10-701
Machine Learning

Logistic regression

Back to classification

1. Instance based classifiers
   - Use observation directly (no models)
   - e.g. K nearest neighbors

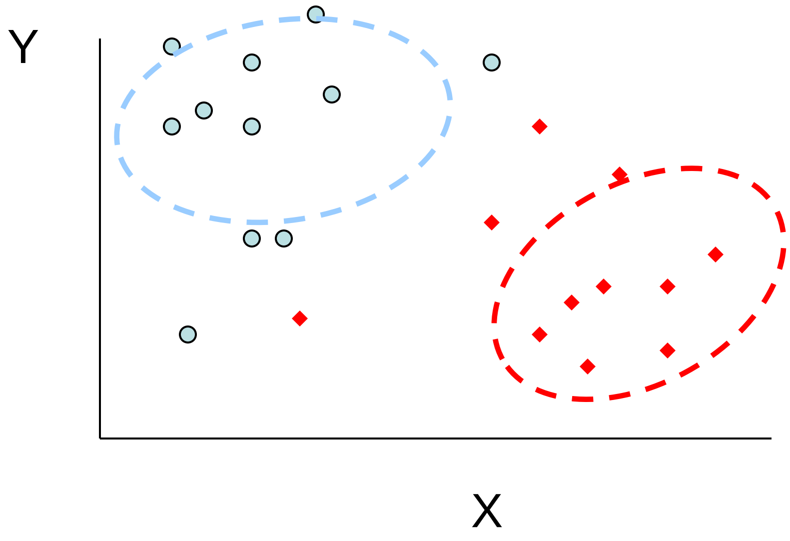
2. Generative:
   - build a generative statistical model
   - e.g., Bayesian networks

3. Discriminative
   - directly estimate a decision rule/boundary
   - e.g., decision tree

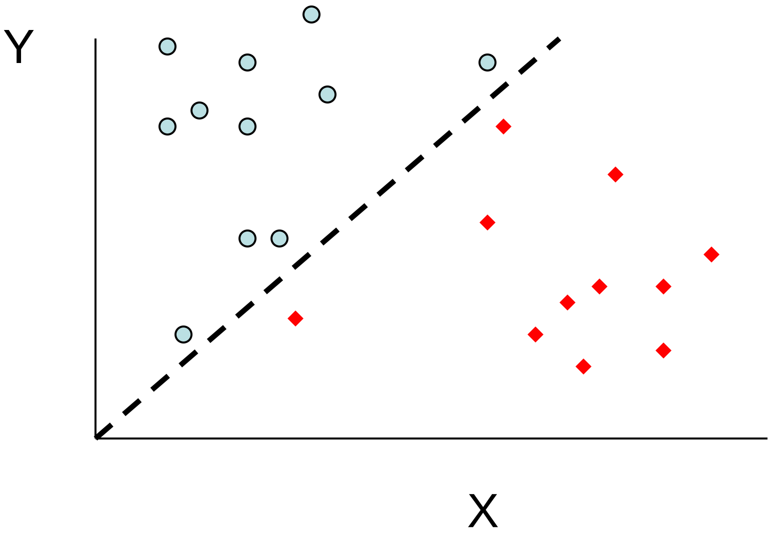
Generative vs. discriminative classifiers

- When using generative classifiers we relied on all points to learn the generative model
- When using discriminative classifiers we mainly care about the boundary

