormige grafiek)*

- Rule of thumb:

$-3 < x_i < 3$

**Hoe maak je die scaling (mean normalization)**

Door de volgende formule toe te passen:

$$x_i = \frac{x_i - \mu_i}{S_i}$$

$\mu_i = avarage\ value$

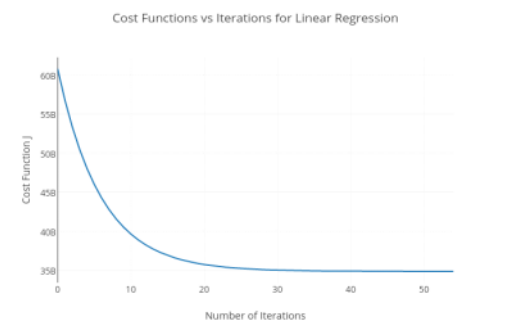
$S_i = Range\ (\max - \min)\ trainingset$

- Wordt ook wel standard diviation genoemd

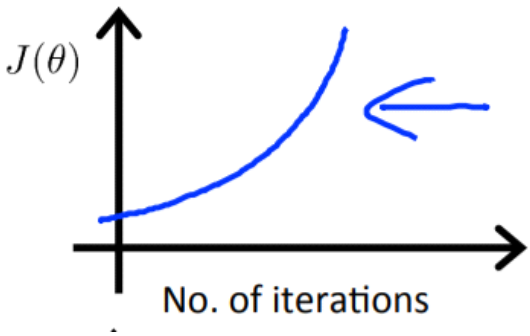
**Gradient descent** (debugging)
Ze gaan hebben het hierweer over gradient descent

## Debugging
-> make sure gradient descent is working -> find θ that minimize J(θ)

**Ideale situatie**
In de afbeelding hieronder is de minimize J(θ) geplot met op de x as het aantal iteraties. Als je hier goed naar kijkt zie je dat J(θ) kleiner wordt na elke iteratie. Het aantal iteraties kan verschillen van 30 tot soms wel 3000000.

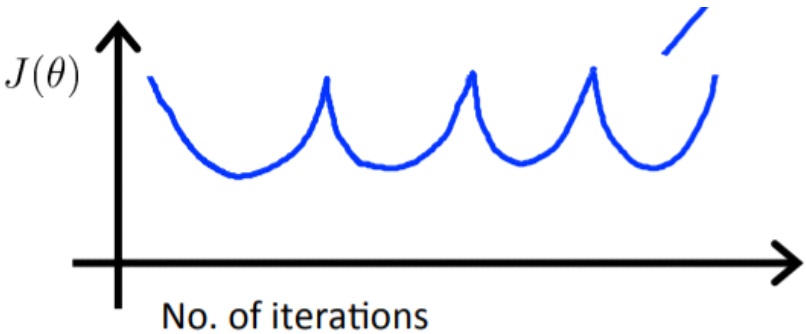



**Realiteit**
Uiteraard kan het voor komen dat er geen grafiek zoals hierboven uitkomt.
De volgende grafieken kunnen er uit komen:




Gradient descent is not working

**α <u>to small</u> = slow convergence**
**α <u>to large</u> = J(θ) may not decrease on iteration**



**Polynomial regression**

Je hoeft niet altijd $x_1$ en $x_2$ features te gebruiken. Je kunt ook zelf verzinnen

## Housing prices prediction

$$h_\theta(x) = \theta_0 + \theta_1 \times \underbrace{frontage}_{x_1} + \theta_2 \times \underbrace{depth}_{x_2}$$


**Normal Equation**

Voor sommige lineare regressie problemen een betere manier om optimale values van $\theta$ te vinden.

$\frac{d}{d\vartheta} J(\vartheta) = 0$ krijg je laagste punt van θ

1. De dataset bestaat uit je verschillende features ($x_1$ size) en je $y$
2. Je volgt er een extra features $x_0$ toe met alleen maar 1en
3. Van de dataset stop je alle features in een matrix X = [alle features] ( *m x (n+1)-dimensional*)
4. Daarna maak je van de y een matrix (*m-dimensional*)
5. Daarna pas je de formule toe

| $x_0$ | Size (feet²) $x_1$ | Number of bedrooms $x_2$ | Number of floors $x_3$ | Age of home (years) $x_4$ | Price ($1000) $y$ |
|---|---|---|---|---|---|
| 1 | 2104 | 5 | 1 | 45 | 460 |
| 1 | 1416 | 3 | 2 | 40 | 232 |
| 1 | 1534 | 3 | 2 | 30 | 315 |
| 1 | 852 | 2 | 1 | 36 | 178 |

$$X = \begin{bmatrix} 1 & 2104 & 5 & 1 & 45 \\ 1 & 1416 & 3 & 2 & 40 \\ 1 & 1534 & 3 & 2 & 30 \\ 1 & 852 & 2 & 1 & 36 \end{bmatrix} \qquad y = \begin{bmatrix} 460 \\ 232 \\ 315 \\ 178 \end{bmatrix}$$

$$\theta = (X^T X)^{-1} X^T y$$

**Gradient descent vs normal equation**

| Grandient descent | Normal equation |
|---|---|
| Need to choose α | No need to choose α |
| Needs many iteration | Dont need iteration |
| Works well even n is large | Need to compute |

| | |
|---|---|
| Needs many iteration | Dont need iteration |
| Works well even n is large | Need to compute |
| | Slow if n is very large (n=10.000) |

