

CS195-5 : Introduction to Machine Learning

Lecture 11

Greg Shakhnarovich

September 29 2006

Announcements

- Problem set Nectarine out.
 - 13 problems + 1 optional;
should be easier than PS1.

Review

- Bayesian view of probability and estimation
 - parameters are random variables; prior $p(\theta)$.
 - probability measures of our degree of belief about an event.

$$p(X) = \int_{\theta} p(X | \theta) p(\theta) d\theta$$

- Conjugate prior: retains form after multiplying by likelihood
- Beta prior for binomial likelihood.
 - Interpretation: pseudocounts.

Today

- Introduction to discriminative models.
- Logistic regression
 - The model
 - Fitting logistic regression via gradient ascent.

Generative versus discriminative classifiers

- A generative approach:
 - Model joint probability density $p(\mathbf{x}, y)$ by modeling $p(\mathbf{x} | y), p(y)$.
 - Estimate the densities for each class, and infer decision boundary of Bayes classifier using Bayes rule;
Estimation may be based on ML, MAP or another principle.

Generative versus discriminative classifiers

- A generative approach:
 - Model joint probability density $p(\mathbf{x}, y)$ by modeling $p(\mathbf{x} | y), p(y)$.
 - Estimate the densities for each class, and infer decision boundary of Bayes classifier using Bayes rule;
Estimation may be based on ML, MAP or another principle.
- In contrast, a discriminative approach would:
 - Model the class posterior $p(y | \mathbf{x})$ directly; don't bother with $p(\mathbf{x}, y)$.
 - Estimate $p(y | \mathbf{x})$ from data, and construct Bayes classifier.

Discriminative models

- Why use discriminative models?

Discriminative models

- Why use discriminative models?
 - We may not have a good idea of what the class densities are, but may still be able to come up with a good decision boundary.

Discriminative models

- Why use discriminative models?
 - We may not have a good idea of what the class densities are, but may still be able to come up with a good decision boundary.
 - Fewer parameters to estimate, often with equivalent expressive power.
- Example: linear discriminant analysis, two classes.
 - Generative: must assume two Gaussians with equal covariances. Need to estimate

Discriminative models

- Why use discriminative models?
 - We may not have a good idea of what the class densities are, but may still be able to come up with a good decision boundary.
 - Fewer parameters to estimate, often with equivalent expressive power.
- Example: linear discriminant analysis, two classes.
 - Generative: must assume two Gaussians with equal covariances. Need to estimate $2 + 2d + d(d + 1)/2$ parameters.

Discriminative models

- Why use discriminative models?
 - We may not have a good idea of what the class densities are, but may still be able to come up with a good decision boundary.
 - Fewer parameters to estimate, often with equivalent expressive power.
- Example: linear discriminant analysis, two classes.
 - Generative: must assume two Gaussians with equal covariances. Need to estimate $2 + 2d + d(d + 1)/2$ parameters.
 - Discriminative: $p(y | \mathbf{x}) \propto$ a linear function. Need to estimate $d+1$ parameters only.

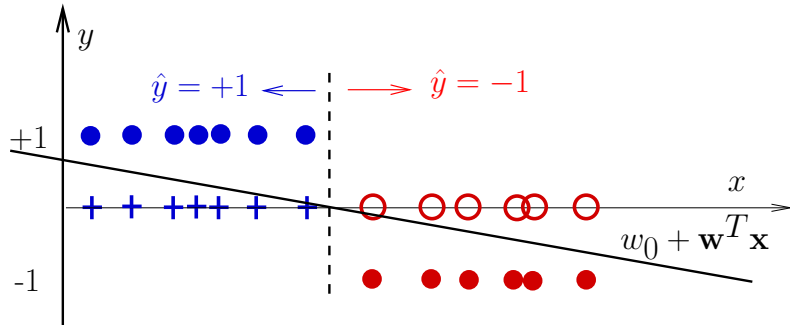
Discriminative models

- Why use discriminative models?
 - We may not have a good idea of what the class densities are, but may still be able to come up with a good decision boundary.
 - Fewer parameters to estimate, often with equivalent expressive power.
- Example: linear discriminant analysis, two classes.
 - Generative: must assume two Gaussians with equal covariances. Need to estimate $2 + 2d + d(d + 1)/2$ parameters.
 - Discriminative: $p(y | \mathbf{x}) \propto$ a linear function. Need to estimate $d+1$ parameters only.
- We already have seen a discriminative classification method:

Discriminative models

- Why use discriminative models?
 - We may not have a good idea of what the class densities are, but may still be able to come up with a good decision boundary.
 - Fewer parameters to estimate, often with equivalent expressive power.
- Example: linear discriminant analysis, two classes.
 - Generative: must assume two Gaussians with equal covariances. Need to estimate $2 + 2d + d(d + 1)/2$ parameters.
 - Discriminative: $p(y | \mathbf{x}) \propto$ a linear function. Need to estimate $d+1$ parameters only.
- We already have seen a discriminative classification method: least-squares regression on ± 1 labels, Lecture 5.

