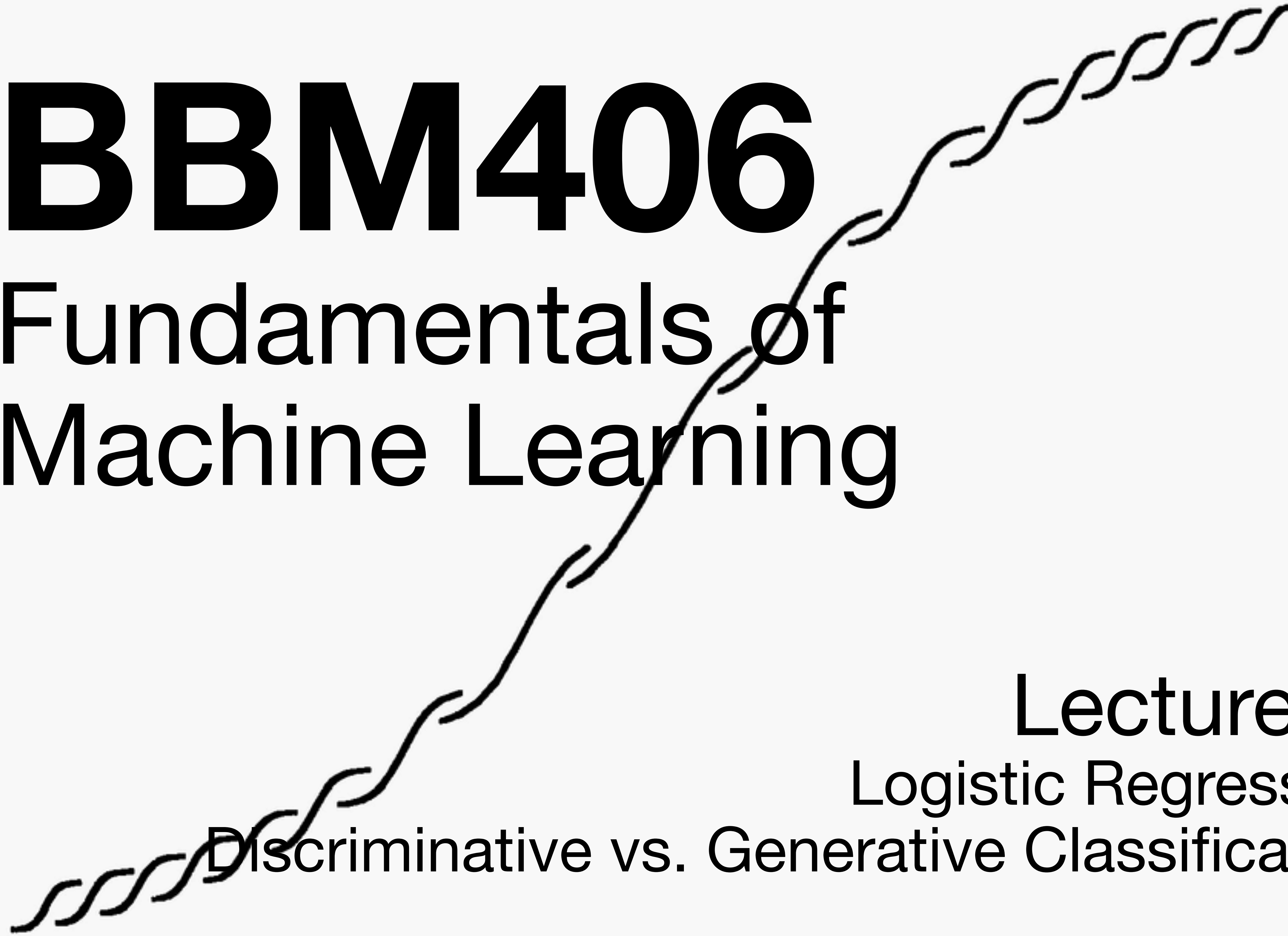
Illustration: Theodore Modis



BBM406

Fundamentals of Machine Learning

Lecture 9:

Logistic Regression

Discriminative vs. Generative Classification

HACETTEPE UNIVERSITY COMPUTER VISION LAB

Erkut Erdem // Hacettepe University // Fall 2021

Last time… Naïve Bayes Classifier

Given:

- Class prior $P(Y)$
- d conditionally independent features $X_1,\ldots X_d$ given the class label Y
- For each X_i feature, we have the conditional likelihood $P(X_i|Y)$

Naïve Bayes Decision rule:

$$
\begin{aligned}
f_{NB}(\mathbf{x}) &= \arg\max_y \; P(x_1, \ldots, x_d \mid y) P(y) \\
&= \arg\max_y \prod_{i=1}^{d} P(x_i|y) P(y)
\end{aligned}
$$

slide by Barnabás Póczos & Aarti Singh

Last time… Naïve Bayes Algorithm for discrete features

$$f_{NB}(\mathbf{x}) = \arg\max_y \prod_{i=1}^{d} P(x_i|y)P(y)$$

We need to estimate these probabilities!

Estimators

For Class Prior

$$\widehat{P}(y) = \frac{\{\#j : Y^{(j)} = y\}}{n}$$

For Likelihood

$$\frac{\widehat{P}(x_i, y)}{\widehat{P}(y)} = \frac{\{\#j : X_i^{(j)} = x_i, Y^{(j)} = y\}/n}{\{\#j : Y^{(j)} = y\}/n}$$

NB Prediction for test data: $X = (x_1, \ldots, x_d)$

$$Y = \arg\max_y \widehat{P}(y) \prod_{i=1}^{d} \frac{\widehat{P}(x_i, y)}{\widehat{P}(y)}$$

slide by Barnabás Póczos & Aarti Singh

Last time... Text Classification

MEDLINE Article


Brain and Cognition

Syntactic frame and verb bias in aphasia: Plausibility judgments of undergoer-subject sentences

?

MeSH Subject Category Hierarchy

- Antogonists and Inhibitors
- Blood Supply
- Chemistry
- Drug Therapy
- Embryology
- Epidemiology
- …

How to represent a text document?

slide by Dan Jurafsky

Last time... Bag of words model

Typical additional assumption:

Position in document doesn't matter:

$P(X_i=x_i|Y=y) = P(X_k=x_i|Y=y)$

- "Bag of words" model – order of words on the page ignored
  The document is just a bag of words: i.i.d. words
- Sounds really silly, but often works very well!

$\Rightarrow$ K(50000-1) parameters to estimate

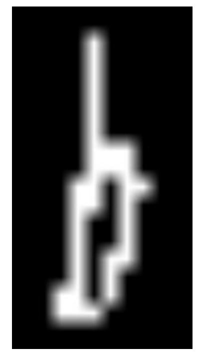
The probability of a document with words $x_1, x_2, \ldots$

$$\prod_{i=1}^{LengthDoc} P(x_i|y) = \prod_{w=1}^{W} P(w|y)^{count_w}$$

slide by Barnabás Póczos & Aarti Singh

Last time… What if features are continuous?

e.g., character recognition: X_i is intensity at i^{th} pixel

Gaussian Naïve Bayes (GNB):

$$P(X_i = x \mid Y = y_k) = \frac{1}{\sigma_{ik}\sqrt{2\pi}} \; e^{\frac{-(x-\mu_{ik})^2}{2\sigma_{ik}^2}}$$

Different mean and variance for each class k and each pixel i.