# Coursera week 3

24 November 2018 13:49

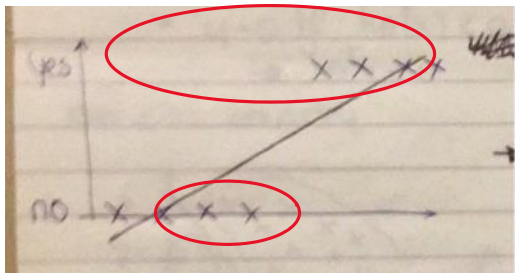
**Classification**
Onderscheid kunnen maken in (meestal) 2 verschillende klassen

0 : "Negative Class"
1: "positive Class"

$y \in \{0,1,2,3\}$

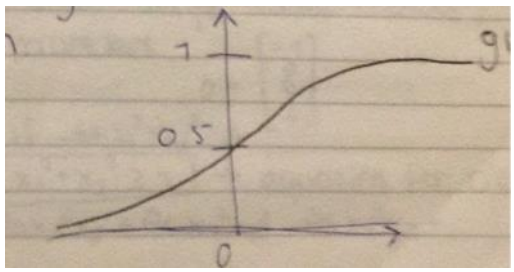
**lineare hypothesis**

$h_\vartheta(x) = \vartheta^T$



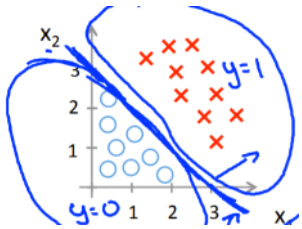

**Sigmoid/logistic function**

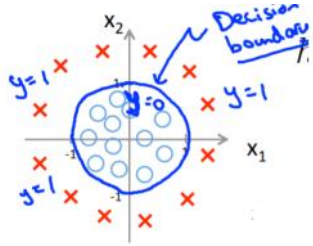
$$h_\vartheta(x) = \frac{1}{1 + e^{-\vartheta^T x}}$$



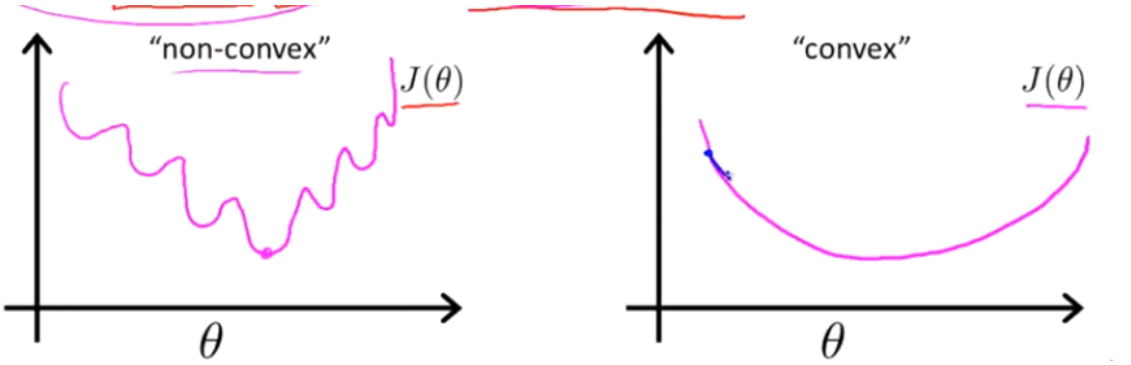

**Decision boundry**
Lijn door data heen. Hierdoor wordt het verdeelt in een 1 en 0 gedeelte

*Non-linear desision boundries*
Door hogere polynomen toe te voegen ontstaat er een boundry. Die niet linear is.



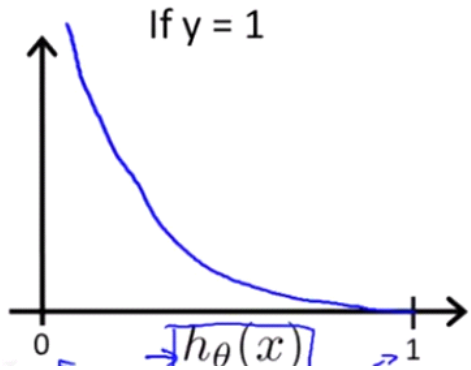

**Cost function**
Je kan de linear regression cost function hier niet op laten lopen want dan wordt het non-convex
Non-convex : hebben veel local optimum. (gradient descent werkt hier niet goed op
Convex:




**Logistic regression cost function**

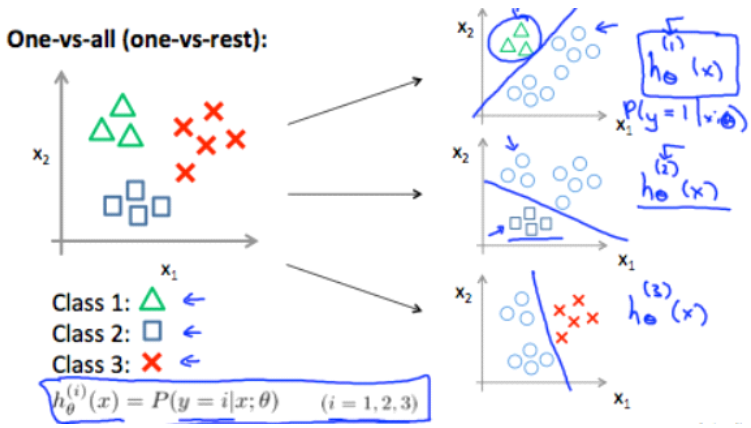
$$\text{Cost}(h_\theta(x), y) = \begin{cases} -\log(h_\theta(x)) & \text{if } y = 1 \\ -\log(1 - h_\theta(x)) & \text{if } y = 0 \end{cases}$$

