Machine Learning Fundamentals – DTSC102

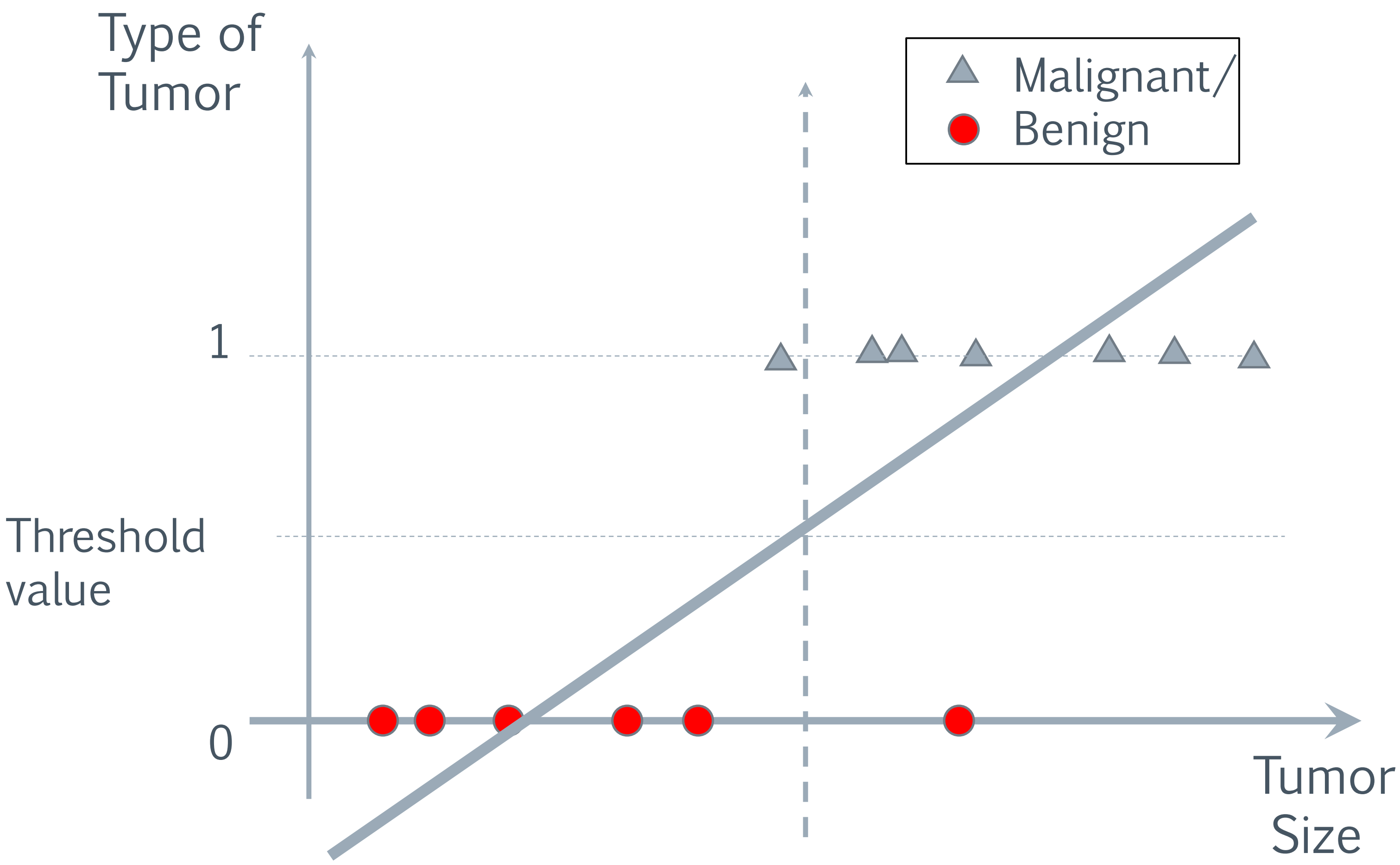
Lecture 6
Logistic Regression

Course Instructor: Dr.-Ing. Maggie Mashaly
maggie.ezzat@guc.edu.eg
C3.220

Contents

Dr.- Ing. Maggie Mashaly

Classification



Classification

- One form of Supervised Learning
- Goal: Learn a mapping from inputs x to outputs y, where y={1,2,…,C}, with C being the number of classes
- Examples:
  - Emails: Spam/Not Spam
  - Online Transactions: Fraudulent (Yes/No)
  - Tumor: Malignant/Benign

$$y = \{0,1\}$$

Negative Class   Positive Class

- Approach: Predict a hypothesis function

$$0 \leq h_\theta(x) \leq 1$$

Dr.- Ing. Maggie Mashaly

Logistic Regression

Approach: Predict a hypothesis function

$$0 \le h_\theta(x) \le 1$$

<u>Solution</u>:

- Use Logistic/Sigmoid Function

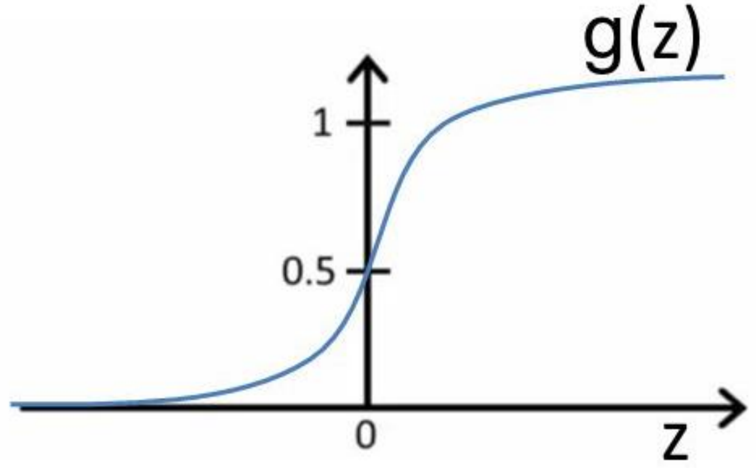
$$\boldsymbol{g(z) = \frac{1}{1+e^{-z}}}$$

