

Classification using Logistic Regression

Ingmar Schuster

Patrick Jähnichen

using slides by Andrew Ng

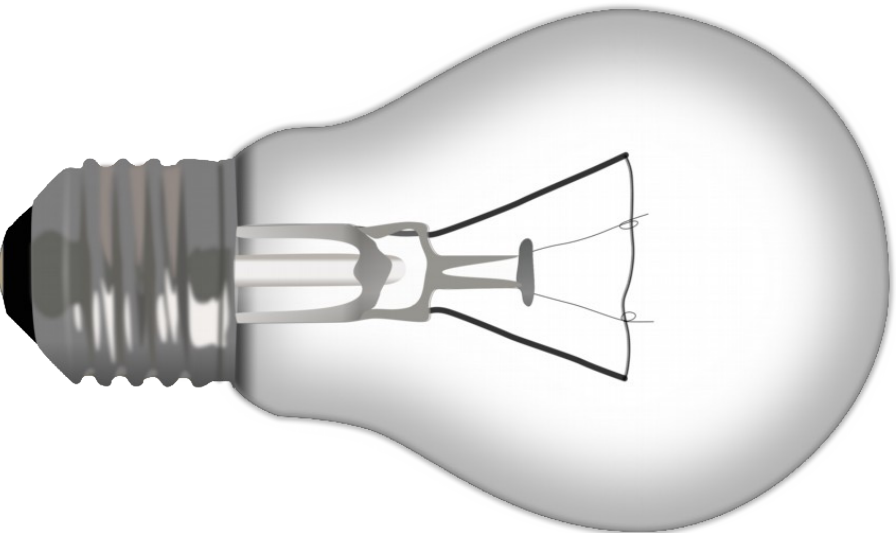
UNIVERSITÄT LEIPZIG

Institut für Informatik



Automatische Sprachverarbeitung

This lecture covers



- **Logistic regression hypothesis**
- **Decision Boundary**
- **Cost function
(why we need a new one)**
- **Simplified Cost function & Gradient Descent**
- **Advanced Optimization Algorithms**
- **Multiclass classification**

Logistic regression

Hypothesis Representation

Classification Problems

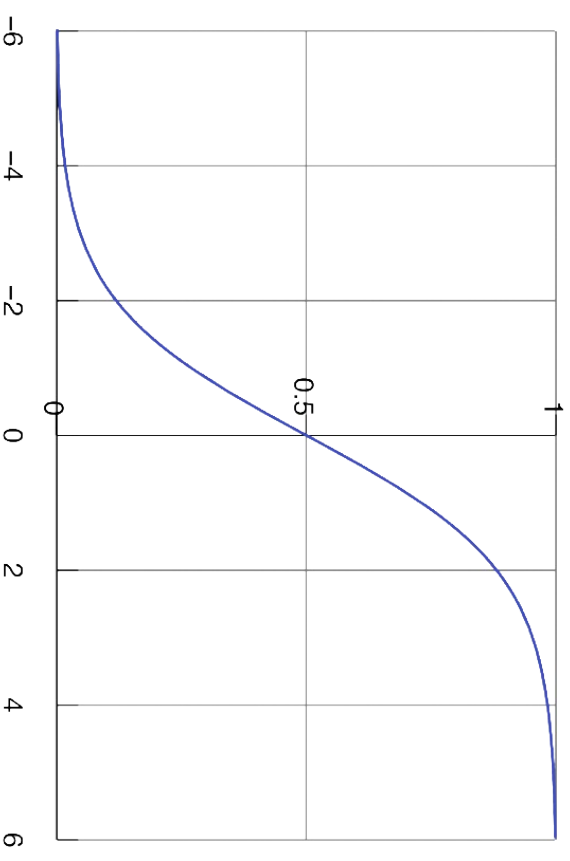
- **Classification**
 - malignant or benign cancer
 - Spam or Ham
 - Human face or no human face
 - Positive Sentiment?
- **Binary Decision Task**
(in most simple case)
 - Want $0 \leq h_{\theta}(x) \leq 1$
 - Data point belongs to class if close to 1
 - Doesn't belong to class if close to 0



Logistic Function (Sigmoid Function)

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

- **maps \mathbb{R} into interval $[0;1]$**
- **0 asymptote for $x \rightarrow -\infty$**
- **1 asymptote for $x \rightarrow \infty$**



Sigmoid Function (S-shape)