Sometimes assume variance
- is independent of Y (i.e., σ_i),
- or independent of X_i (i.e., σ_k)
- or both (i.e., σ)

$$\boxed{\widehat{\mu}_{MLE} = \frac{1}{N}\sum_{j=1}^{N} x_j}$$

$$\boxed{\widehat{\sigma}^2_{unbiased} = \frac{1}{N-1}\sum_{j=1}^{N}(x_j - \widehat{\mu})^2}$$

slide by Barnabás Póczos & Aarti Singh

Logistic Regression

Recap: Naïve Bayes

- NB Assumption: $P(X_1 ... X_d|Y) = \prod_{i=1}^{d} P(X_i|Y)$

- NB Classifier:

$$f_{NB}(x) \;=\; \arg\max_{y} \prod_{i=1}^{d} P(x_i|y)P(y)$$

- Assume parametric form for $P(X_i|Y)$ and $P(Y)$
  - Estimate parameters using MLE/MAP and plug in

slide by Aarti Singh & Barnabás Póczos

Gaussian Naïve Bayes (GNB)

- There are several distributions that can lead to a linear boundary.
- As an example, consider Gaussian Naïve Bayes:

$$Y \sim \mathrm{Bernoulli}(\pi)$$

$$P(X_i|Y=y) = \frac{1}{\sqrt{2\pi\sigma_{i,y}^2}} \, e^{\frac{-(X_i-\mu_{i,y})^2}{2\sigma_{i,y}^2}}$$

Gaussian class conditional densities

- What if we assume variance is independent of class, i.e. $\sigma_{i,0}^2 = \sigma_{i,1}^2$

slide by Aarti Singh & Barnabás Póczos

GNB with equal variance is a Linear Classifier!

$$P(X_i|Y=y) = \frac{1}{\sqrt{2\pi\sigma_i^2}}\, e^{\frac{-(X_i-\mu_{i,y})^2}{2\sigma_i^2}}$$

Decision boundary:

$$\prod_{i=1}^{d} P(X_i|Y=0)P(Y=0) = \prod_{i=1}^{d} P(X_i|Y=1)P(Y=1)$$

slide by Aarti Singh & Barnabás Póczos

GNB with equal variance is a Linear Classifier!

$$P(X_i|Y=y) = \frac{1}{\sqrt{2\pi\sigma_i^2}}\, e^{\frac{-(X_i-\mu_{i,y})^2}{2\sigma_i^2}}$$

Decision boundary:

$$\prod_{i=1}^{d} P(X_i|Y=0)P(Y=0) = \prod_{i=1}^{d} P(X_i|Y=1)P(Y=1)$$

$$\log \frac{P(Y=0)\prod_{i=1}^{d} P(X_i|Y=0)}{P(Y=1)\prod_{i=1}^{d} P(X_i|Y=1)} = \log \frac{1-\pi}{\pi} + \sum_{i=1}^{d} \log \frac{P(X_i|Y=0)}{P(X_i|Y=1)}$$

slide by Aarti Singh & Barnabás Póczos

GNB with equal variance is a Linear Classifier!

$$P(X_i|Y=y) = \frac{1}{\sqrt{2\pi\sigma_i^2}}\, e^{\frac{-(X_i-\mu_{i,y})^2}{2\sigma_i^2}}$$

Decision boundary:

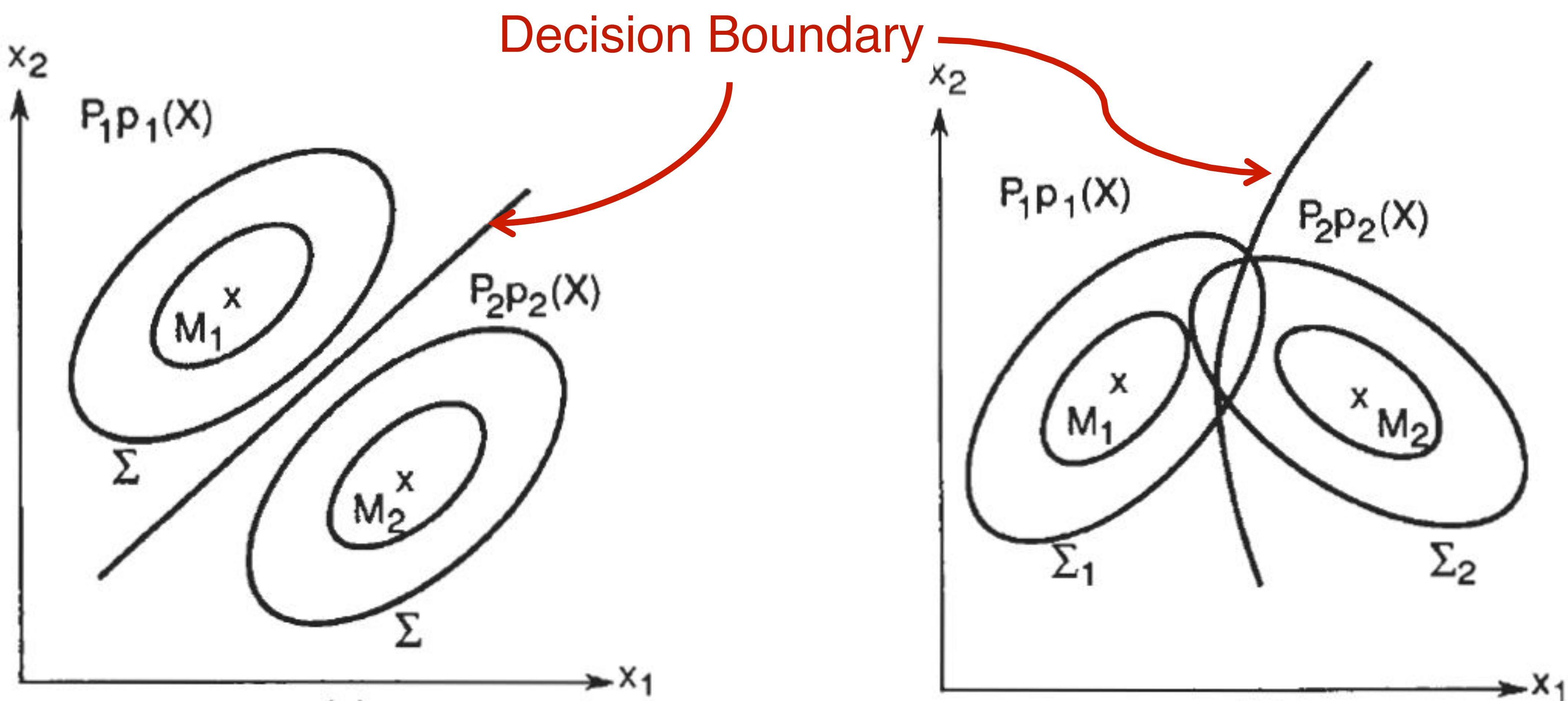
$$\prod_{i=1}^{d} P(X_i|Y=0)P(Y=0) = \prod_{i=1}^{d} P(X_i|Y=1)P(Y=1)$$