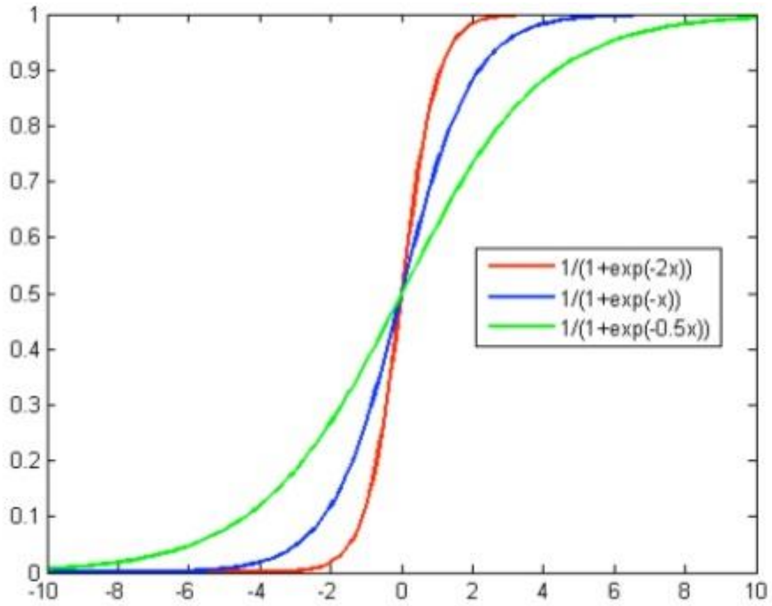
- Define $\boldsymbol{h_\theta(x) = g(\theta^T x)}$

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

- Predict parameters θ to optimize $\boldsymbol{h_\theta(x)}$

But what does $\boldsymbol{h_\theta(x)}$ mean now??





Dr.- Ing. Maggie Mashaly

Logistic Regression - Meaning

- Interpretation of Hypothesis function

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

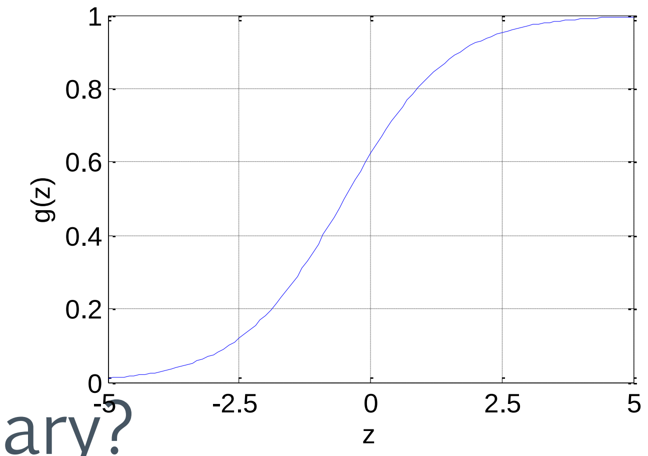
$$\boxed{h_\theta(x) = \text{ estimated probability that } y = 1 \text{ on input x}}$$

<u>Example:</u>

$$h_\theta(x) = p(y = 1|x; \theta) = 0.4$$

Means that the <u>probability that y=1</u>, <u>given x</u>, <u>parameterized by θ</u> is <u>equal to 0.6</u>

- Accordingly, $p(y = 0|x; \theta) = 0.6$
- So the algorithm should predict
  "y=1" if $h_\theta(x) \geq 0.6$ and "y=0" if $h_\theta(x) \leq 0.6$
- Which means also that
  "y=1" if $z \geq 0$and "y=0" if $z \leq 0$



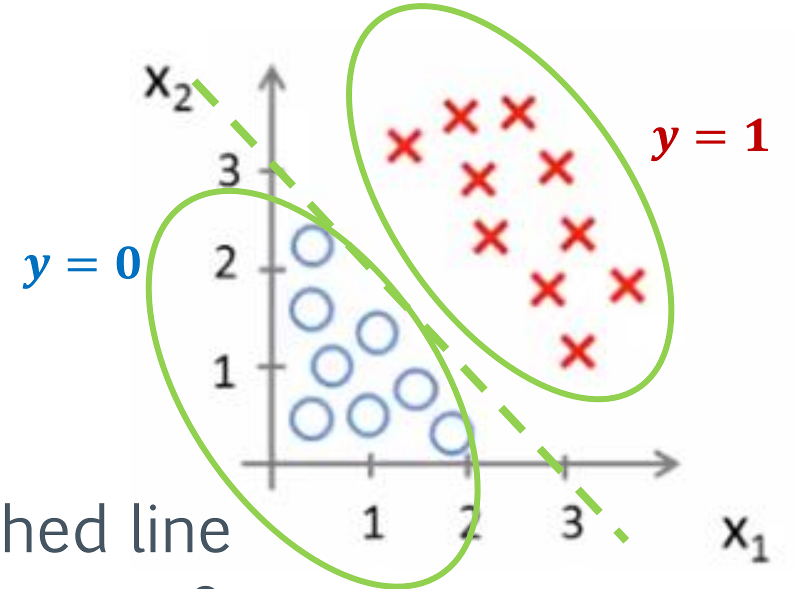
But how can we define this decision boundary?

Dr.- Ing. Maggie Mashaly

Logistic Regression-Decision Boundary

Consider the training set in the graph:

- For all points marked x the output should be $\boldsymbol{y = 1}$
- For all points marked o the output should be $\boldsymbol{y = 0}$



➢ Decision Boundary is shown as the dashed line

- Line equation: $x_1 + x_2 = 3 \;\rightarrow -3 + x_1 + x_2 = 0$

Thus

➢ Predict "$y = 1$" if $\rightarrow -3 + x_1 + x_2 \geq 0 : x_1 + x_2 \geq 3$

➢ Predict "$y = 0$" if $\rightarrow -3 + x_1 + x_2 < 0 : x_1 + x_2 < 3$

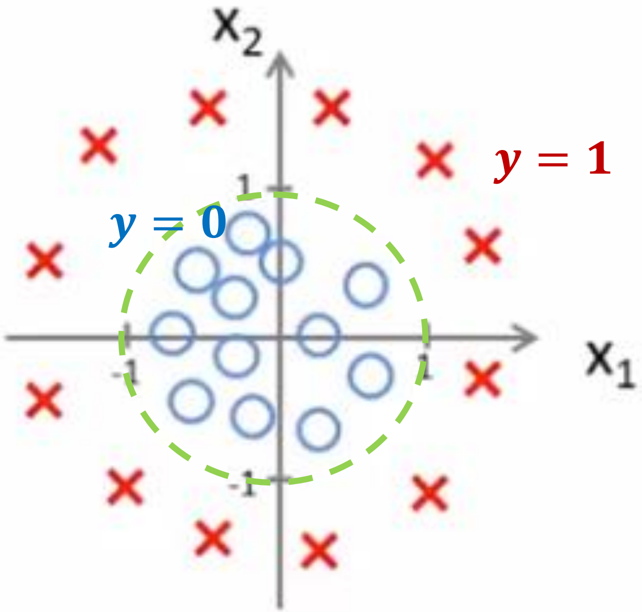
So $\boldsymbol{h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2) \;= g(-3 + x_1 + x_2)}$