Logistic Function

- **Hypothesis** $h_{\theta}(x) = g(\theta^T x)$

$$= \frac{1}{1 + e^{-\theta^T x}}$$

- **Interpretation**

$$h_{\theta}(x) = p(y = 1 | x, \theta)$$

- **Because probabilities should sum to 1, define**

$$p(y = 0 | x, \theta) := 1 - p(y = 1 | x, \theta)$$

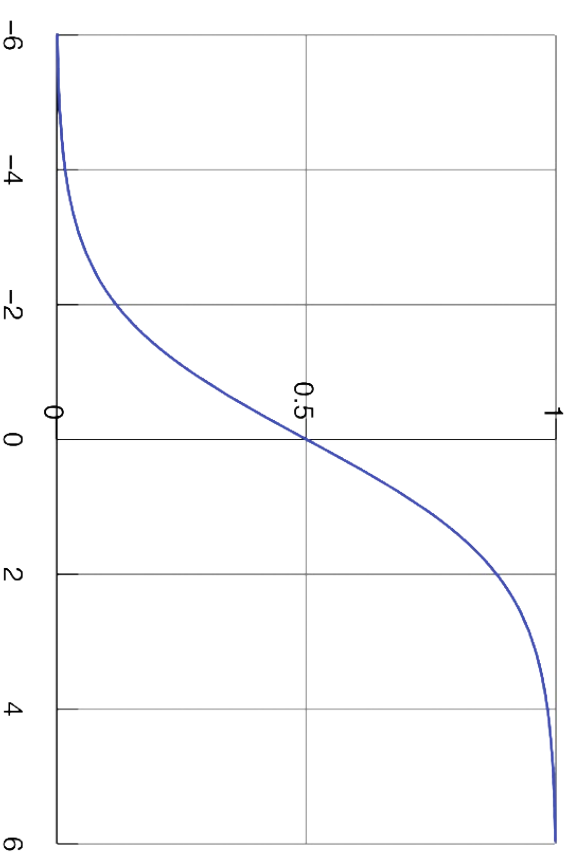
- **If $h_{\theta}(x) = 0.7$ interpret as 70% chance data point belongs to class**
- **If $h_{\theta}(x) \geq 0.5$ classify as *positive sentiment, malignant tumor, ...***

Logistic regression

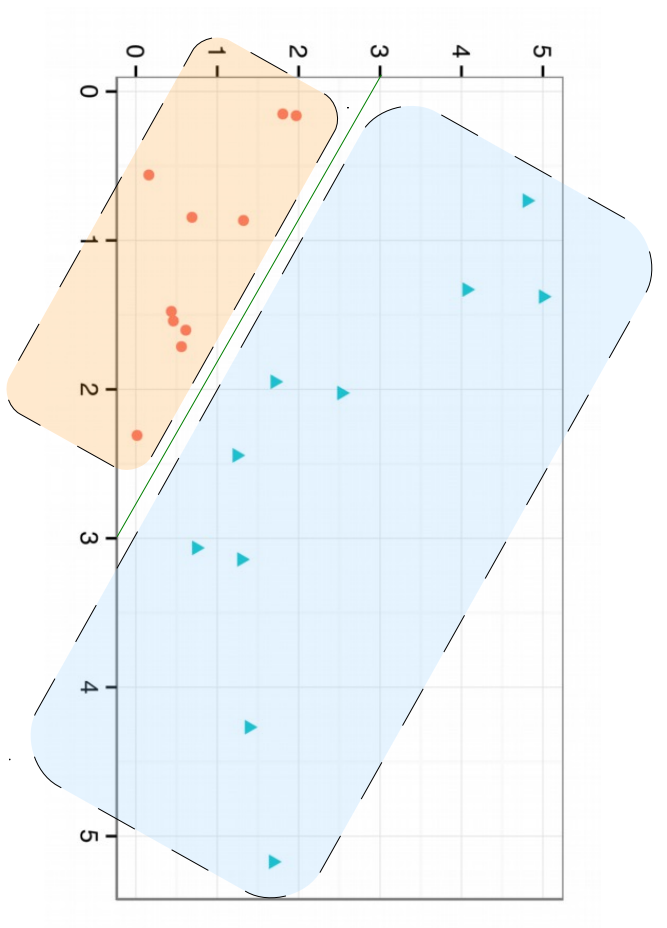
Decision boundary

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

- If $h_{\theta}(x) \geq 0.5$
or equivalently $\theta^T x \geq 0$
predict $y = 1$
- If $h_{\theta}(x) < 0.5$
or equivalently $\theta^T x < 0$
predict $y = 0$