Generative model



Discriminative model



Regression for classification

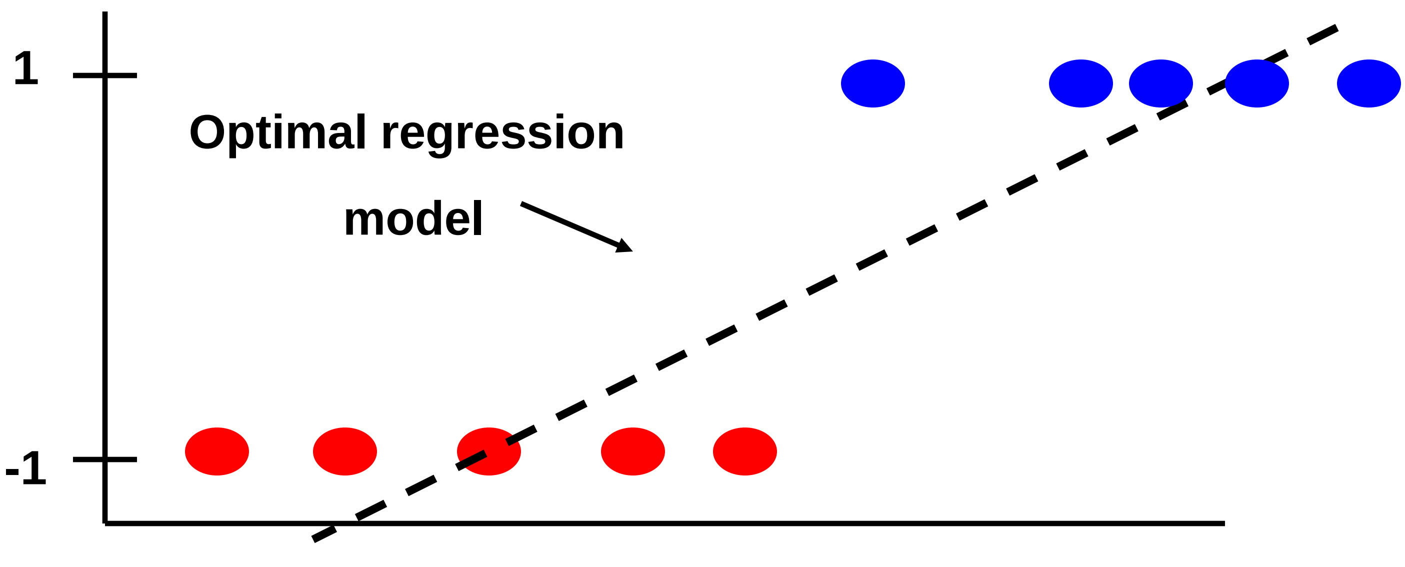
- In some cases we can use linear regression for determining the appropriate boundary.
- However, since the output is usually binary or discrete there are more efficient regression methods
- Recall that for classification we are interested in the conditional probability $p(y \mid x ; \theta)$ where θ are the parameters of our model
- When using regression θ represents the values of our regression coefficients (w).

Regression for classification

- Assume we would like to use linear regression to learn the parameters for $p(y \mid x ; \theta)$
- Problems?

$\mathbf{w}^T\mathbf{x} \geq 0 \Rightarrow$ classify as 1

$\mathbf{w}^T\mathbf{x} < 0 \Rightarrow$ classify as -1



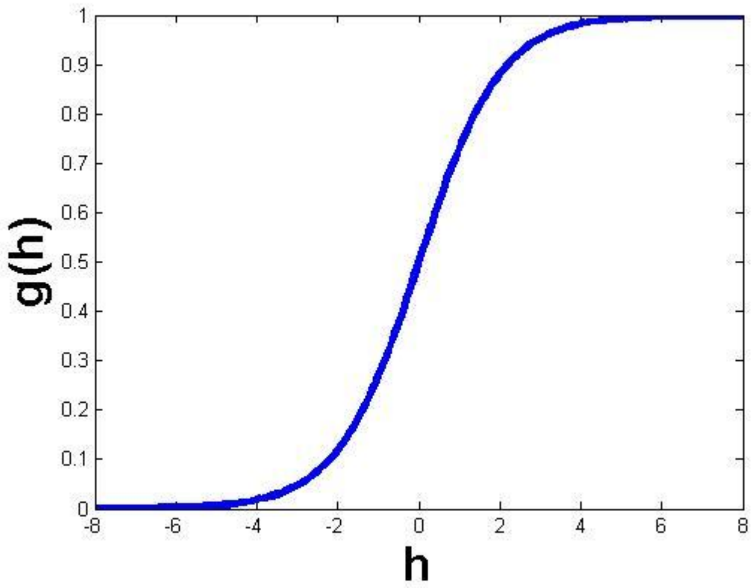
The sigmoid function

$$p(y \mid x;\theta)$$

- To classify using regression models we replace the linear function with the sigmoid function:

Always between 0 and 1

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



- Using the sigmoid we set (for binary classification problems)

$$p(y=0 \mid x;\theta) = g(\mathrm{w}^{\mathrm{T}}x) = \frac{1}{1+e^{\mathrm{w}^{\mathrm{T}}x}}$$

$$p(y=1 \mid x;\theta) = 1 - g(\mathrm{w}^{\mathrm{T}}x) = \frac{e^{\mathrm{w}^{\mathrm{T}}x}}{1+e^{\mathrm{w}^{\mathrm{T}}x}}$$

The sigmoid function

$$p(y \mid x;\theta)$$

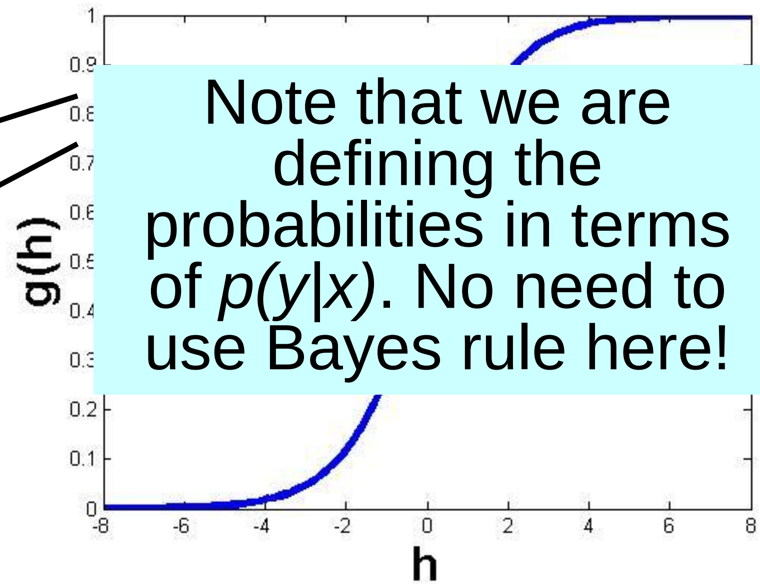
- To classify using regression models we replace the linear function with the sigmoid function:

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

- Using the sigmoid we set (for binary classification problems)

$$p(y=0 \mid x;\theta) = g(\mathrm{w}^{\mathrm{T}}x) = \frac{1}{1+e^{\mathrm{w}^{\mathrm{T}}x}}$$

$$p(y=1 \mid x;\theta) = 1 - g(\mathrm{w}^{\mathrm{T}}x) = \frac{e^{\mathrm{w}^{\mathrm{T}}x}}{1+e^{\mathrm{w}^{\mathrm{T}}x}}$$