Als je dit plot krijg je het volgende:

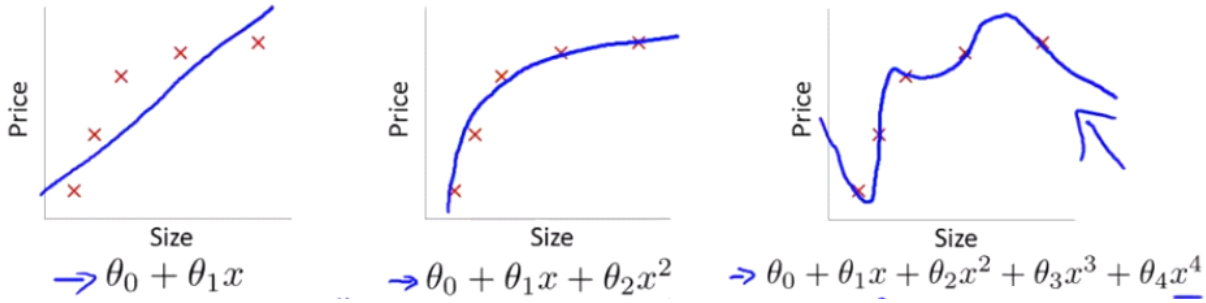



**Lasso regression**

**Multiclass classification: One-vs-all**
In plaatst van y = {0,1} is het nu y={0,1…n}
Wat er dan gebeurt lijkt op wat hierboven is beschreven alleen dan gebeurt dit per class appart.

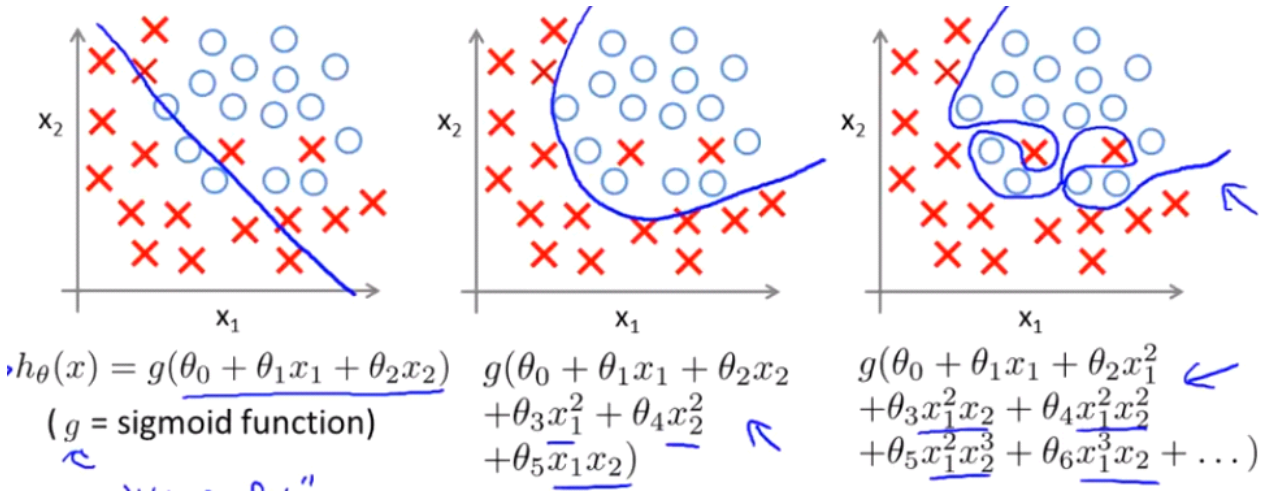



**Overfitting/underfitting**
Hoe de lijn bij de data past
**Linear**




Underfit = high bias | Just right | Overfit = high variance

*Overfitting: if we have to many features, the learned hypothesis may fit the training set very well, but fail to generalize to new examples.*




Underfit = high bias | Just right | Overfit = high variance

**Oplossen van overfitting**

1. Verminderen van het aantal features
2. Regularization (keep futures, but reduce magnitude)

**Regularization**

De techniek die overfitting op lost.

$$\theta_j := \theta_j(1 - \alpha\frac{\lambda}{m}) - \alpha\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$$

λ: maakt de waarde kleiner

# Coursera week 6

22 November 2018 14:31

**Debugging a learning algorihm:**

- Get more training examples
- Try smaller set of features
- Try getting additional features
- Try adding polynomial features
- Try dicreasing λ
- Try increasing λ

$J_{cv} = J_{test}$

**Evaluating your hypothesis** $(x^{(n)}, y^{(n)})$

1. Dataset verdeling Trainingset 70% $(x^n_{test}, y^n_{test})$
   Testset 30%
2. Learn parameter θ met de trainings data
3. Compute test set error:

$$J_{test}(\Theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} (h_\Theta(x^{(i)}_{test}) - y^{(i)}_{test})^2$$

**Evaluating your hypothesis**

Training set error geeft geen goede voorspelling voor de hypothis.