Example



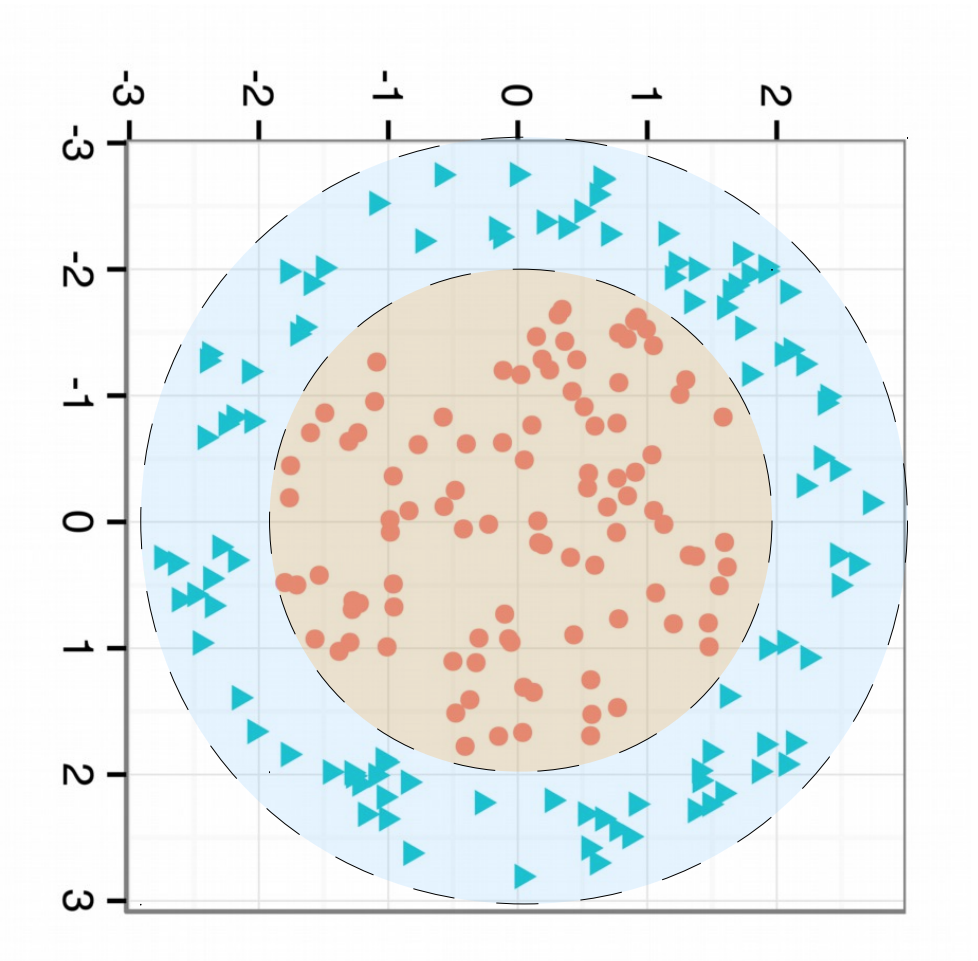
- **If** $h_{\theta}(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$

and $\theta = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$

- **Prediction $y = 1$ whenever**

$$\begin{aligned}
 \theta^T x &\geq 0 \\
 \Leftrightarrow -3 + x_1 + x_2 &\geq 0 \\
 \Leftrightarrow x_1 + x_2 &\geq 3
 \end{aligned}$$

Example



- **If**

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

and

$$\theta = [-2 \quad 0 \quad 0 \quad 1 \quad 1]^T$$

- **Prediction $y = 1$ whenever**

$$x_1^2 + x_2^2 \geq 2$$

Logistic regression

Cost Function

Training and cost function

- **Training data with m datapoints, n features**

$$\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)})\}$$

where

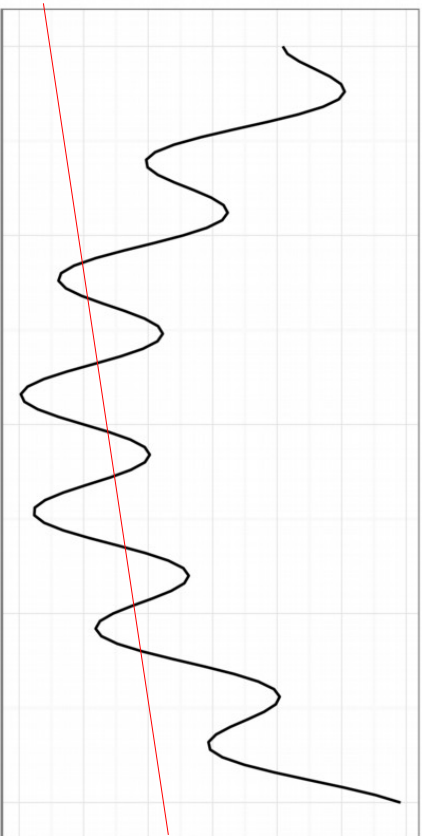
$$x \in \mathbb{R}^{n+1} \text{ with } x_0 := 1, y \in \{0, 1\}$$