Reminder: classification via regression



- What are the drawbacks of fitting y with least squares?
 - No guarantee that y has valid values; sensitive to outliers.
 - Most importantly: no probabilistic interpretation or basis for the decision boundary.
- Optimal decision boundary is given by log-odds ratio:

$$\log \frac{p(y = 1 | \mathbf{x})}{p(y = 0 | \mathbf{x})} = 0.$$

The logistic model

- We can model the (unknown) decision boundary directly:

$$\log \frac{p(y = 1 | \mathbf{x})}{p(y = 0 | \mathbf{x})} = w_0 + \mathbf{w}^T \mathbf{x}.$$

- Since $p(y = 1 | \mathbf{x}) = 1 - p(y = 0 | \mathbf{x})$, we have (after exponentiating):

$$\frac{p(y = 1 | \mathbf{x})}{1 - p(y = 1 | \mathbf{x})} = \exp(w_0 + \mathbf{w}^T \mathbf{x})$$

The logistic model

- We can model the (unknown) decision boundary directly:

$$\log \frac{p(y = 1 | \mathbf{x})}{p(y = 0 | \mathbf{x})} = w_0 + \mathbf{w}^T \mathbf{x}.$$

- Since $p(y = 1 | \mathbf{x}) = 1 - p(y = 0 | \mathbf{x})$, we have (after exponentiating):

$$\begin{aligned} \frac{p(y = 1 | \mathbf{x})}{1 - p(y = 1 | \mathbf{x})} &= \exp(w_0 + \mathbf{w}^T \mathbf{x}) \\ \Rightarrow \frac{1}{p(y = 1 | \mathbf{x})} &= 1 + \exp(-w_0 - \mathbf{w}^T \mathbf{x}) \end{aligned}$$

The logistic model

- We can model the (unknown) decision boundary directly:

$$\log \frac{p(y = 1 | \mathbf{x})}{p(y = 0 | \mathbf{x})} = w_0 + \mathbf{w}^T \mathbf{x}.$$

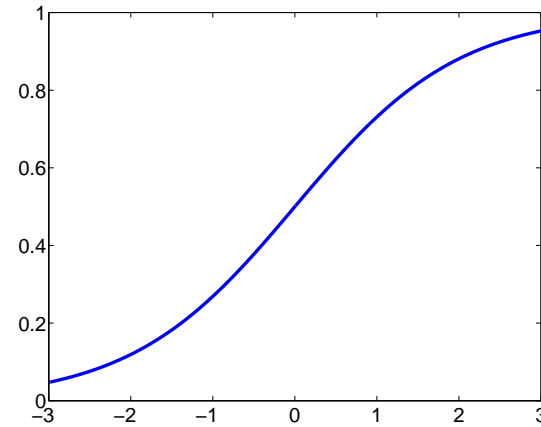
- Since $p(y = 1 | \mathbf{x}) = 1 - p(y = 0 | \mathbf{x})$, we have (after exponentiating):

$$\begin{aligned} \frac{p(y = 1 | \mathbf{x})}{1 - p(y = 1 | \mathbf{x})} &= \exp(w_0 + \mathbf{w}^T \mathbf{x}) \\ \Rightarrow \frac{1}{p(y = 1 | \mathbf{x})} &= 1 + \exp(-w_0 - \mathbf{w}^T \mathbf{x}) \\ \Rightarrow p(y = 1 | \mathbf{x}) &= \frac{1}{1 + \exp(-w_0 - \mathbf{w}^T \mathbf{x})}. \end{aligned}$$