Dr.- Ing. Maggie Mashaly

Logistic Regression-Decision Boundary

- Decision Boundaries need not to be always linear, but could be any function that describes any shape that fits our data

- Consider the training set in the graph:



➢We can predict hypothesis function as

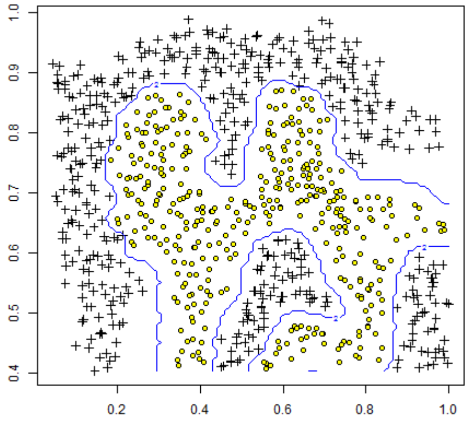
$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1{}^2 + \theta_4 x_2{}^2)$$

➢Using the training data we fit parameters θ

$$\theta^T = [-1 \quad 0 \quad 0 \quad 1 \quad 1]$$

➢Predict "$y = 1$" if → $-1 + x_1{}^2 + x_2{}^2 \geq 0 : x_1 + x_2 \geq 1$

- More complex hypotheses functions can be predicted to present decision boundaries for training data



Dr.- Ing. Maggie Mashaly

Logistic Regression – Cost Function

- Cost function in linear regression

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$

y^i is continuous

Square the error

- Cost function for logistic regression

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} Cost\left(h_\theta\left(x^{(i)}\right), y^{(i)}\right)$$

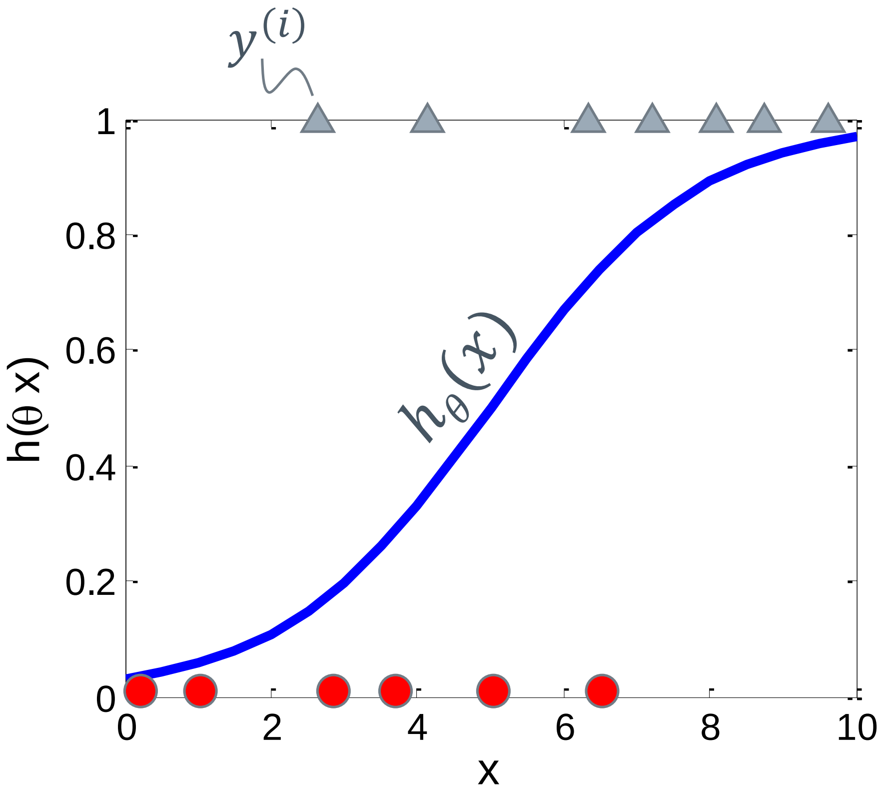
$y = 0$

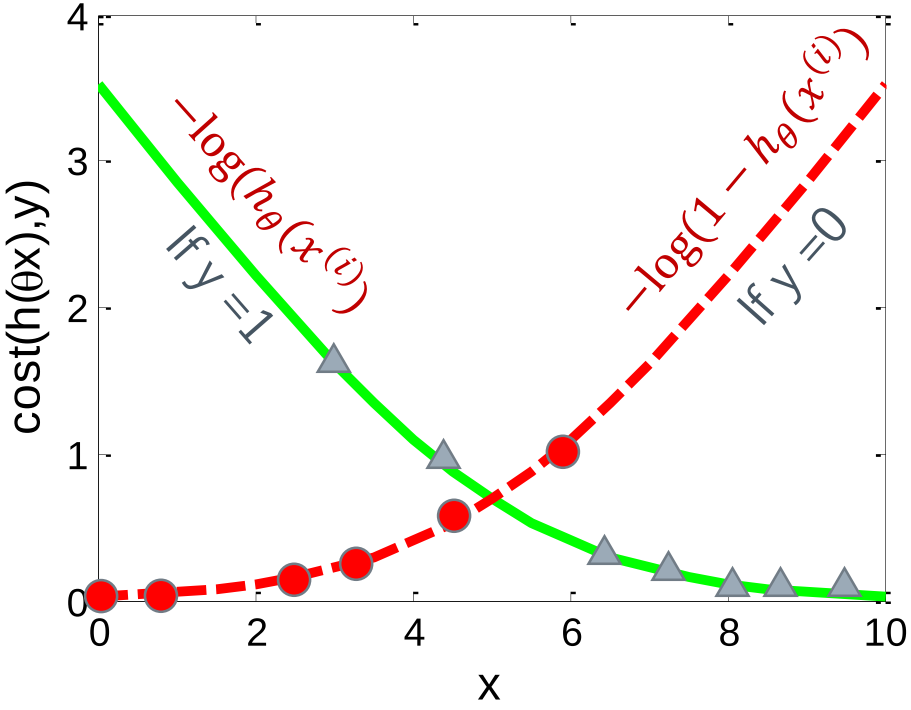
$$\boxed{Cost\left(h_\theta\left(x^{(i)}\right), 1\right) = -\log(h_\theta(x^{(i)})}$$

$y = 1$

$$\boxed{Cost\left(h_\theta\left(x^{(i)}\right), 1\right) = -\log(1 - h_\theta(x^{(i)})}$$

Logistic Regression Cost Function (Example)