- **Average cost**

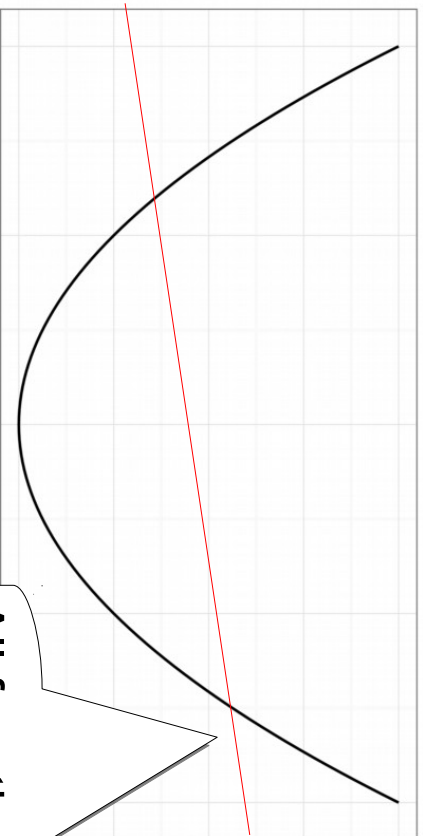
$$J(\theta) = \frac{1}{m} \sum_{i=1}^m \text{Cost}(h_{\theta}(x^{(i)}), y^{(i)})$$

Reusing Linear Regression cost

Nonconvex function



Convex function



- **Cost from linear regression**

$$Cost(h_{\theta}(x), y) := \frac{1}{2} (h_{\theta}(x) - y)^2$$

with logistic regression hypothesis

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

leads to non-convex average cost

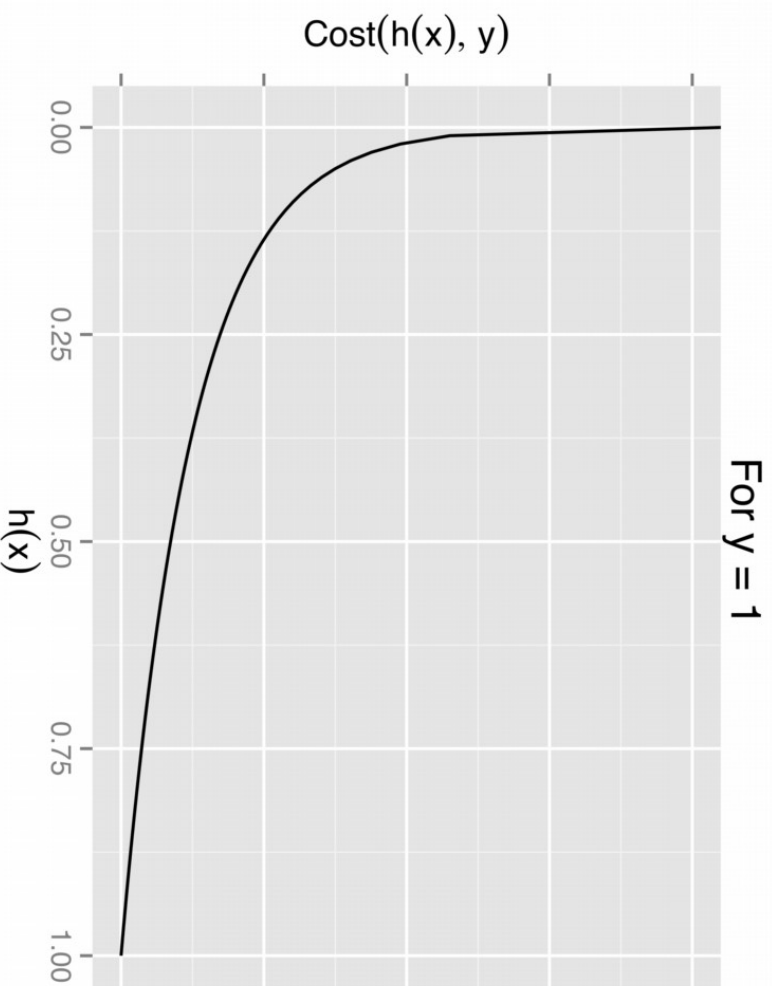
$$J(\theta) = \frac{1}{m} \sum_{i=1}^m Cost(h_{\theta}(x^{(i)}), y^{(i)})$$