De traingsdata test je op de test set. Vervolgens komt daar geen eerlijke waarde uit bij de Jtest. Daarom moet er cross validation set. (Met waarde die nooit zijn toegevoegd)

**Cross validation** = simple strategy to detect overfitting

d = degree of polynomial -> niet eerlijk als je op test set doet

Kijken polynoom het best geschikt is:

Verdelen dataset in 3 stukken

- Trainings set 60% $(x^{(n)}, y^{(n)})$
- Cross validation 20% $(x^n_{cv}, y^n_{cv})$
- Test set 20% $(x^n_{test}, y^n_{test})$

Training error:

$$J_{train}(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$$

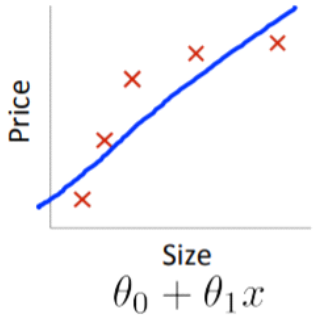
Cross Validation error:

$$J_{cv}(\theta) = \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} (h_\theta(x^{(i)}_{cv}) - y^{(i)}_{cv})^2$$

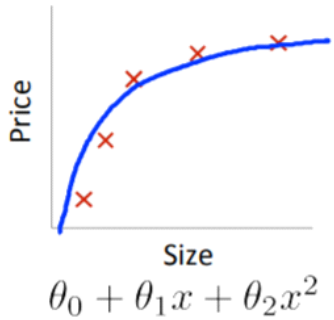
Test error:

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} (h_\theta(x^{(i)}_{test}) - y^{(i)}_{test})^2$$

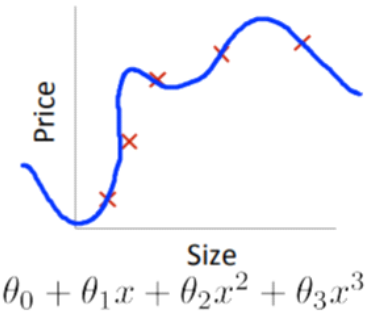
**Diagnosing bias vs. variance**




$\theta_0 + \theta_1 x$




$\theta_0 + \theta_1 x + \theta_2 x^2$




$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

High bias (underfit)

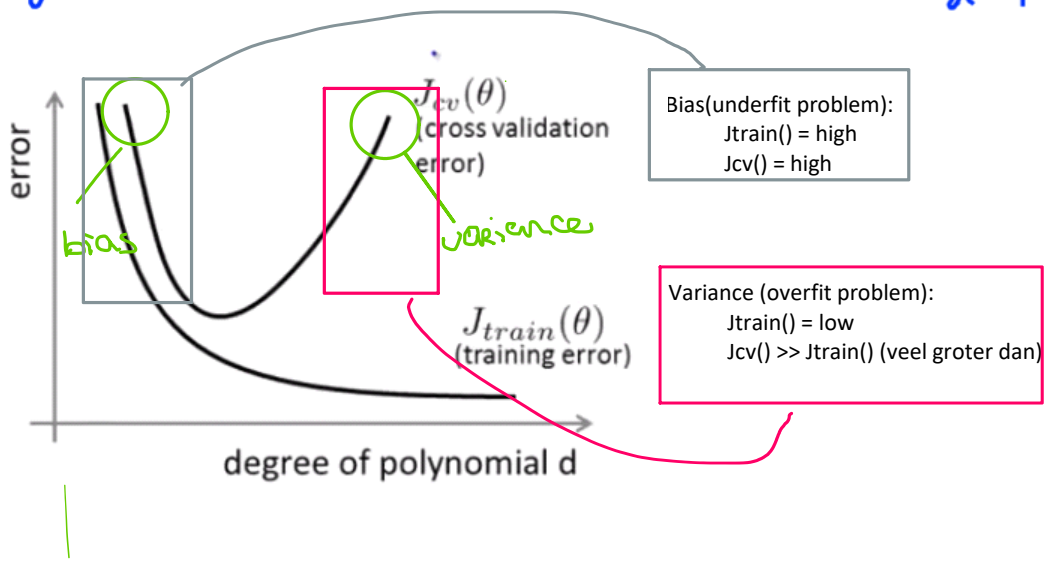
d=1

"Just right"

d=2

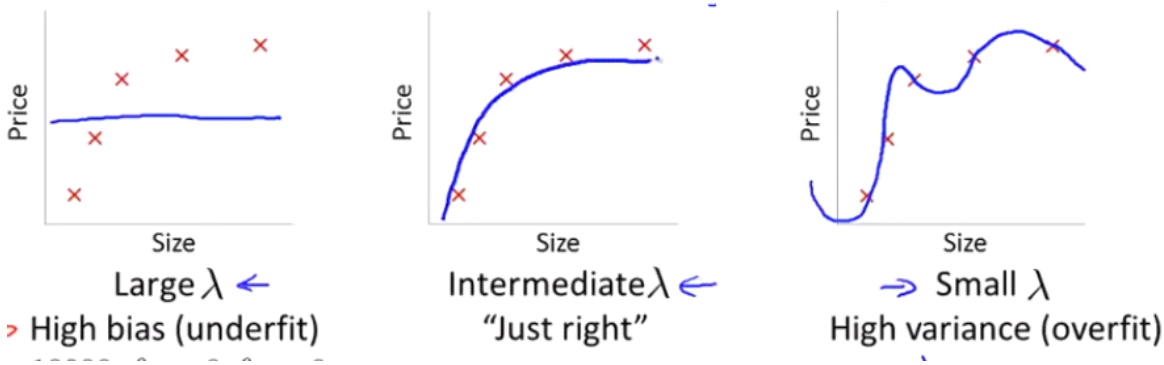
High variance (overfit)

d=4



**Regularization with Bias/variance**

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \boxed{\frac{\lambda}{2m}\sum_{j=1}^{m}\theta_j^2}$$