Logistic Regression
Combined Cost Function

$Cost\left(h_\theta\left(x^{(i)}\right), 1\right) = -\log(h_\theta(x^{(i)})$

$Cost\left(h_\theta\left(x^{(i)}\right), 0\right) = -\log(1 - h_\theta(x^{(i)})$

$$Cost\left(h_\theta\left(x^{(i)}\right), y^{(i)}\right) = -\mathrm{y}^{(\mathrm{i})}\log(h_\theta(x^{(i)}) - \left(1 - \mathrm{y}^{(\mathrm{i})}\right)\log(1 - h_\theta(x^{(i)})$$

Will only have value when $y^{(i)} = 1$

Will only have value when $y^{(i)} = 0$

Average cost $J(\theta)$ for a given Hypothesis

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} -\mathrm{y}^{(\mathrm{i})}\log(h_\theta(x^{(i)}) - \left(1 - \mathrm{y}^{(\mathrm{i})}\right)\log(1 - h_\theta(x^{(i)})$$

Logistic Regression – Gradient Calculation

$$J(\theta) = \frac{1}{m}\sum_{i=1}^{m} -\mathrm{y}^{(\mathrm{i})}\log(h_\theta(x^{(i)}) - \left(1-\mathrm{y}^{(\mathrm{i})}\right)\log\left(1-h_\theta\left(x^{(i)}\right)\right)$$

$$J(\theta) = \sum \quad \frac{1}{m}\sum_{i=1}^{m} -\mathrm{y}^{(\mathrm{i})}\log(a) - \left(1-\mathrm{y}^{(\mathrm{i})}\right)\log(1-a)$$

$$a = h_\theta\left(x^{(i)}\right) = \frac{1}{1+\mathrm{e}^{-\mathrm{z}}}$$

where $\mathrm{z} = \theta^T x$

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{\partial J(\theta)}{\partial a}\frac{\partial a}{\partial z}\frac{\partial z}{\partial \theta_j}$$

Dr.- Ing. Maggie Mashaly

Logistic Regression – Gradient Calculation

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{\partial J(\theta)}{\partial a}\frac{\partial a}{\partial z}\frac{\partial z}{\partial \theta_j}$$

$$\frac{\partial J(\theta)}{\partial a} = \frac{1}{m}\sum_{i=1}^{m} -\mathrm{y}^{(\mathrm{i})}\frac{1}{a} + \left(1-\mathrm{y}^{(\mathrm{i})}\right)\frac{1}{1-a}$$

$$= \frac{1}{m}\sum_{i=1}^{m}\frac{-\mathrm{y}^{(\mathrm{i})}(1-a) + a\left(1-\mathrm{y}^{(\mathrm{i})}\right)}{a(1-a)}$$

$$= \frac{1}{m}\sum_{i=1}^{m}\frac{\mathrm{a}-\mathrm{y}^{(\mathrm{i})}}{a(1-a)}$$

$$\frac{\partial a}{\partial z} = \frac{-\mathrm{e}^{-\mathrm{z}}}{(1+\mathrm{e}^{-\mathrm{z}})^2}$$

$$= \frac{1}{1+\mathrm{e}^{-\mathrm{z}}}\left(1-\frac{1}{1+\mathrm{e}^{-\mathrm{z}}}\right)$$

$$= \mathrm{a}(1-\mathrm{a})$$

$$\frac{\partial z}{\partial \theta_j} = x_j^{(i)}$$

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\frac{\mathrm{a}-\mathrm{y}^{(\mathrm{i})}}{a(1-a)}\mathrm{a}(1-\mathrm{a})\, x_j^{(i)}$$

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)\, x_j^{(i)}$$

Logistic Regression – Gradient Descent

- Similar to linear regression, gradient descent calculates θ using the following iteration

$$Repeat\ \{\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \quad for\ j{=}0,1,\dots n\}$$

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m} \sum_{i=1}^{m} x_j^{(i)} \left[h_\theta\left(x^{(i)}\right) - \mathrm{y}^{(\mathrm{i})}\right]$$

$$Repeat\ \{\theta_j = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} x_j^{(i)} \left[h_\theta\left(x^{(i)}\right) - \mathrm{y}^{(\mathrm{i})}\right] \quad for\ j = 0,1,\dots n\}$$

$$\Rightarrow \quad \theta = \theta - \frac{\alpha}{m} X^T \left(H_\theta(X) - Y\right)$$

Dr.- Ing. Maggie Mashaly

Logistic Regression Gradient Descent - Example

$$X = \begin{bmatrix} 1 & 2 & 4 \\ 1 & 1 & 7 \\ 1 & 5 & 10 \\ 1 & 6 & 2 \end{bmatrix} \quad Y = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad \theta = \begin{bmatrix} -4 \\ -1 \\ 1 \end{bmatrix} \quad \alpha = 0.3$$

$$H_\theta(X) = \frac{1}{1 + \exp(-X\theta)}$$

$$= \frac{1}{1 + \exp\left(-\begin{bmatrix} 1 & 2 & 4 \\ 1 & 1 & 7 \\ 1 & 5 & 10 \\ 1 & 6 & 2 \end{bmatrix}\begin{bmatrix} -4 \\ -1 \\ 1 \end{bmatrix}\right)}$$

$$H_\theta(X) = \begin{bmatrix} 0.1192 \\ 0.8808 \\ 0.7311 \\ 0.0003 \end{bmatrix}$$

$$\theta_{new} = \begin{bmatrix} -4 \\ -1 \\ 1 \end{bmatrix} - \frac{0.3}{4} \begin{bmatrix} 1 & 2 & 4 \\ 1 & 1 & 7 \\ 1 & 5 & 10 \\ 1 & 6 & 2 \end{bmatrix}^T \left(\begin{bmatrix} 0.1192 \\ 0.8808 \\ 0.7311 \\ 0.0003 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \right)$$

$$\theta_{new} = \begin{bmatrix} -3.9799 \\ -1.1332 \\ 0.7785 \end{bmatrix}$$

$$\theta = \theta - \frac{\alpha}{m} X^T (H_\theta(X) - Y)$$

Machine Learning Fundamentals – DTSC102

Lecture 6
ML Diagnostics

Course Instructor: Dr.-Ing. Maggie Mashaly
maggie.ezzat@guc.edu.eg
C3.220

Contents

- Machine Learning Diagnostics
- Evaluating your Hypothesis
- Model Selection
- Diagnosing Bias vs. Variance

Machine Learning Diagnostic

- Suppose you have implemented linear regression to predict housing prices.

$$J(\theta_0, \theta_1, .., \theta_m) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$

- However, when you test your hypothesis on a new set of houses, you find that it makes unacceptably large errors in its predictions. What should you try next?