$$\log \frac{P(Y=0)\prod_{i=1}^{d} P(X_i|Y=0)}{P(Y=1)\prod_{i=1}^{d} P(X_i|Y=1)} = \log\frac{1-\pi}{\pi} + \sum_{i=1}^{d} \log\frac{P(X_i|Y=0)}{P(X_i|Y=1)}$$

$$= \underbrace{\log\frac{1-\pi}{\pi} + \sum_i \frac{\mu_{i,1}^2-\mu_{i,0}^2}{2\sigma_i^2}}_{\text{Constant term}} + \underbrace{\sum_i \frac{\mu_{i,0}-\mu_{i,1}}{\sigma_i^2}X_i}_{\text{First-order term}} =: w_0 + \sum_i w_i X_i$$

slide by Aarti Singh & Barnabás Póczos

Gaussian Naive Bayes (GNB)



$$
\begin{aligned}
X &= (x_1, x_2) \\
P_1 &= P(Y=0) \\
P_2 &= P(Y=1) \\
p_1(X) &= p(X|Y=0) \sim \mathcal{N}(M_1, \Sigma_1) \\
p_2(X) &= p(X|Y=1) \sim \mathcal{N}(M_2, \Sigma_2)
\end{aligned}
$$

slide by Aarti Singh & Barnabás Póczos

Generative vs. Discriminative Classifiers

- Generative classifiers (e.g. Naïve Bayes)
  - Assume some functional form for P(X,Y) (or P(X|Y) and P(Y))
  - Estimate parameters of P(X|Y), P(Y) directly from training data

- But arg max_Y P(X|Y) P(Y) = arg max_Y P(Y|X)

- Why not learn P(Y|X) directly? Or better yet, why not learn the decision boundary directly?

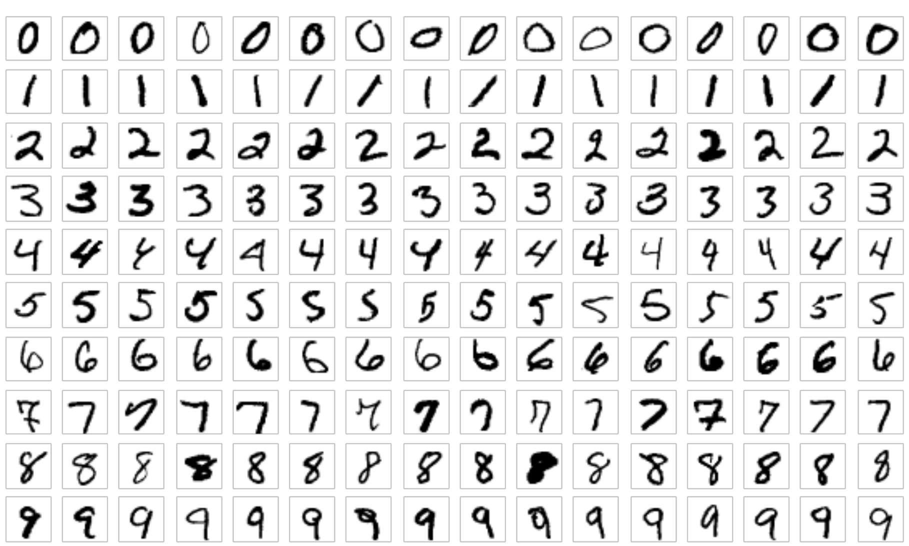
- Discriminative classifiers (e.g. Logistic Regression)
  - Assume some functional form for P(Y|X) or for the decision boundary
  - Estimate parameters of P(Y|X) directly from training data

slide by Aarti Singh & Barnabás Póczos

Regression vs. Classification

- Regression estimates a continuous value
- Classification predicts a discrete category

MNIST: classify hand-written digits
(10 classes)



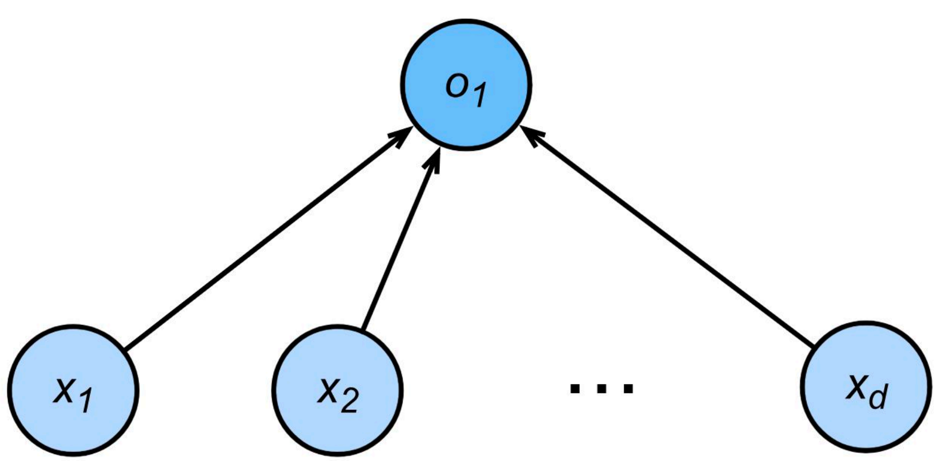
ImageNet: classify nature objects
(1000 classes)

slide by Alex Smola and Mu Li

From Regression to Multi-class Classification

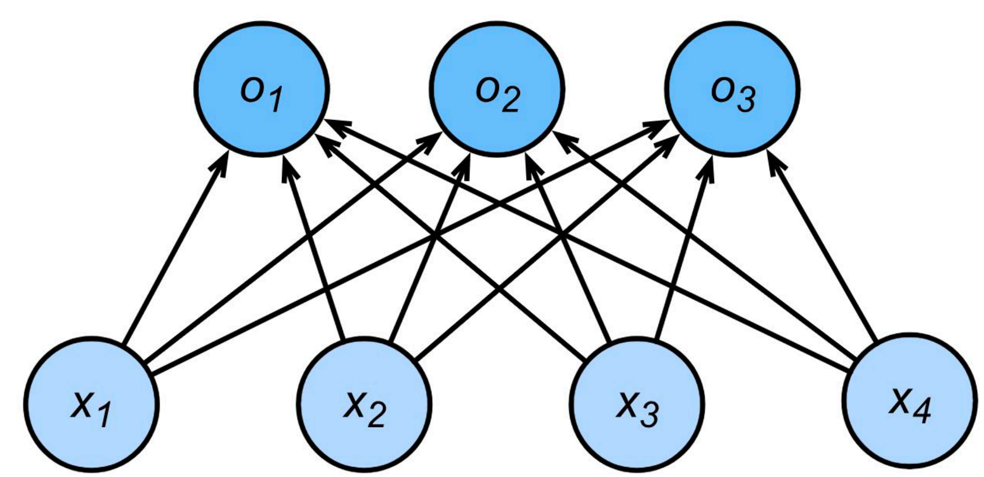
Regression

- Single continuous output
- Natural scale in
- Loss given e.g. in terms of difference $y - f(x)$