- **Convex J easier to optimize (no local optima)**

All function values below intersection with *any* line

Logistic Regression Cost function

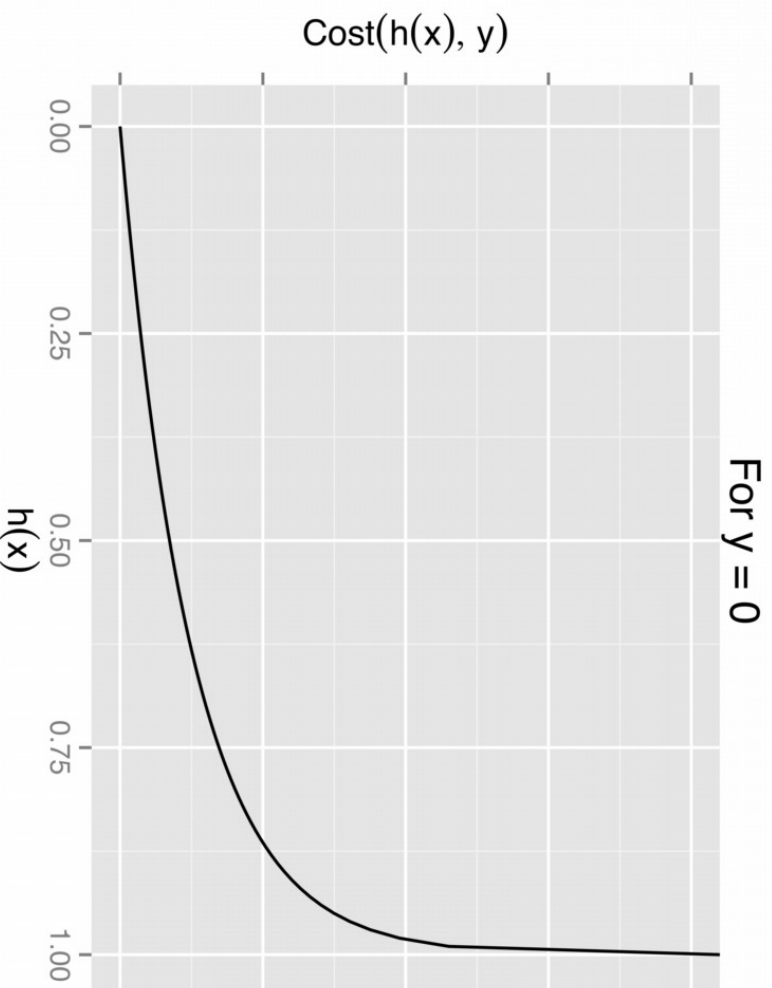
$$Cost(h_{\theta}(x), y) = \begin{cases} -\log(h_{\theta}(x)) & \text{if } y = 1 \\ -\log(1 - h_{\theta}(x)) & \text{if } y = 0 \end{cases}$$



- If $y = 1$ and $h(x) = 1$, $Cost = 0$
- But for $h(x) \rightarrow 0$
 $Cost \rightarrow \infty$
- Corresponds to intuition:
if prediction is $h(x) = 0$ but
actual value was $y = 1$,
learning algorithm will be
penalized by large cost

Logistic Regression Cost function

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



- If $y = 0$ **and** $h(x) = 0$, $Cost = 0$
- **But for** $h(x) \rightarrow 1$
 $Cost \rightarrow \infty$

Logistic regression Simplified Cost Function & Gradient Descent

Simplified Cost Function (1)

- **Original cost of single training example**

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

- **Because we always have $y = 0$ or $y = 1$ we can simplify the cost function definition to**

$$Cost(h_{\theta}(x), y) = -y \log(h_{\theta}(x)) - (1 - y) \log(1 - h_{\theta}(x))$$

- **To convince yourself, use the simplified cost function to calculate**

$$\begin{aligned} Cost(h_{\theta}(x), 1) &= -\log(h_{\theta}(x)) \\ Cost(h_{\theta}(x), 0) &= -\log(1 - h_{\theta}(x)) \end{aligned}$$

Simplified Cost Function (2)

- **Cost function for training set**

$$\begin{aligned}
 J(\theta) &= \frac{1}{m} \sum_{i=1}^m \text{Cost}(h_{\theta}(x^{(i)}), y^{(i)}) \\
 &= -\frac{1}{m} \left(\sum_{i=1}^m y^{(i)} \log(h_{\theta}(x^{(i)})) + (1 - y^{(i)}) \log(1 - h_{\theta}(x^{(i)})) \right)
 \end{aligned}$$

- **Find parameter argument θ' that minimizes J : $\underset{\theta}{\operatorname{argmin}} J(\theta)$**
- **To make predictions given new x output**

$$\begin{aligned}
 h_{\theta'}(x) &= \frac{1}{1 + e^{-\theta'^T x}} \\
 &= p(y = 1 | x, \theta')
 \end{aligned}$$

Logistic regression

Gradient Descent for logistic regression

- **Gradient Descent to minimize logistic regression cost function**

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

with identical algorithm as for linear regression

while not converged :

for all j :