➢ Get more training examples

➢ Try smaller sets of features

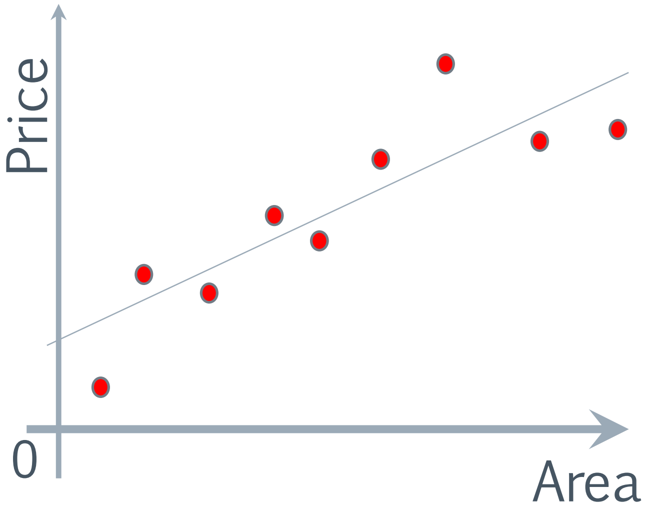
➢ Try getting additional features

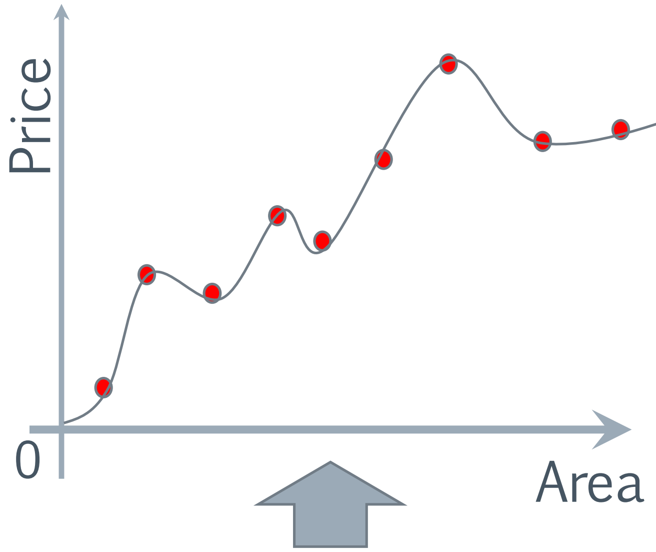
➢ Try adding polynomial features $(x_1^2, x_2^2, x_1 \ x_2)$

➢ Try a different hypothesis

Choosing your Hypothesis

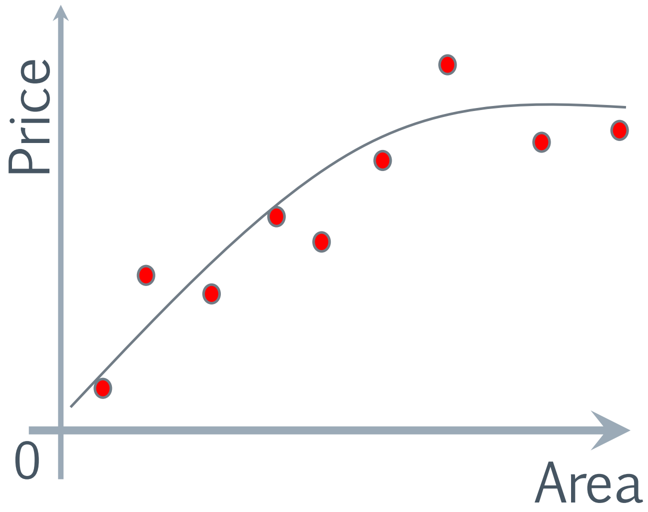
Which one is the best hypothesis ??





This one will present the lowest cost function (lowest error)

But this may not present a general learning



It represents an **Over Fitting** of the data

Over Fitting & Under Fitting

Over Fitting

- Excellent fitting of the training
- Poor ability to predict unlearned data
- Called high variance
- Caused by
  - Very complex Hypothesis
  - Too many features

Under Fitting

- Poor Fitting of the learning data
- Called high bias
- Caused by:
  - Simple hypothesis
  - Too few features
  - Small number of training data

Solution

Model Selection

Regularization

Model Selection

Consider your given dataset:

- Split the data into a **Training Set** (70%) and a **Test Set** (30%)

For **Linear** Regression:

- Learn parameter θ from training data to minimize training error $J(\theta)$
- Compute test set error:

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} \left(h_\theta\left(x_{test}^{(i)}\right) - y_{test}^{(i)}\right)^2$$

Size	Price
2104	400
1600	330
2400	369
1416	232
3000	540
1985	300
1534	315
1427	199
1380	212
1494	243

Training Set Size=m (rows 2104–1534)

Test Set Size=m_{test} (rows 1427–1494)

Dr.- Ing. Maggie Mashaly

Model Selection

For **Logistic** Regression:

- Learn parameter θ from training data to minimize training error $J(\theta)$
- Compute test set error:

$$J_{test}(\theta) = -\frac{1}{m_{test}} \sum_{i=1}^{m_{test}} y_{test}^{(i)} \log h_\theta\left(x_{test}^{(i)}\right) + \left(1 - y_{test}^{(i)}\right) \log h_\theta\left(x_{test}^{(i)}\right)$$

- Alternatively, we can calculate Misclassification Error (0/1 Error)

$$err(h_\theta(x), y) = \begin{cases} 1 \;\; if\, h_\theta(x) \geq 0.5 \; and \; y = 0 \; or \; h_\theta(x) < 0.5 \; and \; y = 1 \\ 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad Otherwise \end{cases}$$

which gives us a binary 0/1 error based on a misclassification, then calculate Average Test error

$$Test\ Error = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} err\left(h_\theta\left(x_{test}^{(i)}\right) - y_{test}^{(i)}\right)$$

Dr.- Ing. Maggie Mashaly

Model Selection

Which polynomial to choose for Hypothesis Function??

1. $h_\theta(x) = \theta_0 + \theta_1 x$ for d=1 : $\theta^{(1)} \longrightarrow J_{test}(\theta^{(1)})$

2. $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$ for d=2 : $\theta^{(2)} \longrightarrow J_{test}(\theta^{(2)})$

3. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3$ for d=3 : $\theta^{(3)} \longrightarrow J_{test}(\theta^{(3)})$

⋮

10. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10}$ for d=10 : $\theta^{(10)} \longrightarrow J_{test}(\theta^{(10)})$

➢Choose degree with the lowest test-set error (ex:$h_\theta(x) = \theta_0 + \cdots + \theta_5 x^5$)

- But does this model generalize well? Not really...