Classification

- Multiple classes, typically multiple outputs
- Score *should* reflect confidence …



slide by Alex Smola and Mu Li

From Regression to Multi-class Classification

Square Loss

- One hot encoding per class

$$\mathbf{y} = [y_1, y_2, \ldots, y_n]^\top$$

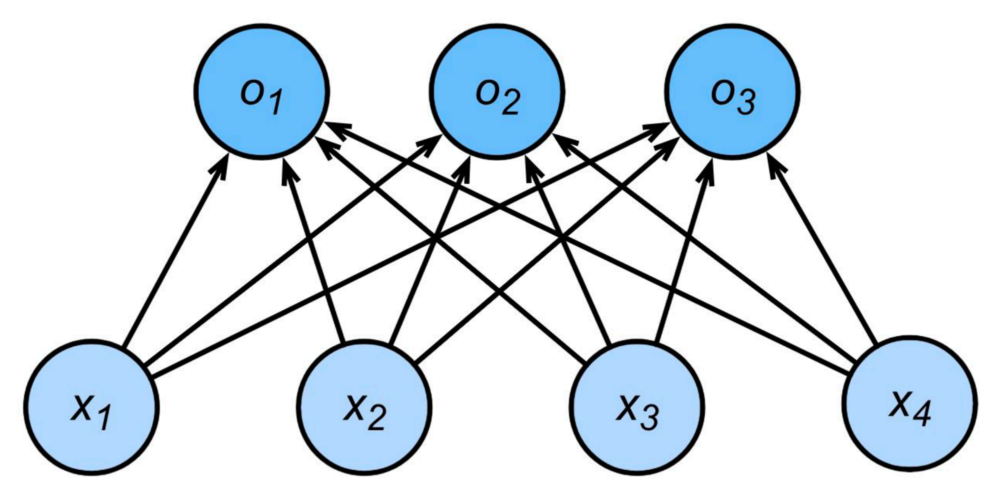
$$y_i = \begin{cases} 1 \text{ if } i = y \\ 0 \text{ otherwise} \end{cases}$$

- Train with squared loss
- Largest output wins

$$\hat{y} = \underset{i}{\operatorname{argmax}}\, o_i$$

Classification

- Multiple classes, typically multiple outputs
- Score *should* reflect confidence …



slide by Alex Smola and Mu Li

From Regression to Multi-class Classification

Uncalibrated Scale

- One output per class
- Largest output wins

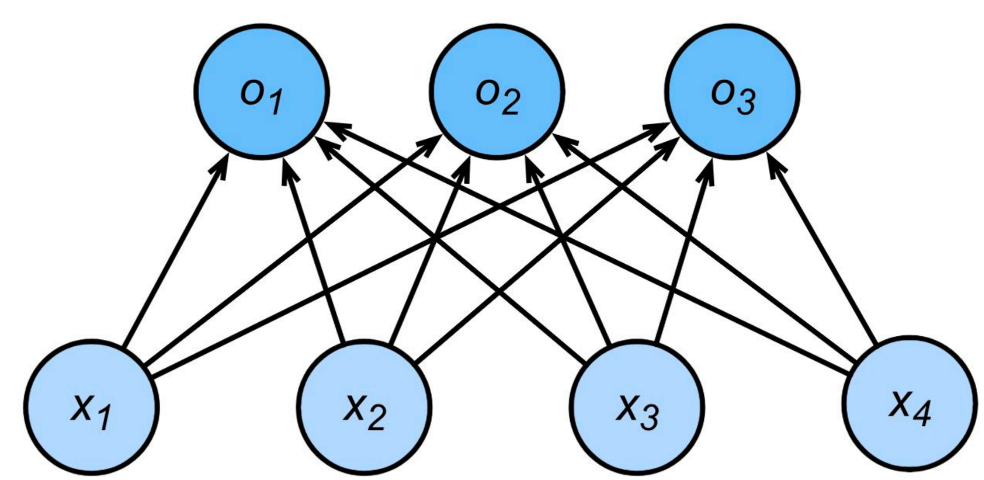
$$\hat{y} = \underset{i}{\operatorname{argmax}}\, o_i$$

- Want that correct class is recognized confidently (**large margin**)

$$o_y - o_i \geq \Delta(y, i)$$

Classification

- Multiple classes, typically multiple outputs
- Score *should* reflect confidence …



slide by Alex Smola and Mu Li

From Regression to Multi-class Classification

Calibrated Scale

- Output matches probabilities (nonnegative, sums to 1)

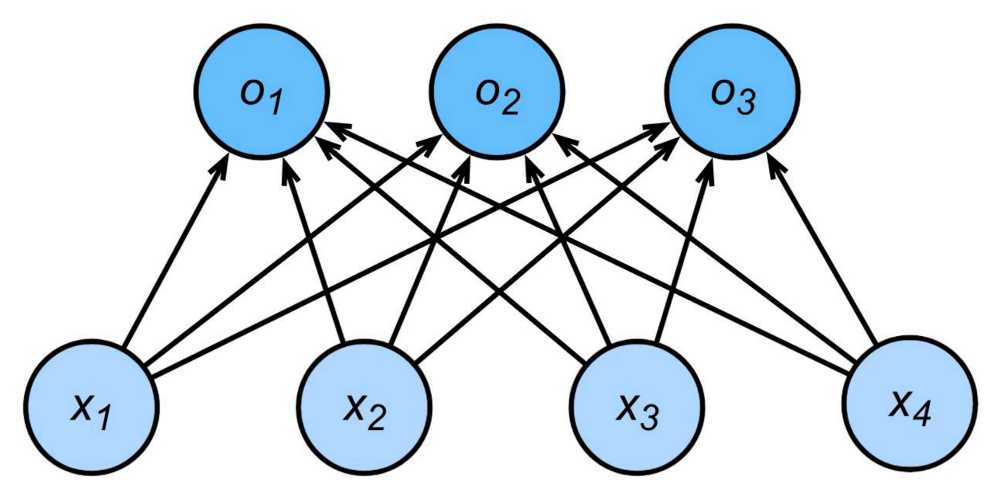
$$p(y\,|\,o) = \mathrm{softmax}(o) = \frac{\exp(o_y)}{\sum_i \exp(o_i)}$$

- Negative log-likelihood

$$-\log p(y\,|\,o) = \log \sum_i \exp(o_i) - o_y$$

Classification

- Multiple classes, typically multiple outputs
- Score *should* reflect confidence …