The logistic function

$$\sigma(x) = \frac{1}{1 + e^{-x}}.$$

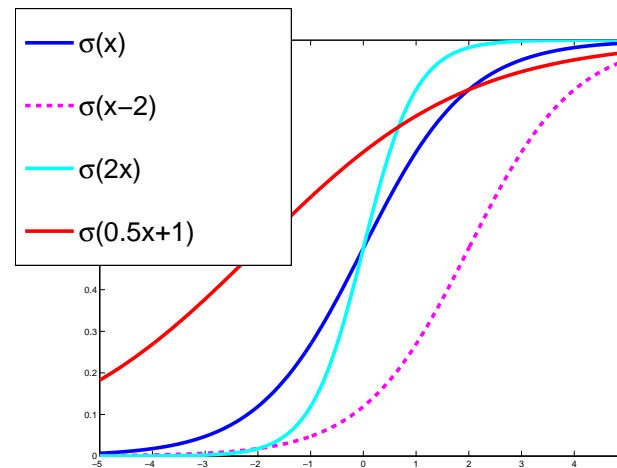
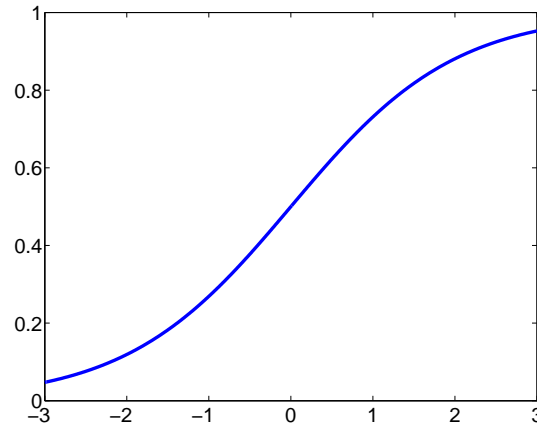
- For any x , $0 \leq \sigma(x) \leq 1$;
- Monotonically increasing; $\sigma(-\infty) = 0$;
 $\sigma(+\infty) = 1$.



The logistic function

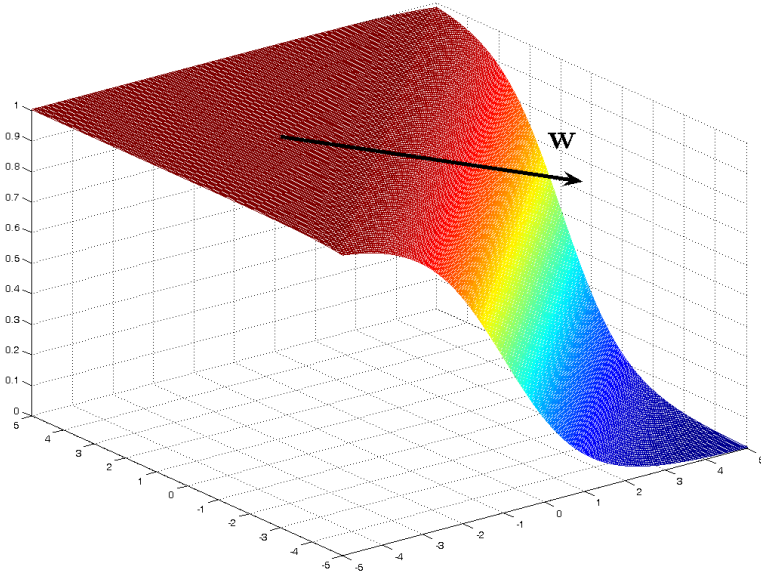
$$\sigma(x) = \frac{1}{1 + e^{-x}}.$$

- For any x , $0 \leq \sigma(x) \leq 1$;
- Monotonically increasing; $\sigma(-\infty) = 0$;
 $\sigma(+\infty) = 1$.
- $\sigma(0) = 1/2$. To shift the crossing to
an arbitrary z : $\sigma(x - z)$.
- To change the “slope”: $\sigma(ax)$.



Logistic function in \mathbb{R}^d

- What if $\mathbf{x} \in \mathbb{R}^d = [x_1 \dots x_d]^T$?
- $\sigma(w_0 + \mathbf{w}^T \mathbf{x})$ is a scalar function of a scalar variable $w_0 + \mathbf{w}^T \mathbf{x}$.

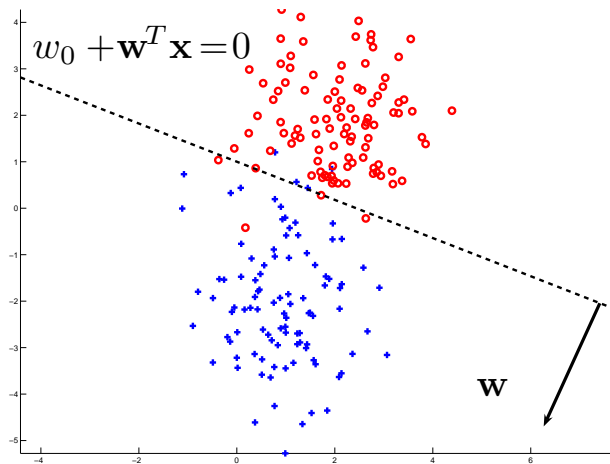


- the direction of \mathbf{w} determines orientation;
- w_0 determines the location;
- $\|\mathbf{w}\|$ determines the slope.