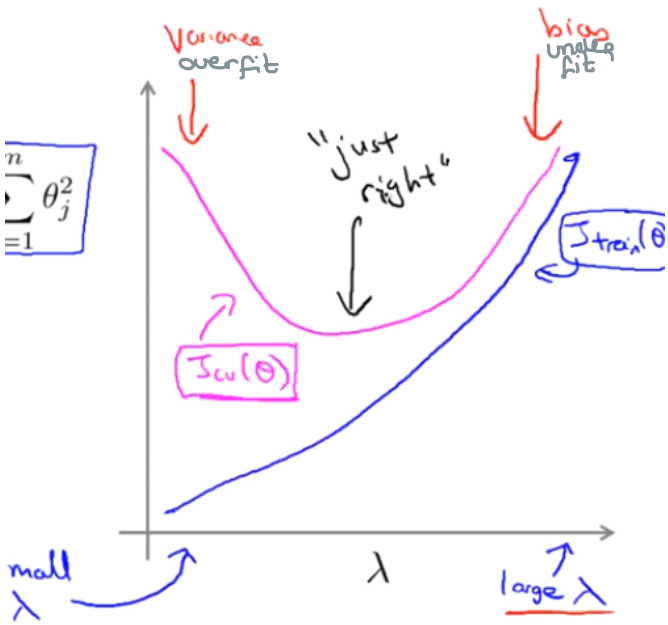



Large $\lambda$ ← High bias (underfit)

Intermediate $\lambda$ ← "Just right"

→ Small $\lambda$ High variance (overfit)

Als je dit uitplot op de garafiek krijg je het volgnde:




Door regularisation te vergroten verlaag je de kans op overfitting en worde die meer underfit

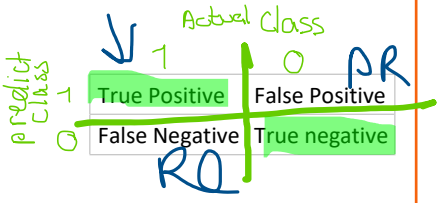
**Debugging a learning algorihm:**

| | |
|---|---|
| Get more training examples | fixes high variance |
| Try smaller set of features | fixes high variance |
| Try getting additional features | fixes high bias |
| Try adding polynomial features | fixes high bias |
| Try decrease λ | fixes high bias |
| Try increasing λ | fixes high variance |

**Neural networks and overfitting**

| | |
|---|---|
| Small NN | Fewer parameters; moreprone to underfitting |
| Large NN | More parameters; more prone to overfit |

Use regularization (λ) to adress overfitting




**Hoe vergelijk je algo met elkaar**
Door het uitrekenen van de f score

$$F_1 score = 2\frac{PR}{P+R}$$

**Precision**
Van iedereen die we voorspelde kanker te hebben had daadwerkelijk kanker Hoe hoger hoe beter

$$\frac{True\ positves}{predicted\ positive} = \frac{True\ positive}{True\ pos + false\ pos}$$

**Recall**
Van iedereen die geen kanker had had uiteindelijk wel kanker Lager is beter

$$\frac{True\ positves}{actual\ positives} = \frac{True\ positive}{True\ pos + False\ neg}$$

Accuracy: the fraction of cases that was classified correctly.

$$correctly = \frac{TP + TN}{N}$$

**Bias/under fit**
Model is oversimplified

Oplossen:
- + features
- +polynomials
- +iterations (if it's not already on a optimum)
- -Regularizarion

**Variance / overfit**
The model is overfitted on the training set. It remembers the different examples given! Instead of a trend lind

Causes of overfitting:
- +features
- -training examples
- - learning poor samples
- -model selection on training set