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

$$\theta := \begin{bmatrix} tmp_0 \\ \vdots \\ tmp_n \end{bmatrix}$$

Beyond Gradient Descent

- Advanced Optimization

Advanced Optimization Algorithms

- **Given functions to compute**
 - $J(\theta)$
 - $\frac{\partial}{\partial \theta_j} J(\theta)$
- **an optimization algorithm will compute $\underset{\theta}{\operatorname{argmin}} J(\theta)$**

Optimization Algorithms

- *(Gradient Descent)*
- **Conjugate Gradient**
- **BFGS & L-BFGS**

Advantages

- **Often faster convergence**
- **No learning rate to choose**

Disadvantages

- **Complex**

Preimplemented Alorithms

- **Advanced optimization algorithms exist already in Machine Learning packages for important languages**
 - **Octave/Matlab**
 - **R**
 - **Java**
 - **Rapidminer – under the hood**

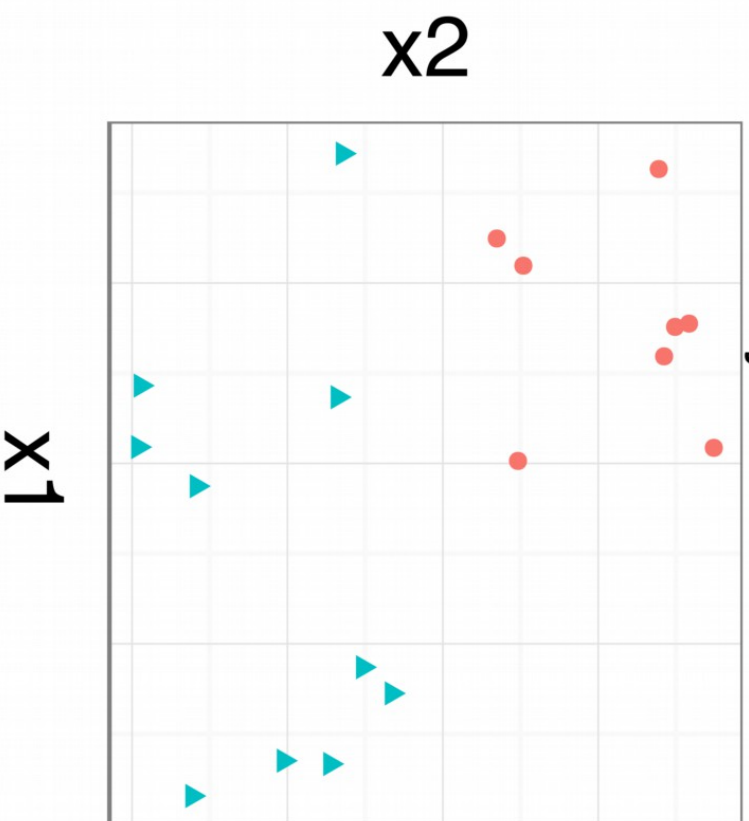
Multiclass Classification (by cheap trickery)

Multiclass classification problems

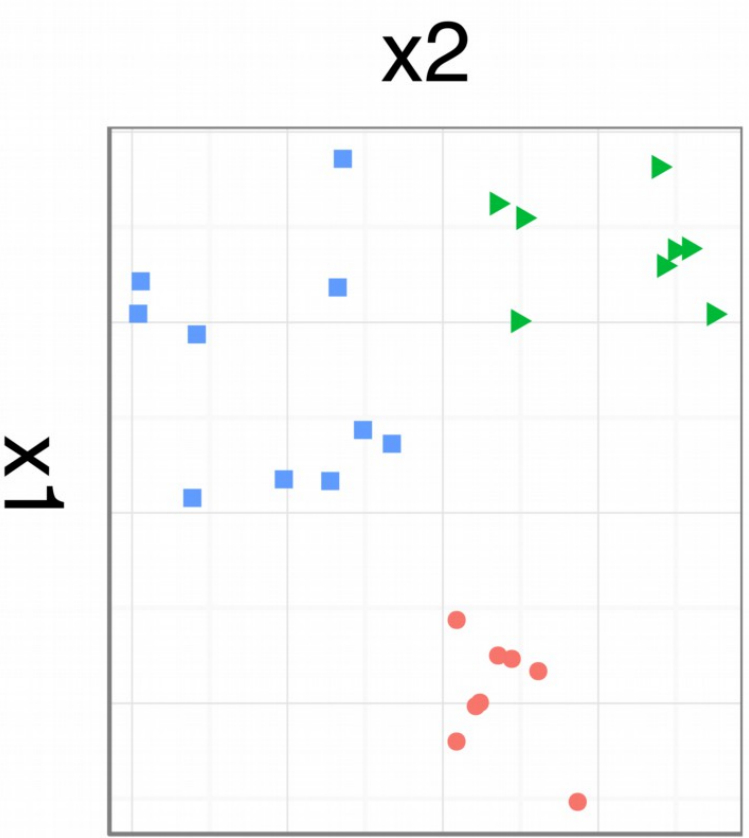
- **Classes of Emails: Work, Friends, Invoices, Job Offers**
- **Medical diagnosis: Not ill, Asthma, Lung Cancer**
- **Weather: Sunny, Cloudy, Rain, Snow**
- **Number classes as 1, 2, 3, ...**