slide by Alex Smola and Mu Li

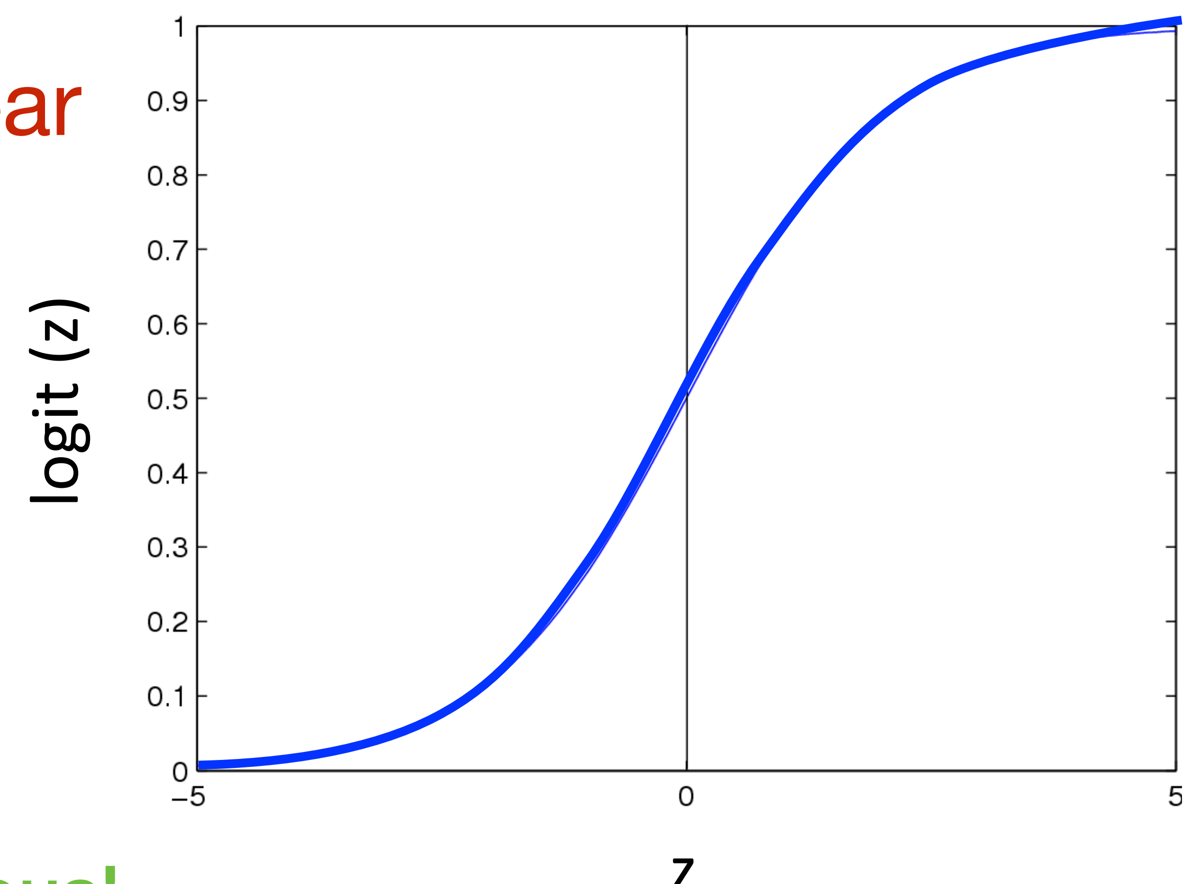
Logistic Regression

Assumes the following functional form for P(Y|X):

$$P(Y=1|X) = \frac{1}{1+\exp(w_0 + \sum_i w_i X_i)}$$

Logistic function applied to linear function of the data

Logistic function (or Sigmoid): $\frac{1}{1+exp(-z)}$



Features can be discrete or continuous!

slide by Aarti Singh & Barnabás Póczos

Logistic Regression is a Linear Classifier!

Assumes the following functional form for P(Y|X):

$$P(Y=1|X) = \frac{1}{1+\exp(w_0 + \sum_i w_i X_i)}$$

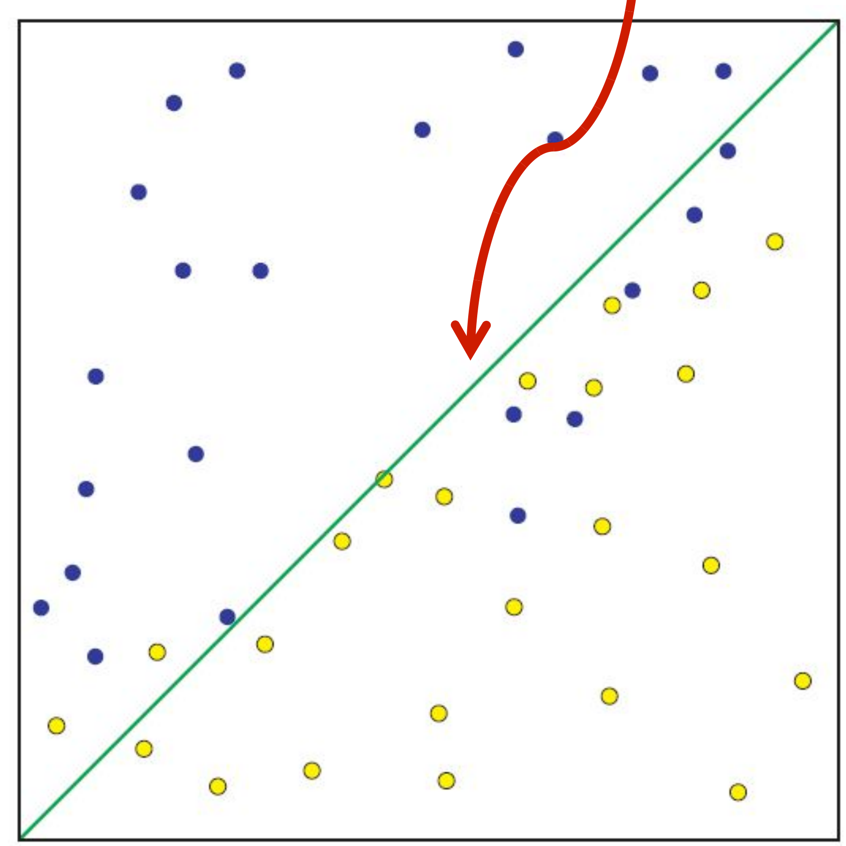
$$w_0 + \sum_i w_i X_i = 0$$

Decision boundary:

$$P(Y=0|X) \overset{0}{\underset{1}{\gtrless}} P(Y=1|X)$$

$$w_0 + \sum_i w_i X_i \overset{0}{\underset{1}{\gtrless}} 0$$

(Linear Decision Boundary)



slide by Aarti Singh & Barnabás Póczos

Logistic Regression is a Linear Classifier!

Assumes the following functional form for P(Y|X):

$$P(Y=1|X) = \frac{1}{1+\exp(w_0 + \sum_i w_i X_i)}$$

$$\Rightarrow P(Y=0|X) = \frac{\exp(w_0 + \sum_i w_i X_i)}{1+\exp(w_0 + \sum_i w_i X_i)}$$

$$\Rightarrow \frac{P(Y=0|X)}{P(Y=1|X)} = \exp(w_0 + \sum_i w_i X_i) \quad \underset{1}{\overset{0}{\gtrless}} \quad 1$$

$$\Rightarrow \boxed{w_0 + \sum_i w_i X_i \quad \underset{1}{\overset{0}{\gtrless}} \quad 0}$$

slide by Aarti Singh & Barnabás Póczos

Logistic Regression for more than 2 classes

- Logistic regression in more general case, where $Y \in \{y_1,\dots,y_K\}$

for k<K

$$P(Y = y_k|X) = \frac{\exp(w_{k0} + \sum_{i=1}^{d} w_{ki}X_i)}{1 + \sum_{j=1}^{K-1} \exp(w_{j0} + \sum_{i=1}^{d} w_{ji}X_i)}$$

for k=K (normalization, so no weights for this class)

$$P(Y = y_K|X) = \frac{1}{1 + \sum_{j=1}^{K-1} \exp(w_{j0} + \sum_{i=1}^{d} w_{ji}X_i)}$$

slide by Aarti Singh & Barnabás Póczos

Training Logistic Regression

We'll focus on binary classification:

$$P(Y=0|\mathbf{X},\mathbf{w}) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$

$$P(Y=1|\mathbf{X},\mathbf{w}) = \frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)}$$

How to learn the parameters w_0, w_1, …, w_d?