- Why? Because $J_{test}(\theta^{(5)})$ is likely to be an optimistic estimate of generalization error, as d was chosen to fit the test set

Dr.- Ing. Maggie Mashaly

Model Selection

Back to our dataset:

- Split the data into a **Training Set** (60%), a **Cross Validation (CV) Set** (20%) and a **Test Set** (20%)
- Optimize the parameters in θ using the training set for each polynomial degree $J_{train}(\theta)$
- Find the polynomial degree d with the least error using the cross validation set.

$$J_{cv}(\theta) = \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} \left(h_\theta\left(x_{cv}^{(i)}\right) - y_{cv}^{(i)}\right)^2$$

- Estimate the generalization error using the test set $J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} \left(h_\theta\left(x_{test}^{(i)}\right) - y_{test}^{(i)}\right)^2$

	Size	Price
Training Set Size=m	2104	400
	1600	330
	2400	369
	1416	232
	3000	540
	1985	300
Cross Validation Set Size=m_{CV}	1534	315
	1427	199
Test Set Size=m_{test}	1380	212
	1494	243

German University in Cairo

Dr.- Ing. Maggie Mashaly

Model Selection

Which polynomial to choose for Hypothesis Function??

1. $h_\theta(x) = \theta_0 + \theta_1 x$ — for d=1 : $\theta^{(1)} \longrightarrow J_{cv}(\theta^{(1)})$

2. $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2$ — for d=2 : $\theta^{(2)} \longrightarrow J_{cv}(\theta^{(2)})$

3. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_3 x^3$ — for d=3 : $\theta^{(3)} \longrightarrow J_{cv}(\theta^{(3)})$

$\vdots$

10. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10}$ — for d=10 : $\theta^{(10)} \longrightarrow J_{cv}(\theta^{(10)})$

- Choose degree with the lowest cross- validation set error (ex:$h_\theta(x) = \theta_0 + \cdots + \theta_5 x^4$)
- Estimate generalization error for test set $J_{test}(\theta^{(4)})$

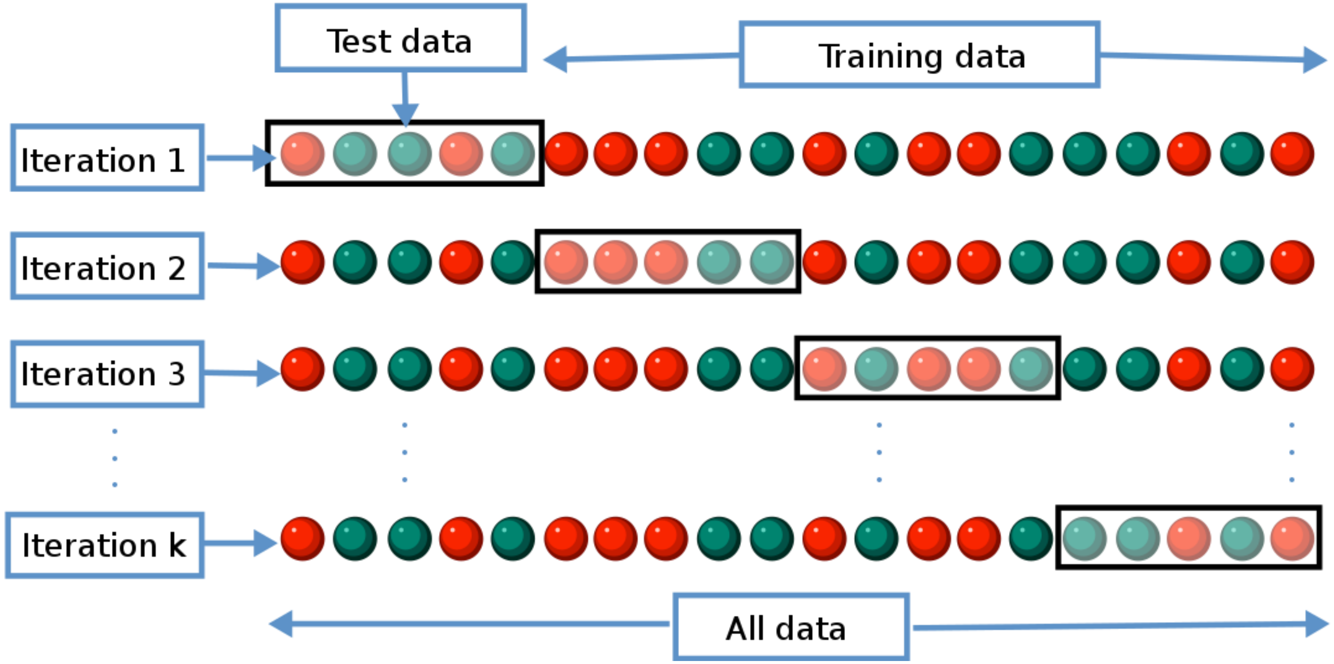
(Thus the degree of the polynomial d has not been trained using the test set)

Model Selection

However there are still limitations for using a single training/CV/test set:

- Original data might not be enough to make a sufficiently large training/CV/test sets
- A single training set doesn't tell us how sensitive accuracy is to a particular training sample

Possible solution: Random Resampling



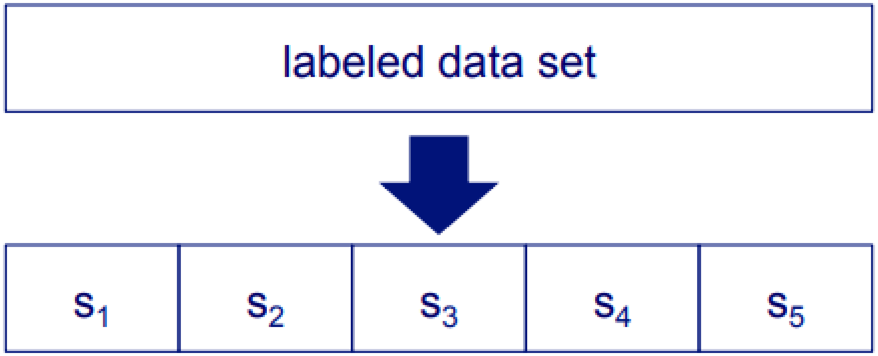
Model Selection

Done more accurately it is called: K-fold Sampling

- ➢ Partition data into n subsamples
- ➢ Iteratively leave one subsample out for the test set, train on the rest

- Suppose we have 100 instances:
  Accuracy=73/100 =73%

- Common value for K is 10, smaller numbers are also used when learning is time consuming