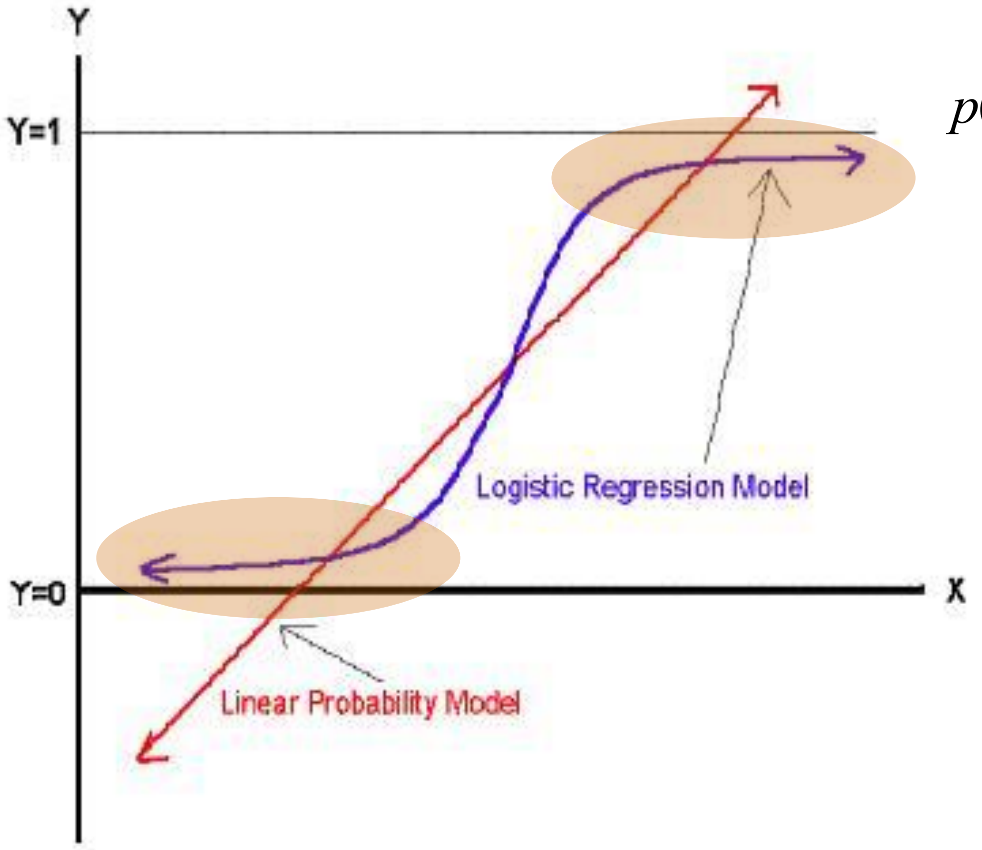


Note that we are defining the probabilities in terms of *p(y|x)*. No need to use Bayes rule here!

Logistic regression vs. Linear regression

$$p(y=0\,|\,x;\theta)=g(\mathrm{w}^{\mathrm{T}}x)=\frac{1}{1+e^{\mathrm{w}^{\mathrm{T}}x}}$$

$$p(y=1\,|\,x;\theta)=1-g(\mathrm{w}^{\mathrm{T}}x)=\frac{e^{\mathrm{w}^{\mathrm{T}}x}}{1+e^{\mathrm{w}^{\mathrm{T}}x}}$$



Determining parameters for logistic regression problems

- So how do we find the parameters?
- Similar to other regression problems we look for the MLE for w
- The likelihood of the data given the model is:

$$p(y=0 \mid x;\theta) = g(x;w) = \frac{1}{1+e^{w^T x}}$$

$$p(y=1 \mid x;\theta) = 1 - g(x;w) = \frac{e^{w^T x}}{1+e^{w^T x}}$$

$$L(y \mid x;w) = \prod_i (1-g(x^i;w))^{y^i} g(x^i;w)^{(1-y^i)}$$

Shorthand notation for using class 0 or 1

Solving logistic regression problems

$$g(x;w)=\frac{1}{1+e^{\mathrm{w}^{\mathrm{T}}x}}$$

$$1-g(x;w)=\frac{e^{\mathrm{w}^{\mathrm{T}}x}}{1+e^{\mathrm{w}^{\mathrm{T}}x}}$$

- The likelihood of the data is: $L(y\mid x;w)=\prod_i (1-g(x^i;w))^{y^i}\, g(x^i;w)^{(1-y^i)}$

- Taking the log we get:

$$LL(y\mid x;w)=\sum_{i=1}^{N} y^i \ln(1-g(x^i;w))+(1-y^i)\ln g(x^i;w)$$

$$=\sum_{i=1}^{N} y^i \ln\frac{1-g(x^i;w)}{g(x^i;w)}+\ln g(x^i;w)$$

$$=\sum_{i=1}^{N} y^i \mathrm{w}^{\mathrm{T}} x^i-\ln(1+e^{w^T x^i})$$

Maximum likelihood estimation

$$\frac{\partial}{\partial w_j} l(w) = \frac{\partial}{\partial w_j} \sum_{i=1}^{N} \{ y^i \mathrm{w}^{\mathrm{T}} x^i - \ln(1 + e^{\mathrm{w}^{\mathrm{T}} x^i}) \}$$