Oplossen:
- +training examples
- -features
- +regularization
- Early termination

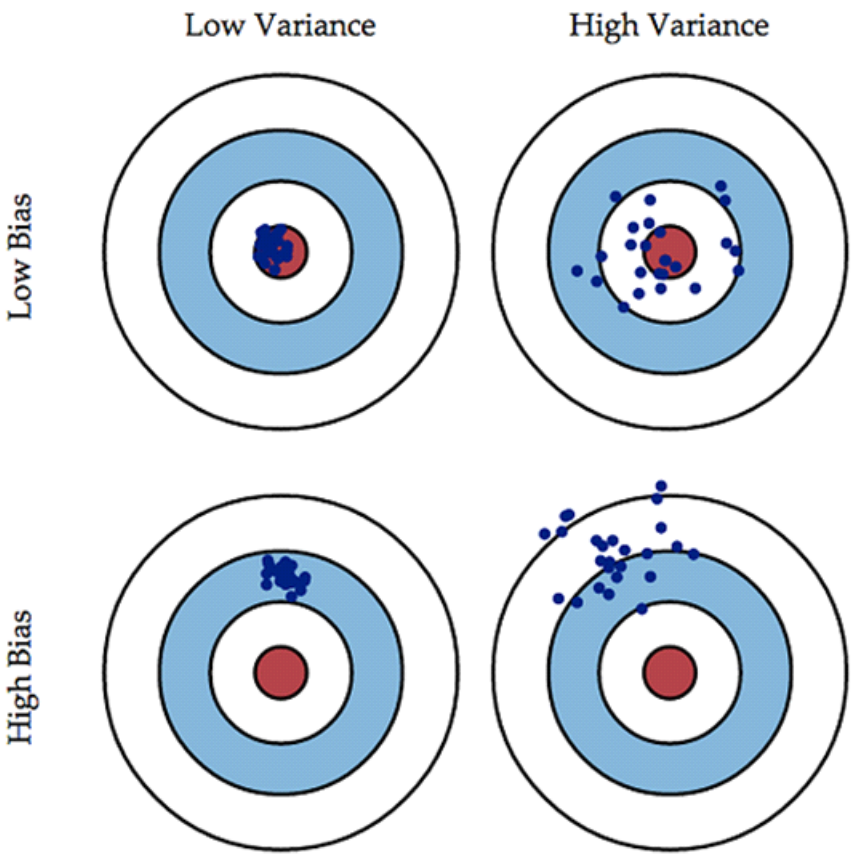
# PowerPoints machine learning

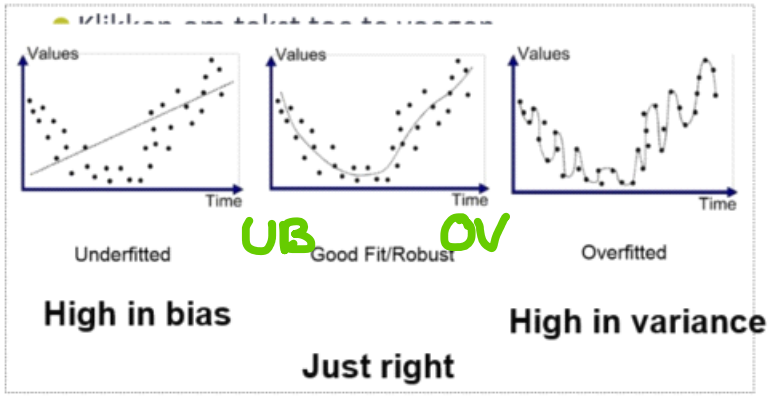
Sunday, 25 November 2018 16:28

**Rules of thumb for sound machine learning**

- Verify data quality
- Verify tussenresultaten
- Measure effectiveness using a ground truth on a hold out test set
- Compare against baseline
- Model selection (the simplest one is the best)

- Regression/continuous scale
  - Mean Absolute Error $\frac{1}{m}\sum_{i=1}^{m}(\hat{y}-y)$
  - Mean Squared Error $\frac{1}{m}\sum_{i=1}^{m}(\hat{y}-y)^2$






## High variance – a.k.a. overfitting

- Causes for overfitting:
  - Too many features (e.g. classify 100MP images)
  - Not enough training data
  - Learning on a poor sample (not representative)
  - Doing model selection on the training set

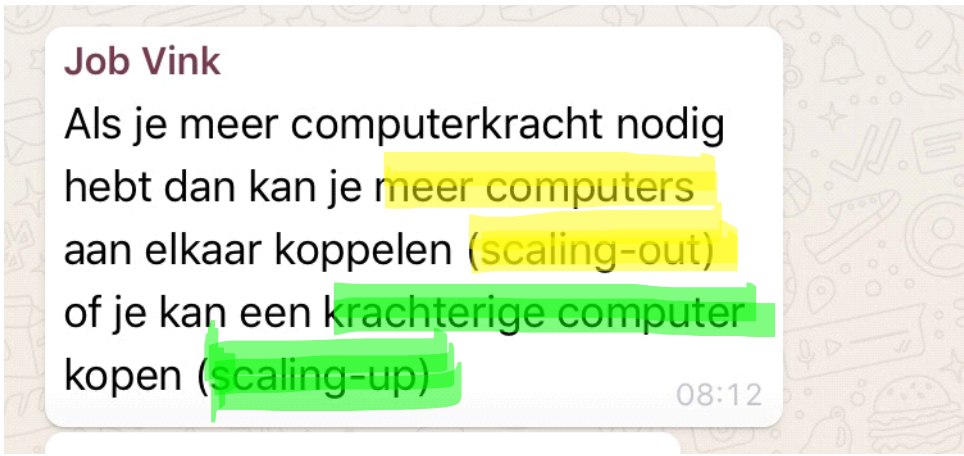
**Regularisation**

Eel penalty toevoegen

- Reduce risk of overfitting

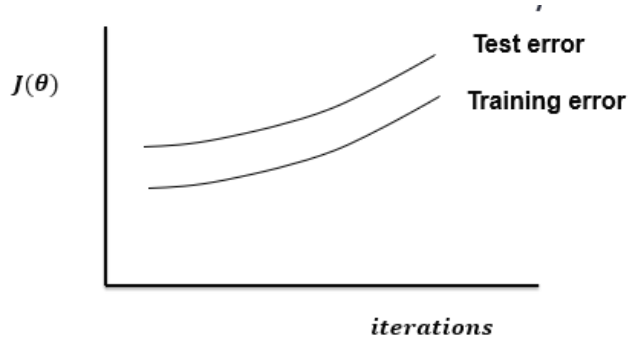
## Feature Engineering

1. Check Data Edges
   a. Check number of rows and columns
   b. Check first few rows and last few rows
   c. Formatting ok? Are the values within realm of reality?
2. Variable Identification (Codebook)
   a. Per set: Where did data come from? How was data collected? Technical information about files. (How many, size, format)
   b. Per variable: Position, name, label, values, data type, numerical/categorical, predictor/target variable, summary statistics.
3. Univariate Analysis
   a. Check if it is a normal distribution
4. Bi-variate
   a. Check the correlations
   b. Plot the data in different graphs to understand the data
5. Missing Values
   a. Find NaN values and delete or replace them
6. Outliers
   a. Find outliers and delete or place them
7. Variable transformation
   a. Mean normalisation
   b. Find a group of outliers and change the scale or multiply everything with a log
8. Variable creation
   a. Change categoricals in numbers
9. Evaluation
   a. Check if the cleaned data has better results than the same model on the raw data. Use Mean Root Squared Error.