Logistic regression: decision boundary

$$p(y = 1 | \mathbf{x}) = \sigma(w_0 + \mathbf{w}^T \mathbf{x}) = 1/2 \Leftrightarrow w_0 + \mathbf{w}^T \mathbf{x} = 0$$

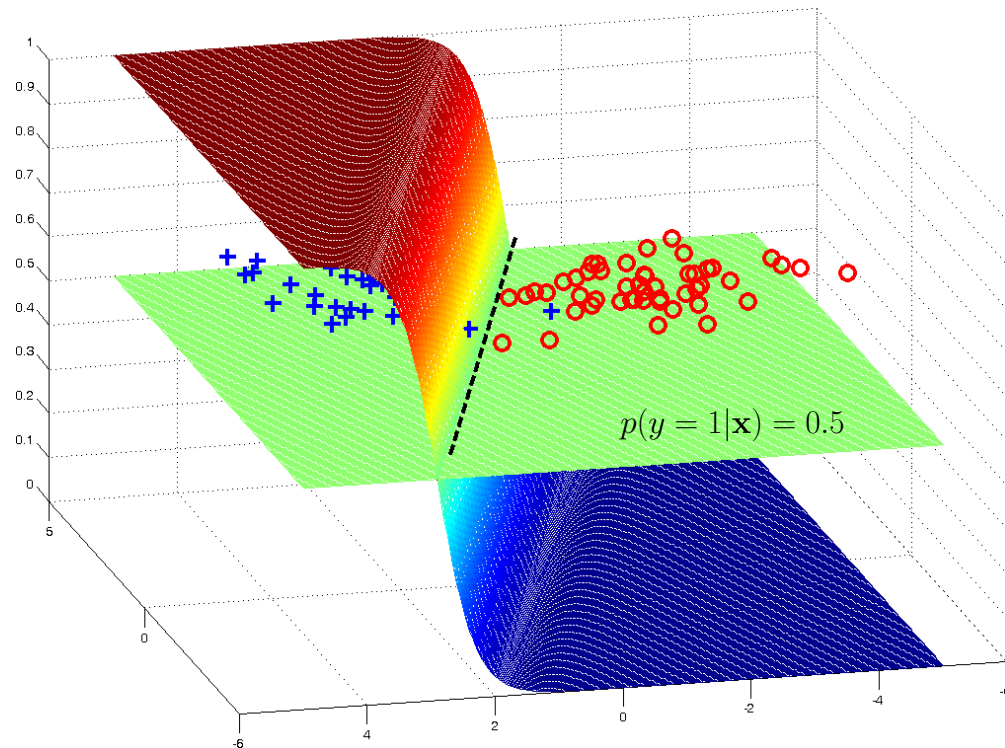
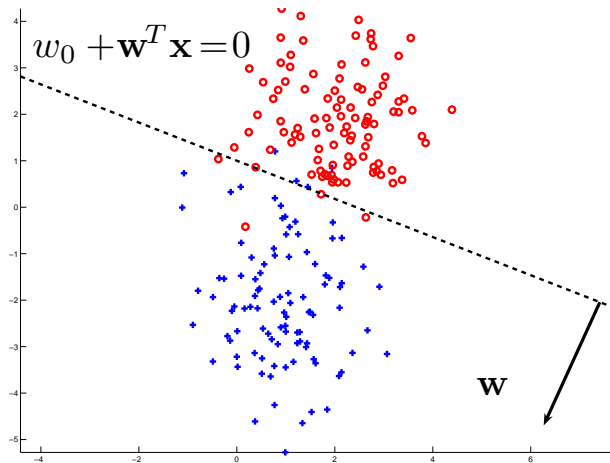
- With linear logistic model we get a linear decision boundary.



Logistic regression: decision boundary

$$p(y = 1 | \mathbf{x}) = \sigma(w_0 + \mathbf{w}^T \mathbf{x}) = 1/2 \Leftrightarrow w_0 + \mathbf{w}^T \mathbf{x} = 0$$

- With linear logistic model we get a linear decision boundary.



Likelihood under the logistic model

- “Regular” regression: observe values, measure their distance from the model.
- Logistic regression: observe labels, measure their probability under the model.

$$p(y_i | \mathbf{x}_i; \mathbf{w}) = \begin{cases} \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) & \text{if } y_i = 1, \\ 1 - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) & \text{if } y_i = 0 \end{cases}$$

Likelihood under the logistic model

- “Regular” regression: observe values, measure their distance from the model.
- Logistic regression: observe labels, measure their probability under the model.

$$\begin{aligned} p(y_i | \mathbf{x}_i; \mathbf{w}) &= \begin{cases} \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) & \text{if } y_i = 1, \\ 1 - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) & \text{if } y_i = 0 \end{cases} \\ &= \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)^{y_i} (1 - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i))^{1-y_i}. \end{aligned}$$

Likelihood under the logistic model

- “Regular” regression: observe values, measure their distance from the model.
- Logistic regression: observe labels, measure their probability under the model.