Training Data $\{(X^{(j)}, Y^{(j)})\}_{j=1}^{n}$ $X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})$

Maximum Likelihood Estimates

$$\widehat{\mathbf{w}}_{MLE} = \arg\max_{\mathbf{w}} \prod_{j=1}^{n} P(X^{(j)}, Y^{(j)} \mid \mathbf{w})$$

slide by Aarti Singh & Barnabás Póczos

Training Logistic Regression

We'll focus on binary classification:

$$P(Y=0|\mathbf{X},\mathbf{w}) = \frac{1}{1+exp(w_0+\sum_i w_i X_i)}$$

$$P(Y=1|\mathbf{X},\mathbf{w}) = \frac{exp(w_0+\sum_i w_i X_i)}{1+exp(w_0+\sum_i w_i X_i)}$$

How to learn the parameters w_0, w_1, …, w_d?

Training Data $\{(X^{(j)}, Y^{(j)})\}_{j=1}^n$ $\quad X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})$

Maximum Likelihood Estimates

$$\widehat{\mathbf{w}}_{MLE} = \arg\max_{\mathbf{w}} \prod_{j=1}^n P(X^{(j)}, Y^{(j)} \mid \mathbf{w})$$

But there is a problem…

Don't have a model for P(X) or P(X|Y) — only for P(Y|X)

slide by Aarti Singh & Barnabás Póczos

Training Logistic Regression

How to learn the parameters $w_0, w_1, \ldots, w_d$?

Training Data $\{(X^{(j)}, Y^{(j)})\}_{j=1}^{n}$ $\quad X^{(j)} = (X_1^{(j)}, \ldots, X_d^{(j)})$

Maximum (Conditional) Likelihood Estimates

$$\widehat{\mathbf{w}}_{MCLE} = \arg\max_{\mathbf{w}} \prod_{j=1}^{n} P(Y^{(j)} \mid X^{(j)}, \mathbf{w})$$

slide by Aarti Singh & Barnabás Póczos

Discriminative philosophy — Don't waste effort learning P(X), focus on P(Y|X) — that's all that matters for classification!

Expressing Conditional log Likelihood

$$l(W) = \sum_l Y^l \ln P(Y^l = 1|X^l, W) + (1 - Y^l) \ln P(Y^l = 0|X^l, W)$$

$$P(Y = 0|X) = \frac{1}{1 + \exp(w_0 + \sum_{i=1}^n w_i X_i)}$$

$$P(Y = 1|X) = \frac{\exp(w_0 + \sum_{i=1}^n w_i X_i)}{1 + \exp(w_0 + \sum_{i=1}^n w_i X_i)}$$

Y can take only values 0 or 1, so only one of the two terms in the expression will be non-zero for any given Y^l

slide by Aarti Singh & Barnabás Póczos

Expressing Conditional log Likelihood

$$l(W) \;=\; \sum_l Y^l \ln P(Y^l = 1|X^l, W) + (1 - Y^l) \ln P(Y^l = 0|X^l, W)$$

slide by Aarti Singh & Barnabás Póczos

Expressing Conditional log Likelihood

$$P(Y=0|X) = \frac{1}{1+\exp(w_0+\sum_{i=1}^n w_i X_i)}$$

$$P(Y=1|X) = \frac{\exp(w_0+\sum_{i=1}^n w_i X_i)}{1+\exp(w_0+\sum_{i=1}^n w_i X_i)}$$

$$\begin{aligned} l(W) &= \sum_l Y^l \ln P(Y^l=1|X^l,W) + (1-Y^l)\ln P(Y^l=0|X^l,W) \\ &= \sum_l Y^l \ln \frac{P(Y^l=1|X^l,W)}{P(Y^l=0|X^l,W)} + \ln P(Y^l=0|X^l,W) \end{aligned}$$

slide by Aarti Singh & Barnabás Póczos

Expressing Conditional log Likelihood

$$P(Y=0|X) = \frac{1}{1+\exp(w_0+\sum_{i=1}^{n} w_i X_i)}$$

$$P(Y=1|X) = \frac{\exp(w_0+\sum_{i=1}^{n} w_i X_i)}{1+\exp(w_0+\sum_{i=1}^{n} w_i X_i)}$$

$$\begin{aligned}
l(W) &= \sum_l Y^l \ln P(Y^l=1|X^l,W) + (1-Y^l)\ln P(Y^l=0|X^l,W) \\
&= \sum_l Y^l \ln \frac{P(Y^l=1|X^l,W)}{P(Y^l=0|X^l,W)} + \ln P(Y^l=0|X^l,W) \\
&= \sum_l Y^l (w_0 + \sum_i^n w_i X_i^l) - \ln(1+\exp(w_0+\sum_i^n w_i X_i^l))
\end{aligned}$$

slide by Aarti Singh & Barnabás Póczos

Maximizing Conditional log Likelihood

$$\max_{\mathbf{w}} \; l(\mathbf{w}) \equiv \ln \prod_j P(y^j | \mathbf{x}^j, \mathbf{w})$$

$$= \sum_j y^j (w_0 + \sum_i^d w_i x_i^j) - \ln(1 + exp(w_0 + \sum_i^d w_i x_i^j))$$

Bad news: no closed-form solution to maximize $l(\mathbf{w})$

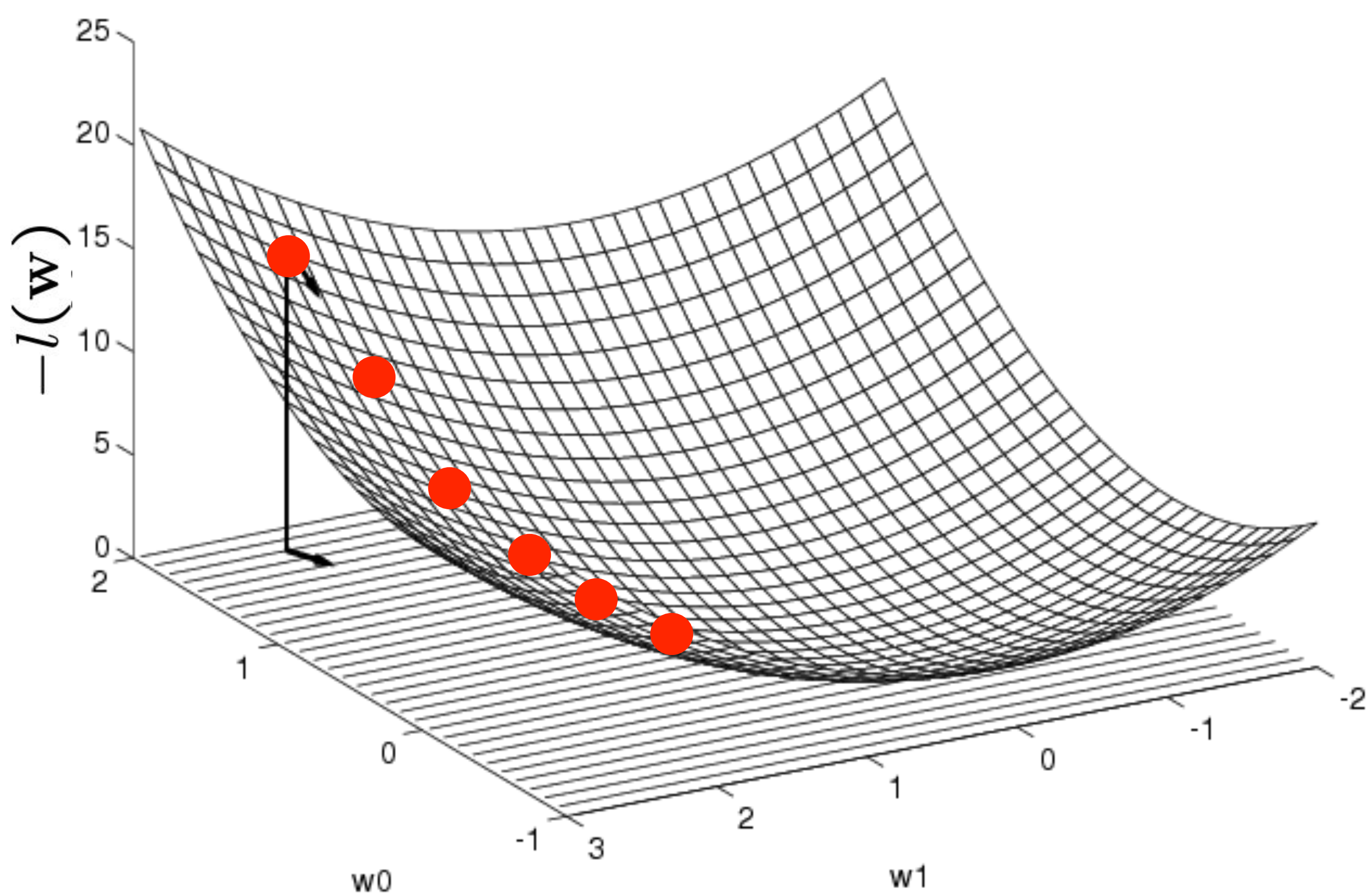
Good news: $l(\mathbf{w})$ is concave function of w! concave functions easy to optimize (unique maximum)