Binary vs. Multiclass Classification

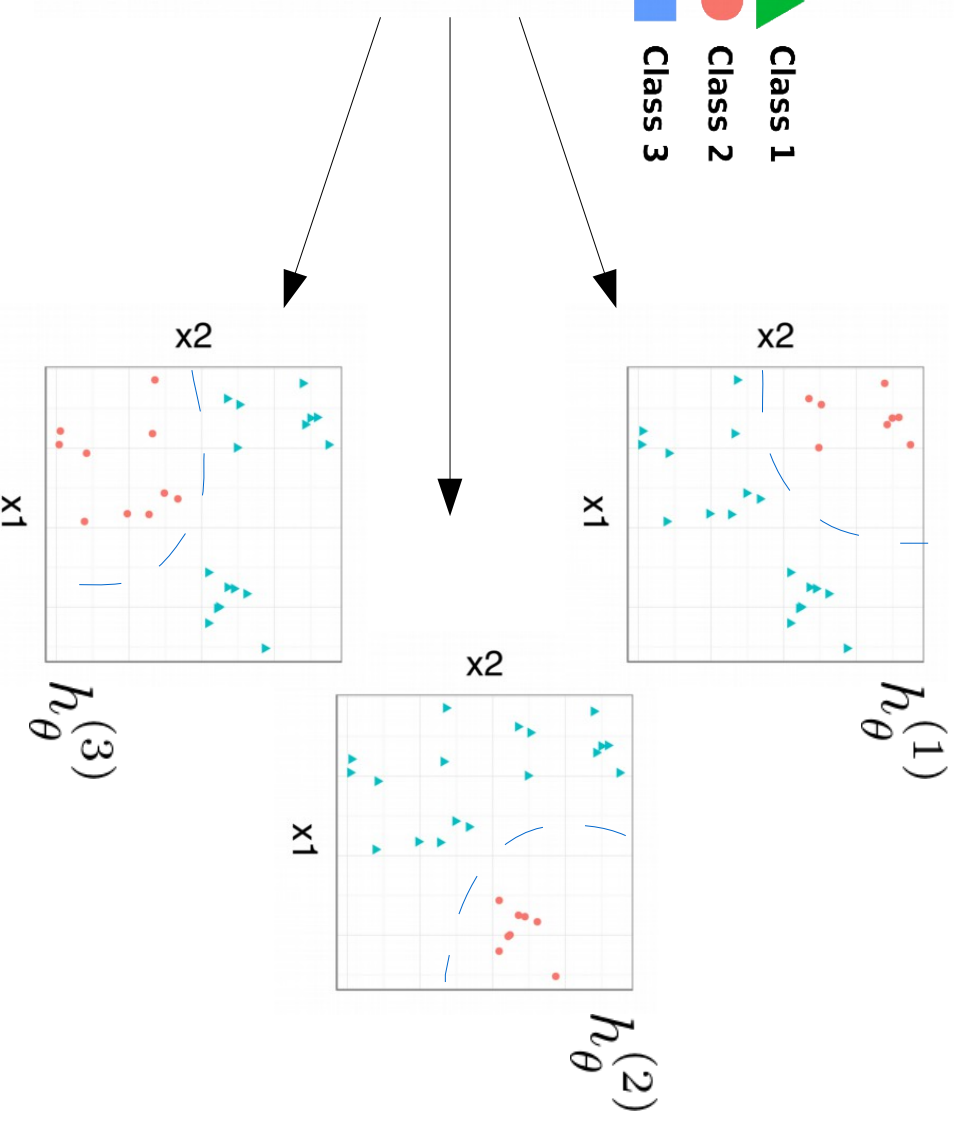
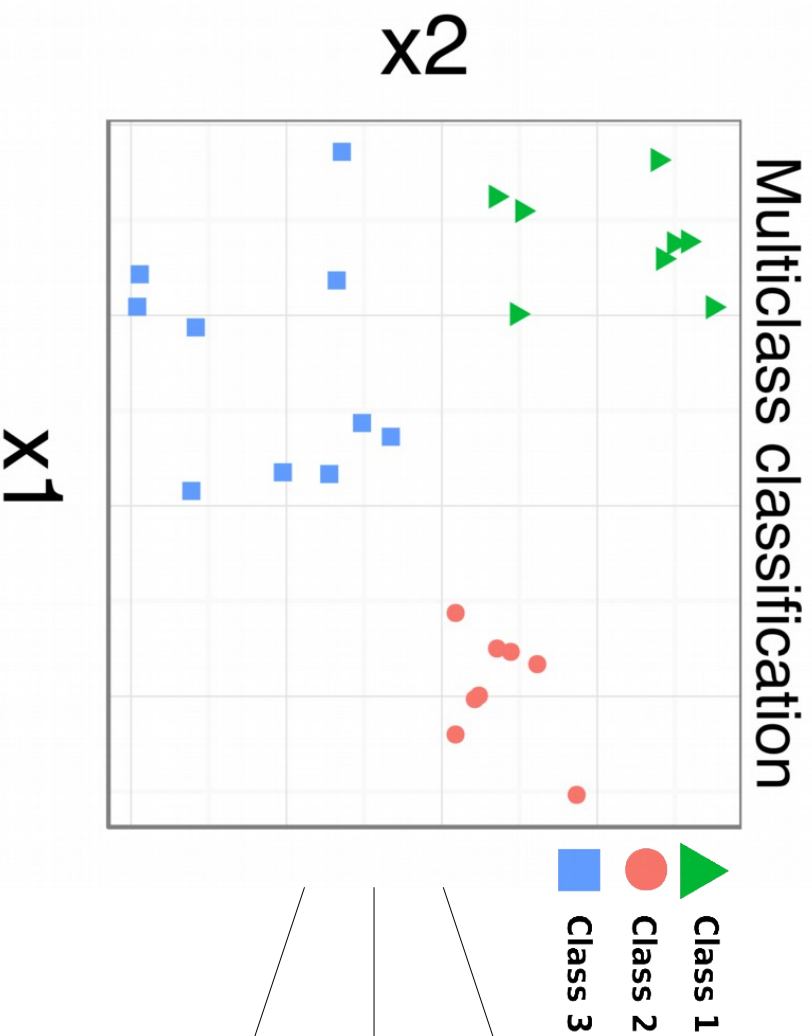
Binary classification