$$\begin{aligned} p(y_i | \mathbf{x}_i; \mathbf{w}) &= \begin{cases} \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) & \text{if } y_i = 1, \\ 1 - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) & \text{if } y_i = 0 \end{cases} \\ &= \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)^{y_i} (1 - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i))^{1-y_i}. \end{aligned}$$

- The log-likelihood of \mathbf{w} :

$$\ell(X_N; \mathbf{w}) = \sum_{i=1}^N \log p(y_i | \mathbf{x}_i; \mathbf{w})$$

Likelihood under the logistic model

- “Regular” regression: observe values, measure their distance from the model.
- Logistic regression: observe labels, measure their probability under the model.

$$\begin{aligned} p(y_i | \mathbf{x}_i; \mathbf{w}) &= \begin{cases} \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) & \text{if } y_i = 1, \\ 1 - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) & \text{if } y_i = 0 \end{cases} \\ &= \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)^{y_i} (1 - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i))^{1-y_i}. \end{aligned}$$

- The log-likelihood of \mathbf{w} :

$$\begin{aligned} \ell(X_N; \mathbf{w}) &= \sum_{i=1}^N \log p(y_i | \mathbf{x}_i; \mathbf{w}) \\ &= \sum_{i=1}^N y_i \log \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i) + (1 - y_i) \log (1 - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)) \end{aligned}$$

The maximum likelihood solution

- Setting the derivatives to zero, we get

$$\frac{\partial}{\partial w_0} \ell(X_N; \mathbf{w}) = \sum_{i=1}^N (y_i - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)) = 0;$$

$$\frac{\partial}{\partial w_j} \ell(X_N; \mathbf{w}) = \sum_{i=1}^N (y_i - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)) x_{ij} = 0.$$

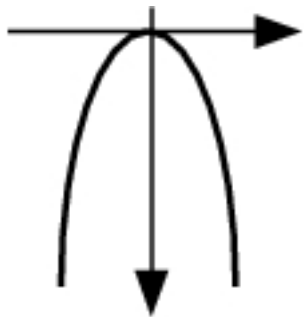
- We can treat $y_i - p(y_i | \mathbf{x}_i) = y_i - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)$ as the *prediction error* of the model on \mathbf{x}_i, y_i .
- As with linear regression: prediction errors are uncorrelated with any linear function of the data.

Finding the maximum

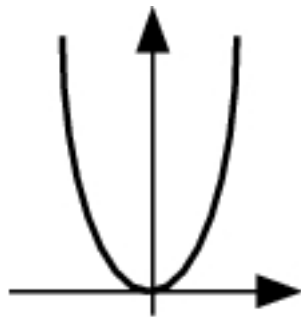
- Unfortunately, there is no close form solution for the maximum likelihood under the logistic model.
- However, $\ell(X_N; \mathbf{w})$ is *jointly concave* in all components of \mathbf{w} .

Finding the maximum

- Unfortunately, there is no close form solution for the maximum likelihood under the logistic model.
- However, $\ell(X_N; \mathbf{w})$ is *jointly concave* in all components of \mathbf{w} .



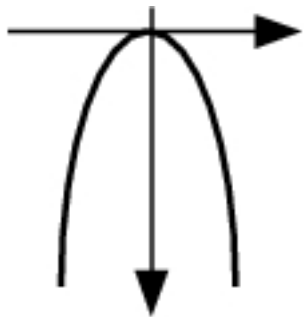
*concave
function*



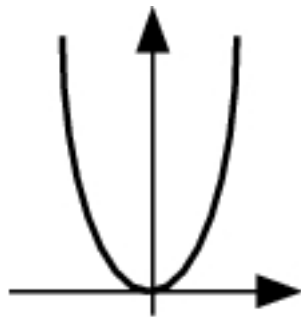
*convex
function*

Finding the maximum

- Unfortunately, there is no close form solution for the maximum likelihood under the logistic model.
- However, $\ell(X_N; \mathbf{w})$ is *jointly concave* in all components of \mathbf{w} .



*concave
function*



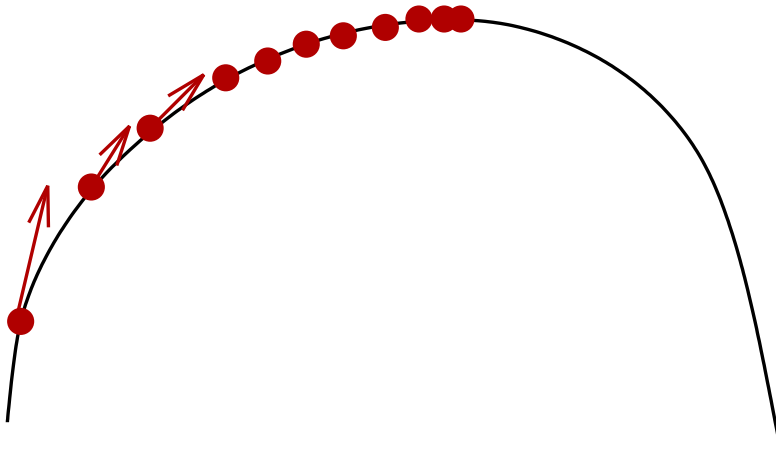
*convex
function*

- Equivalently, the error is convex.
- This means there exist a single (global) maximum!