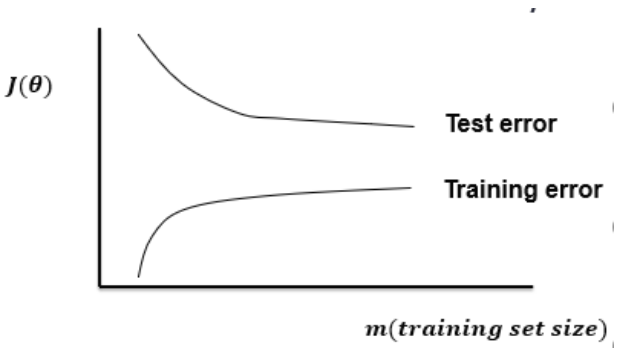

Job Vink
Als je meer computerkracht nodig hebt dan kan je meer computers aan elkaar koppelen (scaling-out) of je kan een krachterige computer kopen (scaling-up)
08:12

# Error Graph

22 November 2018 14:15

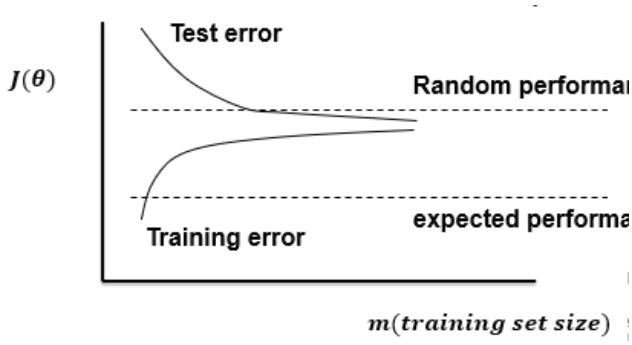



**Fout, learning rate moet lager**



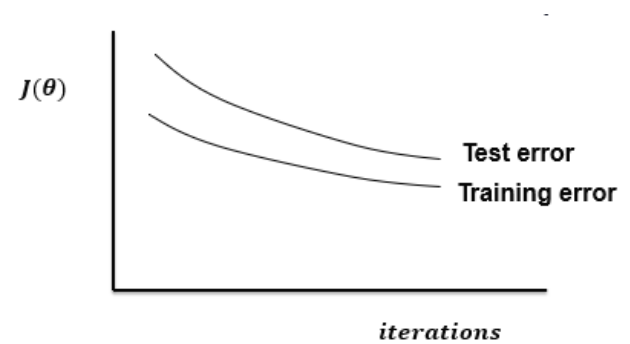

**Fout, overfitting want traning error is veel lager dan test, (dus te veel getraind op training set).**

- **Lambda moet dan hoger.**
- **Polynomials of features minders**
- **Verhoog traing size**

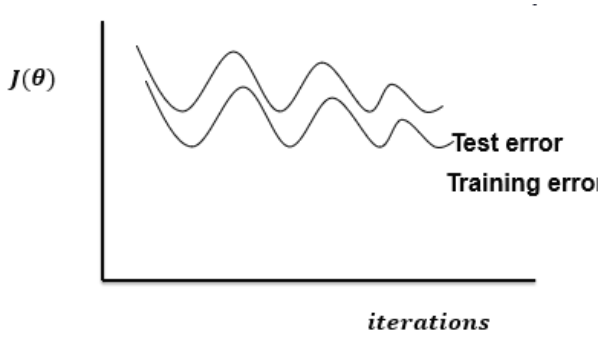



Fout, de lijnen moeten lager komen.
Underfitting, te gegeneraliseerd. (to high in bias)

- Lamba lager maken
- Meer features (wordt specifieker)

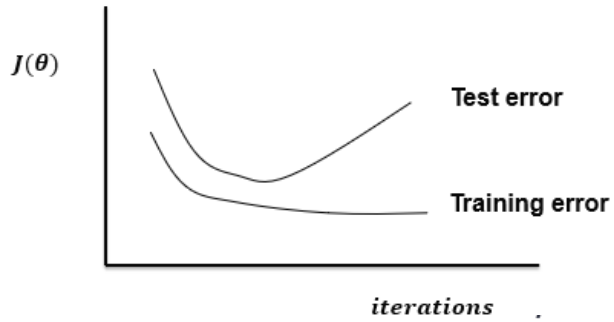



Goed



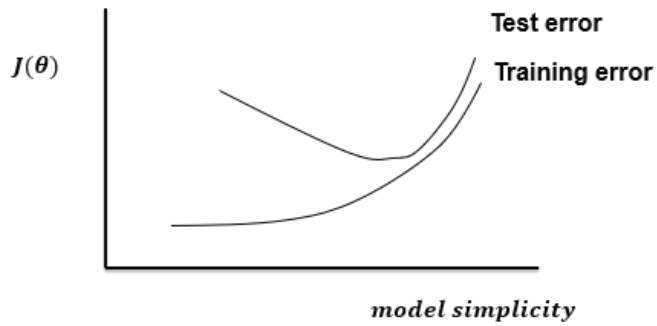

Fout, learning rate is te groot.