slide by Aarti Singh & Barnabás Póczos

Optimizing concave/convex functions

- Conditional likelihood for Logistic Regression is concave
- Maximum of a concave function = minimum of a convex function

Gradient Ascent (concave)/ Gradient Descent (convex)



Gradient:

$$\nabla_{\mathbf{w}} l(\mathbf{w}) = [\frac{\partial l(\mathbf{w})}{\partial w_0}, \ldots, \frac{\partial l(\mathbf{w})}{\partial w_n}]'$$

Update rule:

Learning rate, η>0

$$\Delta \mathbf{w} = \eta \nabla_{\mathbf{w}} l(\mathbf{w})$$

$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \left. \frac{\partial l(\mathbf{w})}{\partial w_i} \right|_t$$

slide by Aarti Singh & Barnabás Póczos

Gradient Ascent for Logistic Regression

Gradient ascent algorithm: iterate until change < ε

$$w_0^{(t+1)} \leftarrow w_0^{(t)} + \eta \sum_j [y^j - \widehat{P}(Y^j = 1 \mid \mathbf{x}^j, \mathbf{w}^{(t)})]$$

For i-1,…,d,

$$w_i^{(t+1)} \leftarrow w_i^{(t)} + \eta \sum_j x_i^j [y^j - \underbrace{\widehat{P}(Y^j = 1 \mid \mathbf{x}^j, \mathbf{w}^{(t)})}]$$

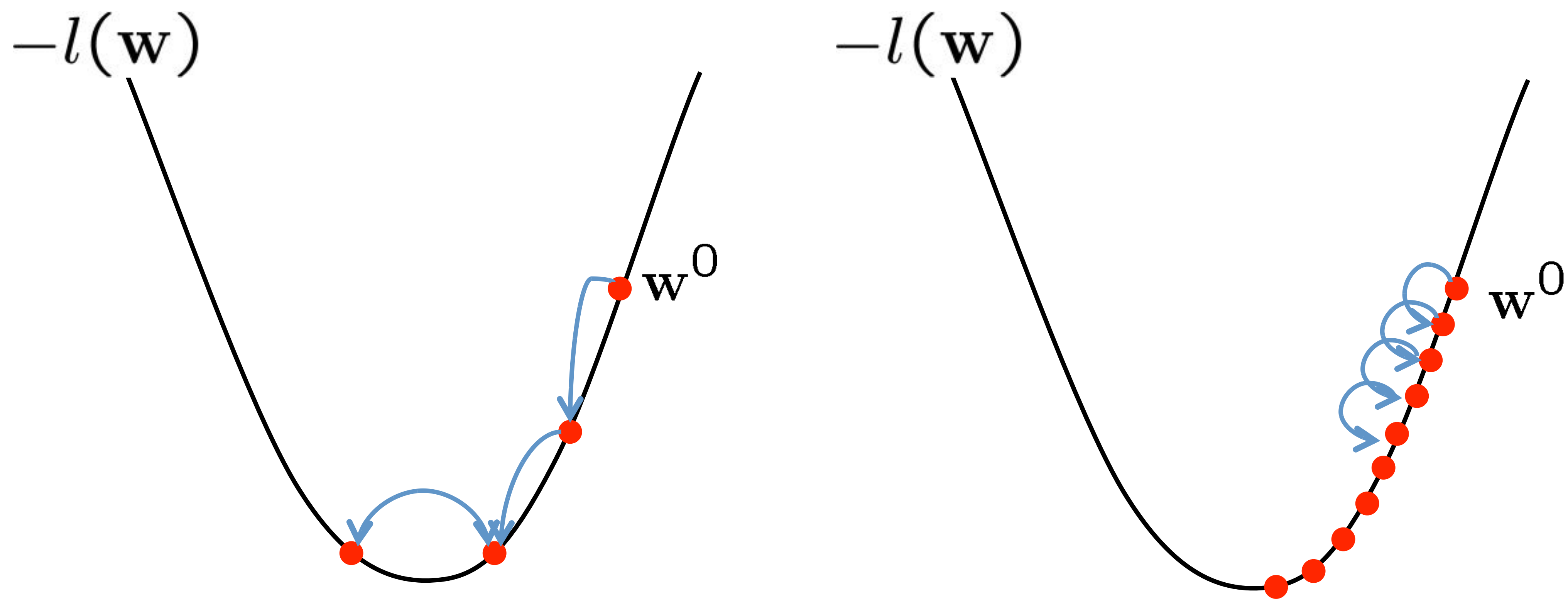
repeat

Predict what current weight thinks label Y should be

- Gradient ascent is simplest of optimization approaches
  - e.g. Newton method, Conjugate gradient ascent, IRLS (see Bishop 4.3.3)

slide by Aarti Singh & Barnabás Póczos

Effect of step-size η



Large η → Fast convergence but larger residual error
Also possible oscillations

Small η → Slow convergence but small residual error

Naïve Bayes vs. Logistic Regression

Set of Gaussian Naïve Bayes parameters (feature variance independent of class label)

Set of Logistic Regression parameters

- Representation equivalence
  - **But only in a special case!!!** (GNB with class-independent variances)
- But what's the difference???

slide by Aarti Singh & Barnabás Póczos

Naïve Bayes vs. Logistic Regression

Set of Gaussian Naïve Bayes parameters (feature variance independent of class label) ⟷ **Set of Logistic Regression parameters**