- A number of methods exist for finding it.

Gradient ascent/descent

- The idea behind gradient ascent: “hill climbing” on the function surface.



- Start at a (random) location
- Make steps in the direction of maximal altitude increase.

- For concave functions, guaranteed to converge to the global maximum
 - Subject to some technical assumptions.
- An equivalent: gradient *descent* on the error $-\log p(y | \mathbf{x}; \mathbf{w})$

Stochastic gradient ascent

- An incremental algorithm:
 - Present examples (\mathbf{x}_i, y_i) one at a time,
 - Modify \mathbf{w} slightly to increase the log-probability of observed y_i :

$$\mathbf{w} := \mathbf{w} + \eta \frac{\partial}{\partial \mathbf{w}} \log p(y_i | \mathbf{x}_i; \mathbf{w})$$

where the *learning rate* η determines how “slightly”.

- The gradient of log-probability:

$$\frac{\partial}{\partial \mathbf{w}} \log p(y_i | \mathbf{x}_i; \mathbf{w}) = \begin{bmatrix} y_i - p(y_i | \mathbf{x}; \mathbf{w}) \\ (y_i - p(y_i | \mathbf{x}; \mathbf{w})) x_{i1} \\ \dots \\ (y_i - p(y_i | \mathbf{x}; \mathbf{w})) x_{id} \end{bmatrix} = (y_i - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)) \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix}.$$

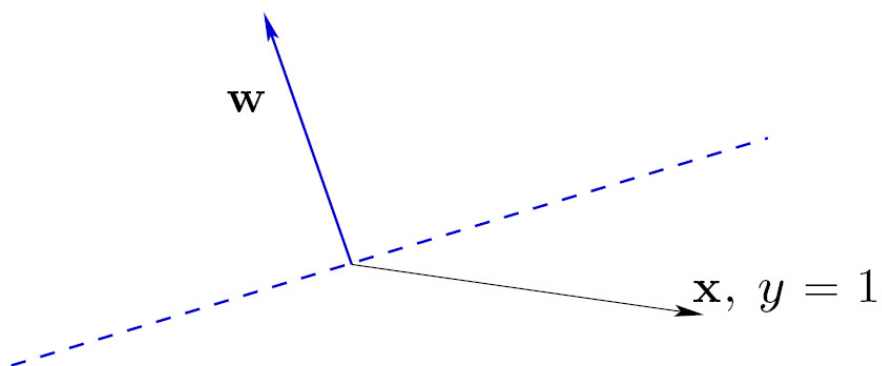
Stochastic gradient ascent

$$\begin{aligned}\mathbf{w}_{new} &:= \mathbf{w} + \eta \frac{\partial}{\partial \mathbf{w}} \log p(y_i | \mathbf{x}_i; \mathbf{w}) \\ &= \mathbf{w} + \eta (y_i - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)) \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix}.\end{aligned}$$

Stochastic gradient ascent

$$\begin{aligned}\mathbf{w}_{new} &:= \mathbf{w} + \eta \frac{\partial}{\partial \mathbf{w}} \log p(y_i | \mathbf{x}_i; \mathbf{w}) \\ &= \mathbf{w} + \eta (y_i - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)) \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix}.\end{aligned}$$