$$= \sum_{i=1}^{N} x_j^i \{ y^i - (1 - g(x^i; w)) \}$$

$$= \sum_{i=1}^{N} x_j^i \{ y^i - p(y^i = 1 \mid x; w) \}$$

$$g(x; w) = \frac{1}{1 + e^{\mathrm{w}^{\mathrm{T}} x}}$$

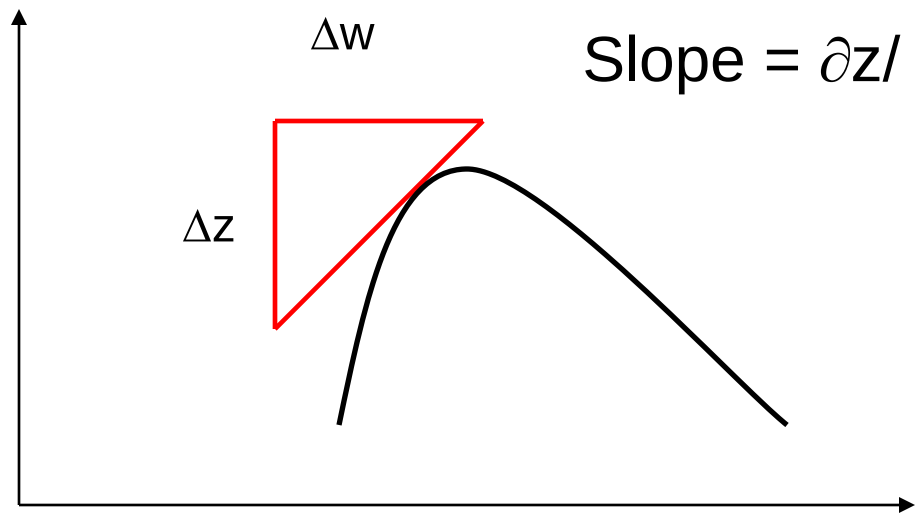
$$1 - g(x; w) = \frac{e^{\mathrm{w}^{\mathrm{T}} x}}{1 + e^{\mathrm{w}^{\mathrm{T}} x}}$$

Taking the partial derivative w.r.t. each component of the **w** vector

Bad news: No close form solution!