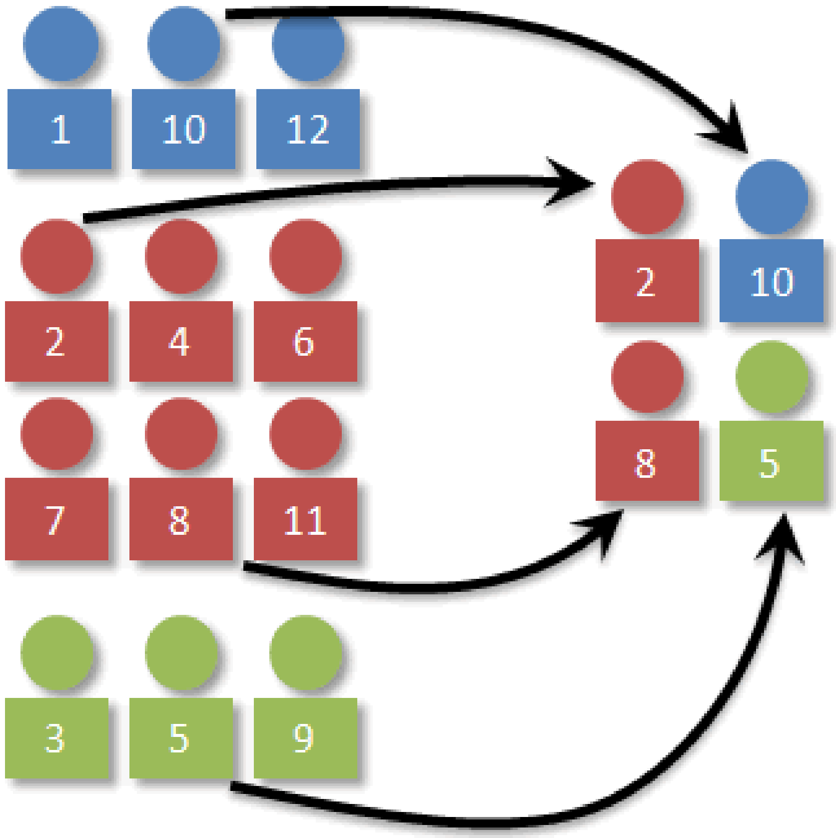


iteration	train on	test on	correct
1	s_2 s_3 s_4 s_5	s_1	11 / 20
2	s_1 s_3 s_4 s_5	s_2	17 / 20
3	s_1 s_2 s_4 s_5	s_3	16 / 20
4	s_1 s_2 s_3 s_5	s_4	13 / 20
5	s_1 s_2 s_3 s_4	s_5	16 / 20

Model Selection

Other possible solution: Stratified Sampling

- Ensures that class proportions are maintained in each selected set
- How it is done:
  1. Stratify instances by class
  2. Randomly select instances from each class proportionally



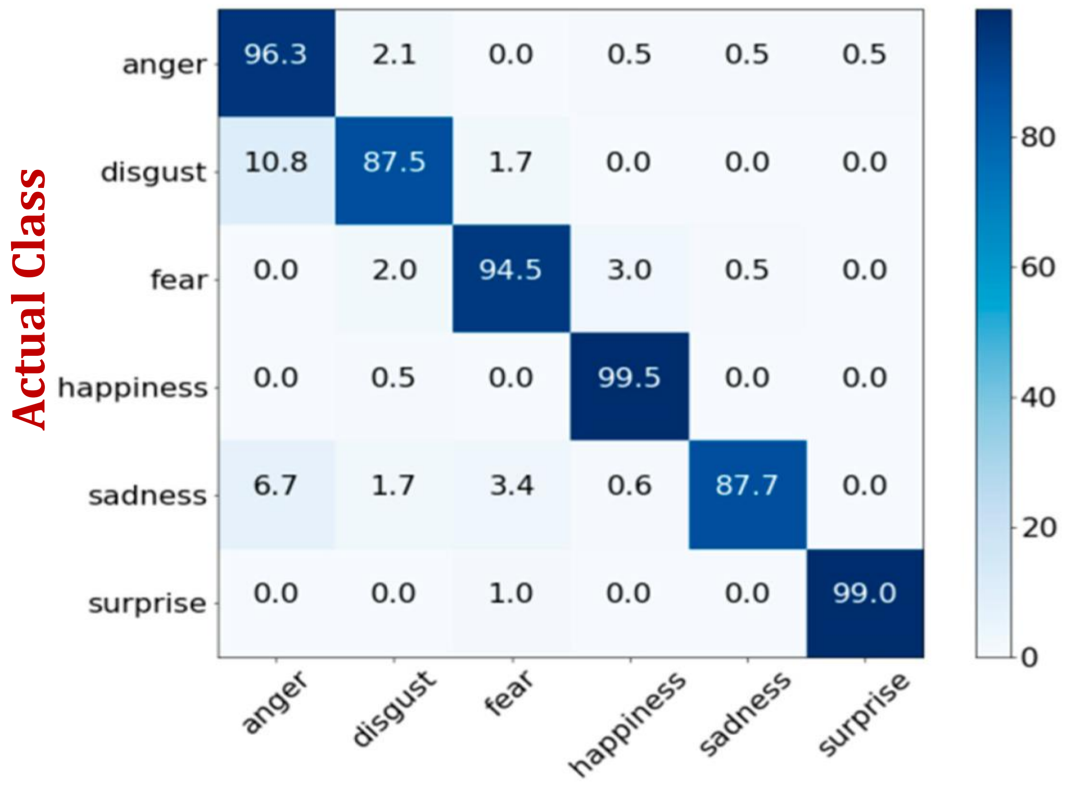
Dr.- Ing. Maggie Mashaly

Model Evaluation

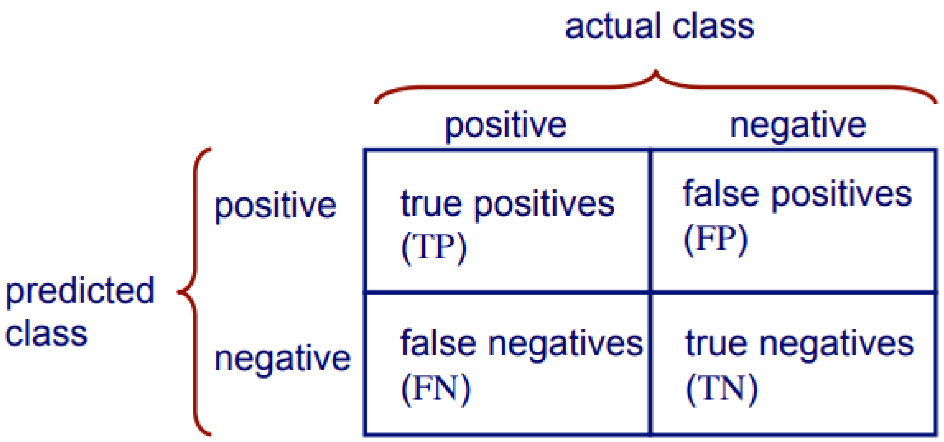
How to understand types of mistakes your model is making?