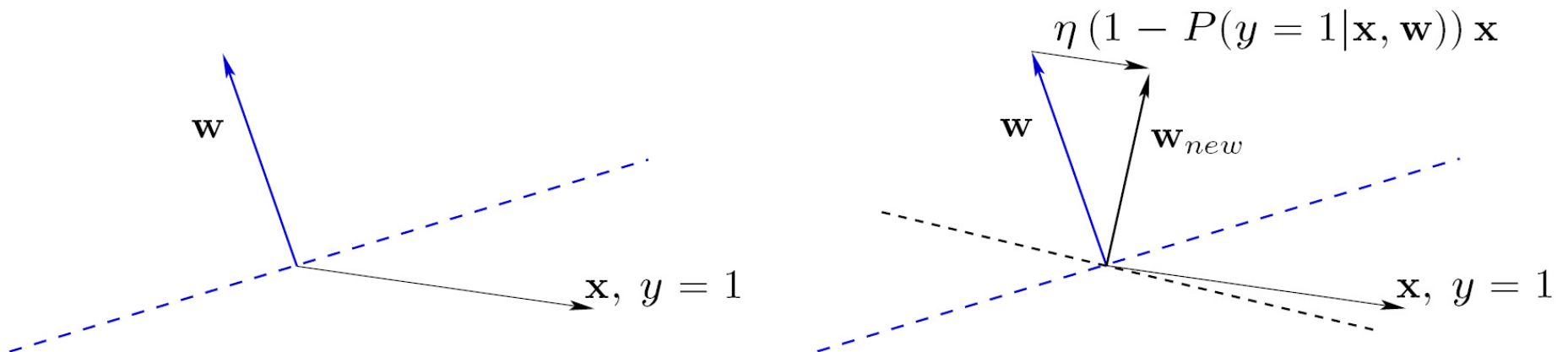
- Focus on one example (and ignore w_0 for simplicity):



Stochastic gradient ascent

$$\begin{aligned}\mathbf{w}_{new} &:= \mathbf{w} + \eta \frac{\partial}{\partial \mathbf{w}} \log p(y_i | \mathbf{x}_i; \mathbf{w}) \\ &= \mathbf{w} + \eta (y_i - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)) \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix}.\end{aligned}$$

- Focus on one example (and ignore w_0 for simplicity):



Batch gradient ascent

- We can also update \mathbf{w} by presenting all examples at once
 - Direct gradient ascent on the log-likelihood.

$$\begin{aligned}\mathbf{w}_{new} &:= \mathbf{w} + \eta \frac{\partial}{\partial \mathbf{w}} \ell(X_N; \mathbf{w}) \\ &= \mathbf{w} + \eta \sum_{i=1}^N (y_i - \sigma(w_0 + \mathbf{w}^T \mathbf{x}_i)) \begin{bmatrix} 1 \\ \mathbf{x}_i \end{bmatrix}\end{aligned}$$

- We need to choose η rather carefully:
 - Too small \Rightarrow slow convergence;
 - Too large: \Rightarrow overshoot and oscillation.

Newton-Raphson

- The *Newton-Raphson* algorithm: approximate the local shape of ℓ as a quadratic function.

$$\mathbf{w}_{new} := \mathbf{w} + \mathbf{H}^{-1} \frac{\partial}{\partial \mathbf{w}} \ell(X_N; \mathbf{w}),$$

where \mathbf{H} is the *Hessian* matrix of second derivatives:

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2}{\partial w_0^2} & \frac{\partial^2}{\partial w_0 w_1} & \cdots & \frac{\partial^2}{\partial w_0 w_d} \\ \frac{\partial^2}{\partial w_0 w_1} & \frac{\partial^2}{\partial w_1^2} & \cdots & \frac{\partial^2}{\partial w_1 w_d} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2}{\partial w_d w_0} & \frac{\partial^2}{\partial w_d w_1} & \cdots & \frac{\partial^2}{\partial w_d^2} \end{bmatrix} \ell(X_N; \mathbf{w}).$$

Generalized additive models

- As with regression we can extend this framework to arbitrary features (basis functions):

$$p(y = 1 | \mathbf{x}) = \sigma(w_0 + \phi_1(\mathbf{x}) + \dots + \phi_m(\mathbf{x})).$$

- Example: quadratic logistic regression in 2D

$$p(y = 1 | \mathbf{x}) = \sigma(w_0 + w_1x_1 + w_2x_2 + w_3x_1^2 + w_4x_2^2).$$

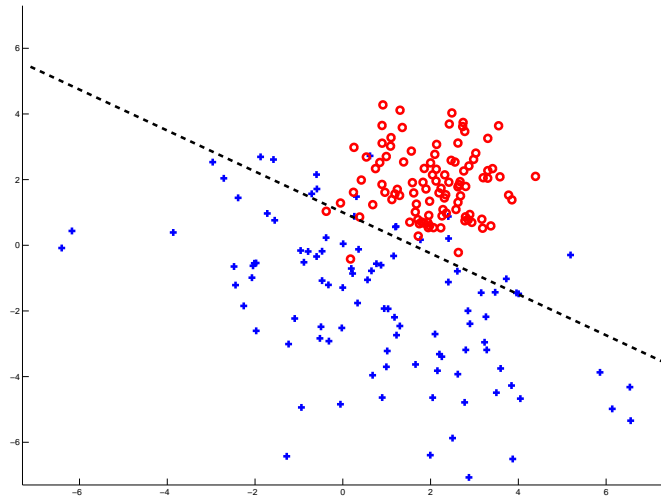
- Decision boundary of this classifier:

$$w_0 + w_1x_1 + w_2x_2 + w_3x_1^2 + w_4x_2^2 = 0,$$

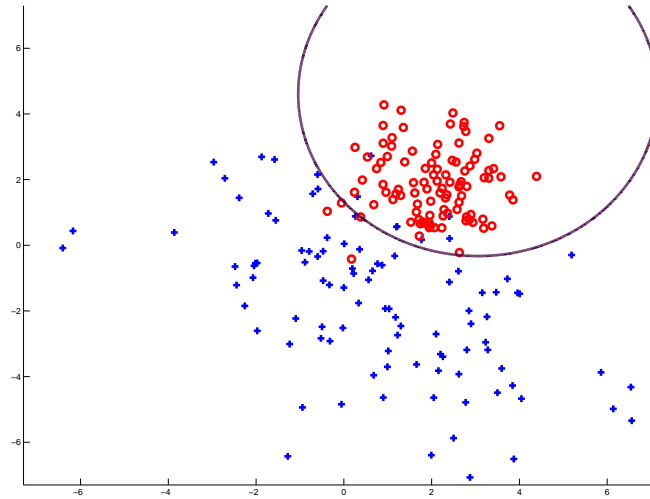
i.e. it's a quadratic decision boundary.

Logistic regression: 2D example

Linear

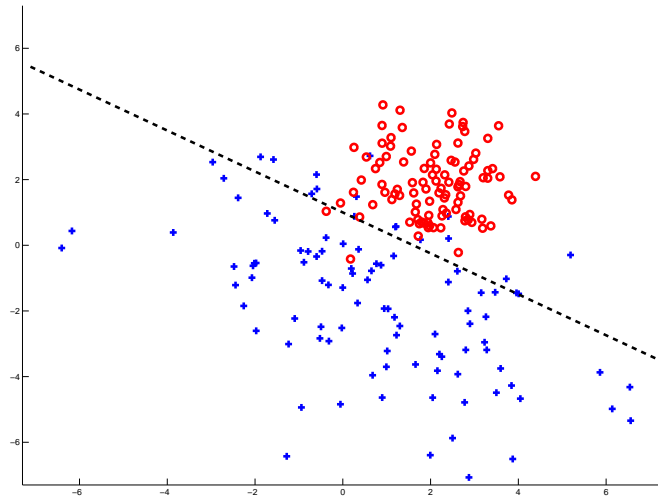


Quadratic

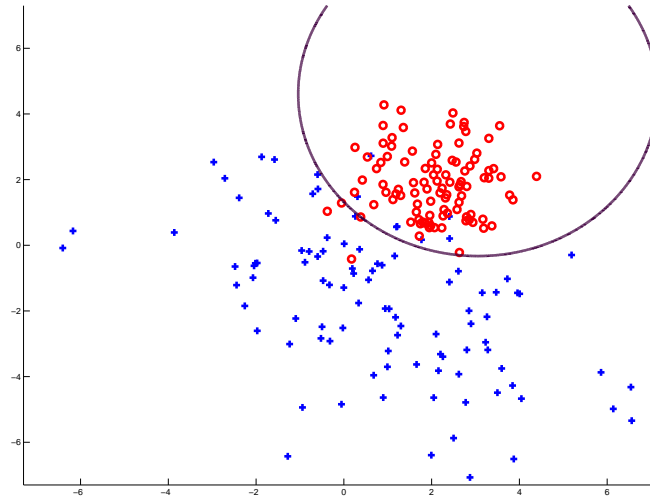


Logistic regression: 2D example

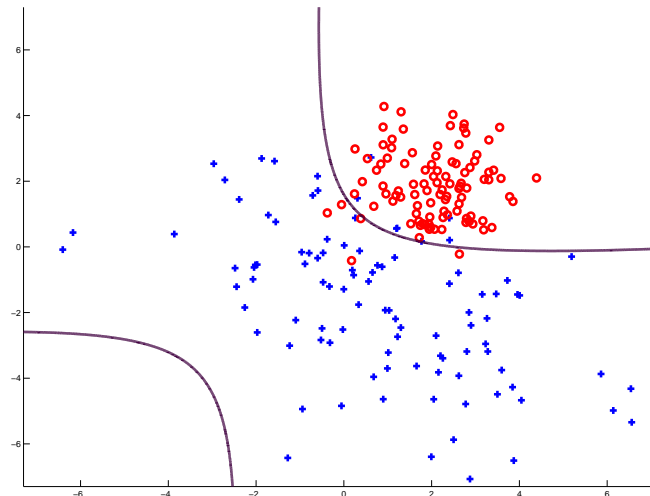
Linear



Quadratic



We can also include x_1x_2 :



Next time

Softmax: multiclass extension of logistic regression.

Computational issues.

Regularization: a principled way of dealing with overfitting.