Multiclass classification



One versus all



$$h_{\theta}^{(i)} = P(y = i | x, \theta)$$

where $i \in \{1, 2, 3\}$

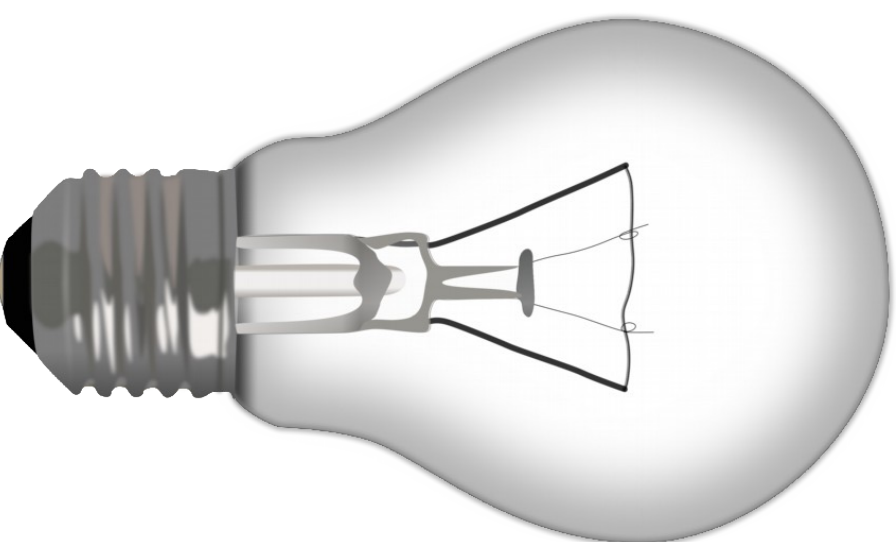
Logistic regression

- Train logistic regression classifier $h_{\theta}^{(i)}(x)$ for each class i to predict probability of $y = i$
- On new x predict class i which satisfies

$$\underset{i}{\operatorname{argmax}} h_{\theta}^{(i)}(x)$$

This lecture covered

- **Logistic regression hypothesis**
- **Decision Boundary**
- **Cost function(why we need a new one)**
- **Simplified Cost function & Gradient Descent**
- **Advanced Optimization Algorithms**
- **Multiclass classification**



Pictures

- *Tumor* picture by flickr-user *bc the path*, License CC SA NC
- Lightbulb picture from openclipart.org, public domain