- Kleiner maken learning rate




**Fout, overfit.**

- **Get more training examples**
- **Try smaller set of features**
- **Try to increasing regularization**




**Model complexity/simplicity tunen om te voorkomen dat het model high bias of high variance heeft. Dat kan bijvoorbeeld zijn door een juiste feature selection te maken, of een juiste order polynomial.**

# Oefenexamen

Tuesday, 27 November 2018 15:51

Written Test example KB-74

Note: The actual test will contain more questions. The grade will be computed over the number of questions in the test (likely 40 multiple choice questions). To pass, you require half of the questions to be answered correctly, corrected for multiple choice (e.g. for 4-choice questions at least 62.5% of the questions needs to be answered correctly).

Topic Machine Learning

Case: Batteries.com wants to use machine learning to predict the number of sold batteries for laptops. They want to learn a predictive model from collected data.

1. What kind of data will be helpful fort his purpose?

A. sales data of laptops
B. sales data of batteries of laptops
C. Both A and B
D. Neither A nor B

2. What type of problem is this?

A. Regression
B. Classification
C. Ranking
D. All of the above

For learning a model, they initially use a rather small dataset. Therefore, they decide to try out several predictive models that are learned on the entire dataset and compute the RMSE over this dataset to decide what the best model is.

3. This evaluation procedure is wrong, because they can no longer detect whether:

A. the model has overfitted
B. the model is underfitted
C. there is redundancy in the used features
D. they used a correct learning rate

4. If a model is underfit, a possible fix is:

A. use more data
B. use less features
C. lower the learning rate
D. increase the number of iterations over the training data

5. A better strategy to evaluate the effectiveness of a predictive model is:

A. n-fold cross validation
B. measuring effectiveness using Recall instead of RMSE
C. measuring effectiveness using Precision instead of RMSE
D. to throw a dice

6. RMSE stands for:

A. Robust Mean Square Effectiveness
B. Root Mean Square Error
C. Random Mean Square Error
D. Random Mean Square Effectiveness

7. During training, a plot was generated of the cost function (loss) over the number of iterations/epochs (see below).



What is possibly happing here that can explain this plot:

A. Every time the cost function reaches the bottom the model has reach the optimum. It is normal that after reaching the bottom the cost function increases. We can simply go back to the first bottom to find optimal settings.
B. The learning rate is likely too big, and as a result the true optimum will never be found.
C. The model has probably overfitted because it does not converge to a stable optimum.
D. The training set may be too large; as a result the training set may contain new examples that were not learned previously and shift the optimum to a new position.

8. For learning a linear regression model for this case, a standard hypothesis is used:

In this model stands for:

A. the expected number of batteries that is sold.
B. the expected number of laptops that is sold.
C. the learning rate of the model
D. the cost function (a.k.a. loss function)

9. For preparing the data, Spark is being used. The data is supplied in a .csv file and looks as follows:

| Battery_name | 2015 | 2016 | 2017 |
|---|---|---|---|
| Asus K53 | 5.000 | 60.000 | 70.000 |

9. For preparing the data, Spark is being used. The data is supplied in a .csv file and looks as follows:

```
Battery_name 2015   2016   2017
Asus K53     5.000  60.000 70.000
HP P100      60.000 40.000 20.000
...
```

The above file has been read into an RDD named 'battery'. They want to compute the total number of sold batteries per battery (in the above example, for the Asus K53 that would be 135.000). Which of the scripts below does that?

A. battery.take(1+2+3)
B. battery.map(lambda x: (x[0], x[1] + x[2] + x[3]) )
C. battery.filter(lambda x: x[1] + x[2] + x[3] )
D. battery.reduceByKey(lambda x, y: x + y)

10. Spark uses distributed processing. In distributed processing you typically increase the the processing capacity of a computer cluster by:

A. scale-up
B. scale-out
C. scale-in
D. scale-over

11. For research project of shoulder injuries 61 persons have answered a questionnaire. The sample is representative for the population. The answers show that 24% of the males and 20 % of the females suffer from shoulder pain. From these results we draw the conclusion that 'a higher percentage of Males have shoulder pain than woman). To what extend is this a valid conclusion based in these results?

A This is a valid conclusion
B To decide whether this conclusion is valid you will need to test for statistical significance.
C To decide whether this conclusion is valid you need additional information that tells you what could have caused shoulder pain for each of these persons
D You can never draw this conclusion, no matter what information is additionally provided.

12. A problem description for research has to meet certain requirements. Which of the following is not a requirement for a problem description?

A Specifiek
B Meetbaar
C Voorspelbaar
D Tijdgebonden

13. Research papers should distinguish between the 'results' and the 'conclusion'. Why is it necessary to make this distinction?

A Because it should be clear what the factual results are in the one hand and what the researcher's interpretation of those results is on the other hand.
B The conclusion should summarize the results, which is convenient for a reader that wants to scan papers efficiently.
C Because these are two separate things: In the results Section you answer the main research question, and in the conclusion you make recommendations for future work.
D Because this is a standard practice in research papers and it is easiest to just conform to it.

14. "Supervised learning problems are categorized into "regression" and "classifcation" problems."

Given data about the size of houses on the real estate market, try to predict their price.
Price as a function of size is a continuous output, so this is a _____(1)_____ problem.
Given a patient with a tumor, we have to predict whether the tumor is malignant or benign.
This is an example of a _____(2)_____ problem

A Both (1) and (2) are regression problems.
B Both( 1) and (2) are classification problems.
C (1) is a regression problem, (2) a classification problem
D (1) is a classification problem, (2) a regression problem

15. What can you do to make gradient descent run more efficiently?

A mean normalisation.
B feature scaling.
C tune the learning rate α
D all of the above.

https://blackboard.hhs.nl/bbcswebdav/pid-2608376-dt-content-rid-20153544_2/courses/ITD-HMVT17-K74-2018/Test%20Example.docx