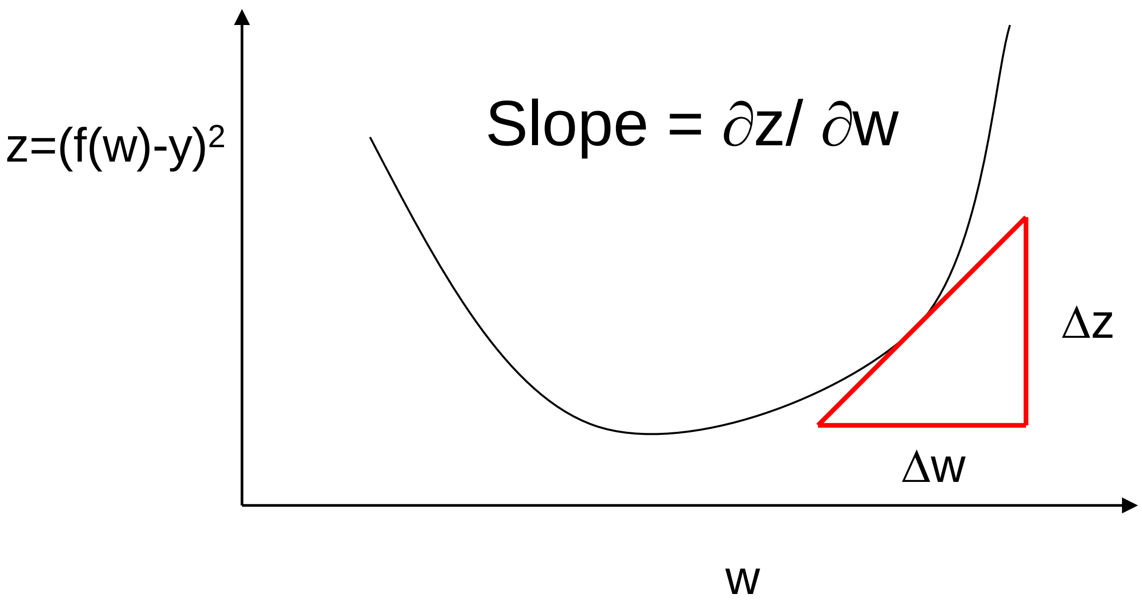
Good news: Concave function

Gradient ascent



- Going in the direction to the slope will lead to a larger z
- But not too much, otherwise we would go beyond the optimal w

Gradient descent



- Going in the *opposite* direction to the slope will lead to a smaller z
- But not too much, otherwise we would go beyond the optimal w

Gradient ascent for logistic regression

$$\frac{\partial}{\partial w_j} l(w) = \sum_{i=1}^{N} x_j^i \{y^i - (1 - g(x^i; w))\}$$

We use the gradient to adjust the value of w:

$$w_j \leftarrow w_j + \varepsilon \sum_{i=1}^{N} x_j^i \{y^i - (1 - g(x^i; w))\}$$

Where ε is a (small) constant

Algorithm for logistic regression

1. Chose λ
2. Start with a guess for **w**
3. For all j set
$$w_j \leftarrow w_j + \varepsilon \sum_{i=1}^{N} x_j^i \{y^i - (1 - g(x^i; w))\}$$
4. If no improvement for
$$\sum_{i=1}^{n} (y^i - (1 - g(x^i; w)))^2$$
stop. Otherwise go to step 3

Example

Regularization

- Like with other data estimation problems, we may not have enough data to learn good models
- One way to overcome this is to ‘regularize’ the model, impose additional constraints on the parameters we are fitting.
- For example, lets assume that w_i comes from a Guassian distribution with mean 0 and variance σ^2 (where σ^2 is a user defined parameter): $w_i \sim N(0, \sigma^2)$
- In that case we have:

$$p(y=1,\theta \mid x) \propto p(y=1 \mid x;\theta)\, p(\theta)$$

Regularization

- If we regularize the parameters we need to take the prior into account when computing the posterior for our parameters

$$p(y=1,\theta \mid x) \propto p(y=1 \mid x;\theta)p(\theta)$$

- Here we use a Gaussian model for the prior.
- Thus, the log likelihood changes to :

$$LL(y;w \mid x) = \sum_{i=1}^{N} y^i \mathrm{w}^{\mathrm{T}} x^i - \ln(1+e^{w^T x^i}) - \sum_j \frac{w_j^2}{2\sigma^2}$$

Assuming mean of 0 and removing terms that are not dependent on w

- And the new update rule (after taking the derivative w.r.t. w_i) is:

$$w_j \leftarrow w_j + \varepsilon \sum_{i=1}^{N} x_j^i \{y^i - (1 - g(x^i;w))\} - \varepsilon \frac{w_j}{\sigma^2}$$