- Representation equivalence
  - **But only in a special case!!!** (GNB with class-independent variances)
- But what's the difference???
- **LR makes no assumption about** P(**X**|Y) **in learning!!!**
- Loss function!!!
  - Optimize different functions! Obtain different solutions

Naïve Bayes vs. Logistic Regression

Consider Y Boolean, X_i continuous $X=<X_1 \ldots X_d>$

Number of parameters:

- NB: 4d+1 $\quad \pi, (\mu_{1,y}, \mu_{2,y}, \ldots, \mu_{d,y}), (\sigma^2_{1,y}, \sigma^2_{2,y}, \ldots, \sigma^2_{d,y})$ y=0,1
- LR: d+1 $\quad w_0, w_1, \ldots, w_d$

Estimation method:

- NB parameter estimates are uncoupled
- LR parameter estimates are coupled

Generative vs. Discriminative

[Ng & Jordan, NIPS 2001]

Given **infinite data** (asymptotically),

If conditional independence assumption holds, Discriminative and generative NB perform similar.

$$\epsilon_{\mathrm{Dis},\infty} \sim \epsilon_{\mathrm{Gen},\infty}$$

If conditional independence assumption does NOT holds, Discriminative outperforms generative NB.

$$\epsilon_{\mathrm{Dis},\infty} < \epsilon_{\mathrm{Gen},\infty}$$

slide by Aarti Singh & Barnabás Póczos

Generative vs. Discriminative

[Ng & Jordan, NIPS 2001]

Given **finite data** (n data points, d features),

$$\epsilon_{\mathrm{Dis},n} \leq \epsilon_{\mathrm{Dis},\infty} + O\left(\sqrt{\frac{d}{n}}\right)$$

$$\epsilon_{\mathrm{Gen},n} \leq \epsilon_{\mathrm{Gen},\infty} + O\left(\sqrt{\frac{\log d}{n}}\right)$$

Naïve Bayes (generative) requires n = O(log d) to converge to its asymptotic error, whereas Logistic regression (discriminative) requires n = O(d).

Why? “Independent class conditional densities”

- parameter estimates not coupled – each parameter is learnt independently, not jointly, from training data.

slide by Aarti Singh & Barnabás Póczos

Naïve Bayes vs. Logistic Regression

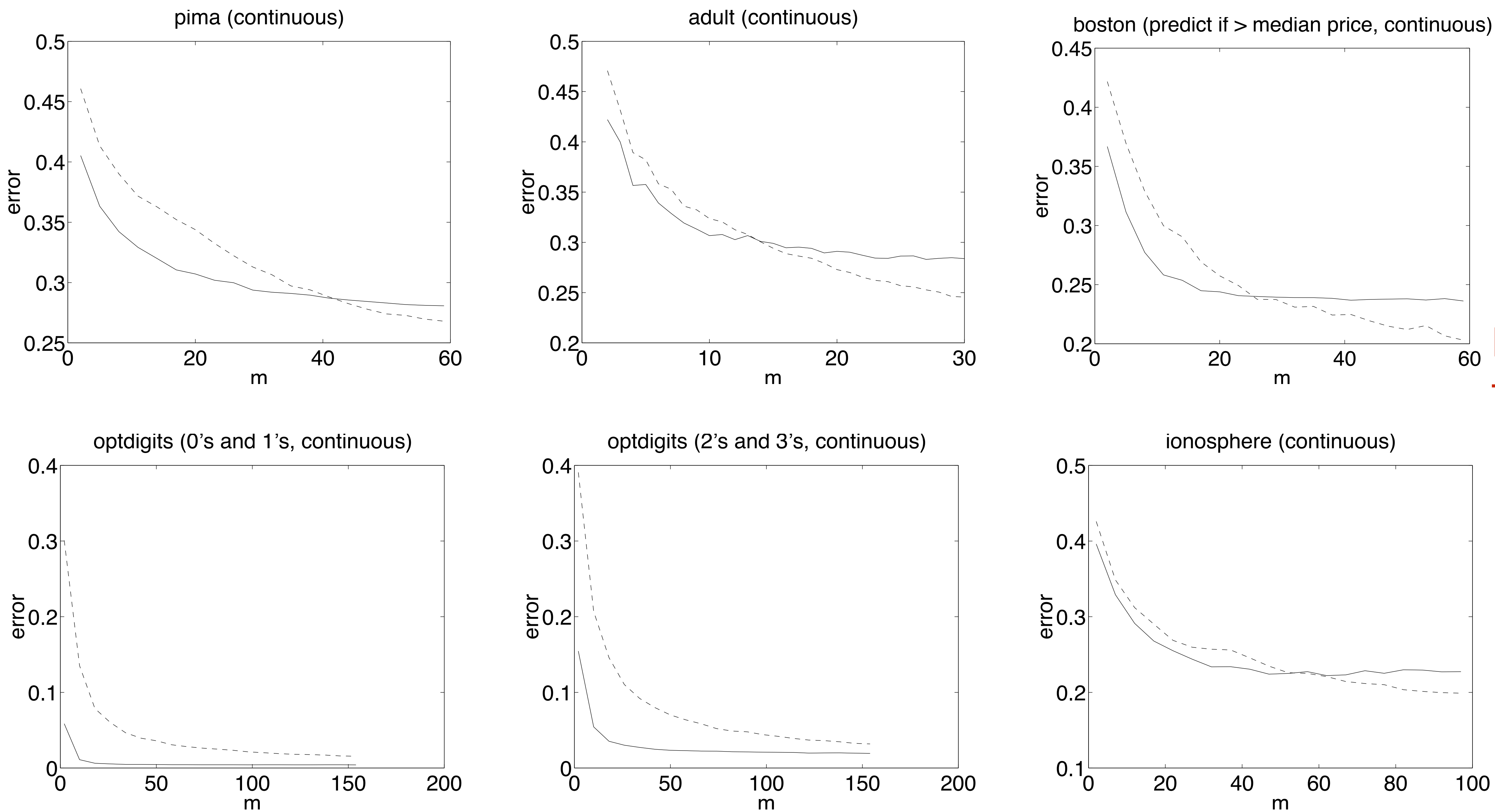
Verdict

Both learn a linear decision boundary.
Naïve Bayes makes more restrictive assumptions and has higher asymptotic error,
BUT
converges faster to its less accurate asymptotic error.

slide by Aarti Singh & Barnabás Póczos

Experimental Comparison (Ng-Jordan'01)

UCI Machine Learning Repository 15 datasets, 8 continuous features, 7 discrete features



More in the paper...

—— Naïve Bayes ----- Logistic Regression

slide by Aarti Singh & Barnabás Póczos

What you should know

- LR is a linear classifier
  - decision rule is a hyperplane
- LR optimized by maximizing conditional likelihood
  - no closed-form solution
  - concave ! global optimum with gradient ascent
- Gaussian Naïve Bayes with class-independent variances representationally equivalent to LR
  - Solution differs because of objective (loss) function
- In general, NB and LR make different assumptions
  - NB: Features independent given class! assumption on P(**X**|Y)
  - LR: Functional form of P(Y|**X**), no assumption on P(**X**|Y)
- Convergence rates
  - GNB (usually) needs less data
  - LR (usually) gets to better solutions in the limit

Next Lecture:

Linear Discriminant Functions

Perceptron