- Construct a Confusion Matrix

Emotion Recognition from Video



Predicted Class



$$\text{accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

German University in Cairo

Model Evaluation

Confusion Matrix

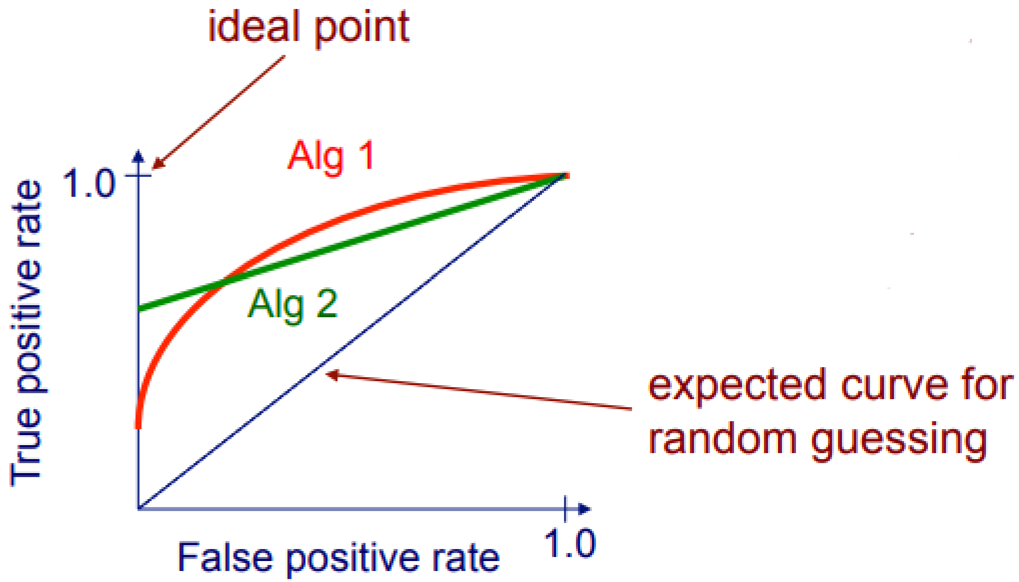
Is the simple accuracy calculation $\frac{TP+TN}{TP+FP+FN+TN}$ sufficient? Not Always…

- Accuracy may not be useful measure in cases where there is a large class skew
  - Is 98% accuracy good if 97% of the instances are negative?
- There are differential misclassification costs; say, getting a positive wrong costs more than getting a negative wrong
  - Consider a medical domain in which a false positive results in an extraneous test but a false negative results in a failure to treat a disease
- We are most interested in a subset of high-confidence predictions

Model Evaluation

Confusion Matrix

- Other Accuracy metrics: TP–rate & FP-rate
- **ROC Curves: "Receiver Operating Characteristics"** curve plots the TP-rate (sensitivity/prob. Of detection) vs. the FP-rate (prob. of false alarm) at various threshold settings



		actual class: positive	actual class: negative
predicted class	positive	true positives (TP)	false positives (FP)
	negative	false negatives (FN)	true negatives (TN)

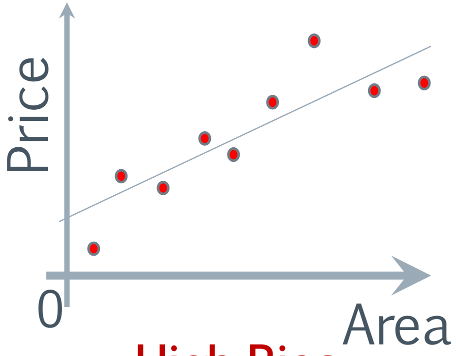
$$\text{true positive rate (recall)} = \frac{\text{TP}}{\text{actual pos}} = \frac{\text{TP}}{\text{TP}+\text{FN}}$$

$$\text{false positive rate} = \frac{\text{FP}}{\text{actual neg}} = \frac{\text{FP}}{\text{TN}+\text{FP}}$$

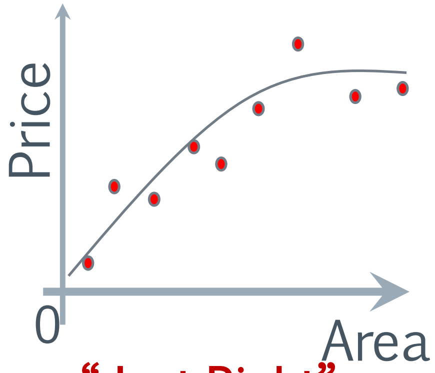
Dr.- Ing. Maggie Mashaly

Model Evaluation
Diagnosing Bias vs. Variance

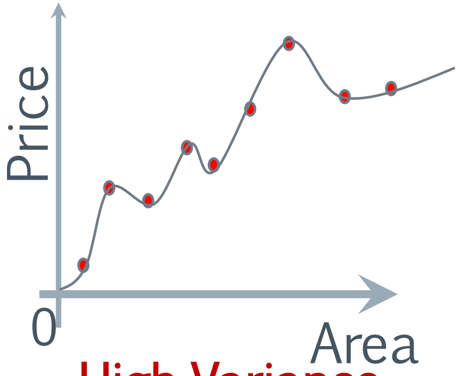
Recall Over/Under fitting hypotheses



High Bias
(Underfit)
d=1



"Just Right"
d=2



High Variance
(Overfit)
d=4

Training Error:

$$J_{Train}(\theta) = \frac{1}{2m_{Train}} \sum_{i=1}^{m_{Train}} \left(h_\theta\left(x_{Train}^{(i)}\right) - y_{Train}^{(i)}\right)^2$$

Cross Validation Error:

$$J_{cv}(\theta) = \frac{1}{2m_{cv}} \sum_{i=1}^{m_{cv}} \left(h_\theta\left(x_{cv}^{(i)}\right) - y_{cv}^{(i)}\right)^2$$

Model Evaluation
Diagnosing Bias vs. Variance

How to diagnose if your learning algorithm is suffering from a Bias/Variance Problem?

Bias (Underfit)

- $J_{Train}(\theta)$ will be high
- $J_{CV}(\theta) \approx J_{Train}(\theta)$

Variance (Overfit)

- $J_{Train}(\theta)$ will be low
- $J_{CV}(\theta) \gg J_{Train}(\theta)$