Also known as the MAP estimate

The variance of our prior model

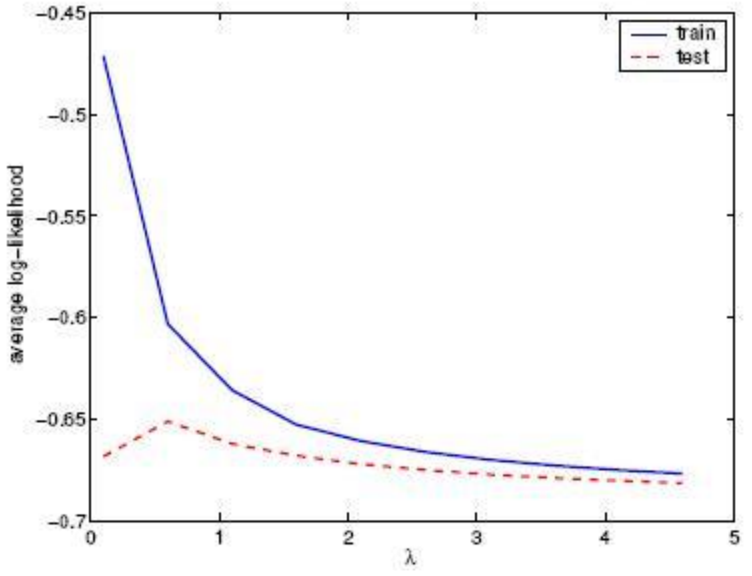
Regularization

- There are many other ways to regularize logistic regression
- The Gaussian model leads to an L2 regularization (we are trying to minimize the square value of *w*)
- Another popular regularization is an L1 which tries to minimize |w|

The importance of the regularization parameter

- Too small does not have a big impact
- Too large overrides the data
- An example of the training/test conditional log likelihoods as a function of the regularization parameter σ^2

Average log likelihood for data only



Logistic regression for more than 2 classes

- Logistic regression can be used to classify data from more than 2 classes:
- for *i*<*k* we set

$$p(y=i\,|\,x;\theta)=g(w_{i0}+w_{i1}x_1+\ldots+w_{id}x_d)=g(\mathrm{w}_\mathrm{i}^\mathrm{T}x)$$

where $$g(z_i)=\frac{e^{z_i}}{1+\sum_{j=1}^{k-1}e^{z_j}} \qquad z_i=w_{i0}+w_{i1}x_1+\ldots+w_{id}x_d$$

And for k we have $$p(y=k\,|\,x;\theta)=1-\sum_{i=1}^{k-1}p(y=i\,|\,x;\theta)\Rightarrow$$

$$p(y=k\,|\,x;\theta)=\frac{1}{1+\sum_{j=1}^{k-1}e^{z_j}}$$

Logistic regression for more than 2 classes

- Logistic regression can be used to classify data from more than 2 classes:
- for *i*<*k* we set

$$p(y=i \mid x;\theta) = g(w_{i0} + w_{i1}x_1 + \ldots + w_{id}x_d) = g(\mathrm{w}_\mathrm{i}^\mathrm{T} x)$$

where $$g(z_i) = \frac{e^{z_i}}{1 + \sum_{j=1}^{k-1} e^{z_j}} \qquad z_i = w_{i0} +$$

Binary logistic regression is a special case of this rule

And for k we have $$p(y=k \mid x;\theta) = 1 - \sum_{i=1}^{k-1} p(y=i \mid x;\theta) \Rightarrow$$

$$p(y=k \mid x;\theta) = \frac{1}{1 + \sum_{j=1}^{k-1} e^{z_j}}$$

Update rule for logistic regression with multiple classes

$$\frac{\partial}{\partial w_{m,j}} l(w) = \sum_{i=1}^{N} x_j^i \{\delta_m(y^i) - p(y^i = m \mid x^i; w)\}$$

Where $\delta(y^i)=1$ if $y^i=m$ and $\delta(y^i)=0$ otherwise

The update rule becomes:

$$w_{m,j} \leftarrow w_{m,j} + \varepsilon \sum_{i=1}^{N} x_j^i \{\delta_m(y^i) - p(y^i = m \mid x^i; w)\}$$

Additive models

- Similar to what we did with linear regression we can extend logistic regression to other transformations of the data

$$p(y=1 \mid x;w) = g(w_{i0} + w_1\phi_1(x) + \ldots + w_d\phi_d(x))$$

- As before, we are free to choose the basis functions

Important points

- Advantage of logistic regression over linear regression for classification
- Sigmoid function
- Gradient ascent / descent
- Regularization
- Logistic regression for multiple classes

Logistic regression

- The name comes from the **logit** transformation:

$$\log \frac{p(y=i \mid x;\theta)}{p(y=k \mid x;\theta)} = \log \frac{g(z_i)}{g(z_k)} = w_{i0} + w_{i1}x_1 + \ldots + w_{id}x_d$$