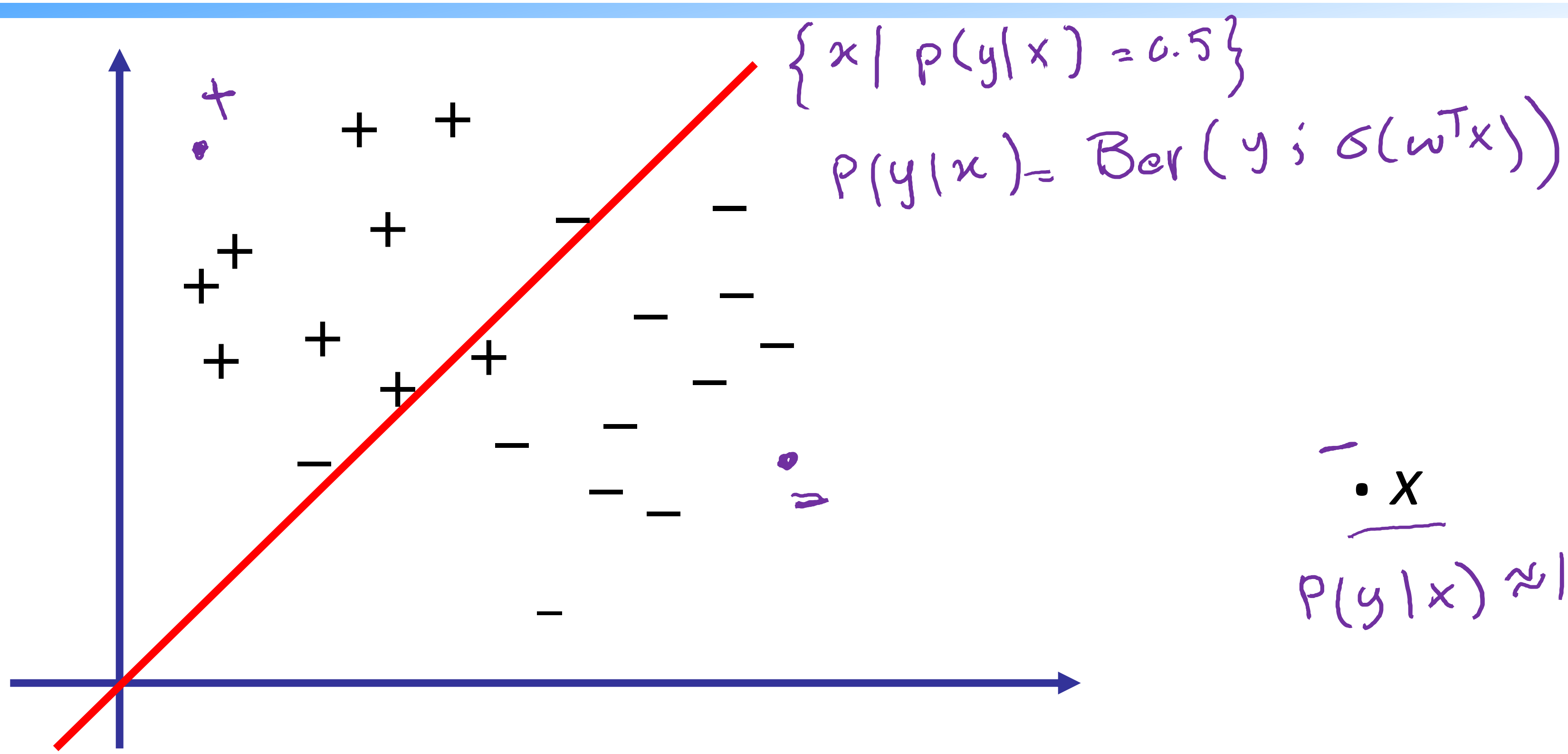
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Introduction to Machine Learning

Discriminative vs. Generative Modeling

Prof. Andreas Krause
Learning and Adaptive Systems (las.ethz.ch)

Motivating example: Logistic regression



- What will logistic regression predict for data point *x*?
- Logistic regression can be overconfident about labels for outliers

Discriminative modeling

- So far, we have considered learning methods that estimate conditional distributions

$$P(y \mid \mathbf{x})$$

- **Examples**: Linear regression, logistic regression, etc.

- Such models *do not* attempt to model $P(\mathbf{x})$
- Thus, they will not be able to detect outliers (i.e., „unusual“ points for which *P(**x**)* is very small)

Discriminative vs. Generative models

- Discriminative models aim to estimate

$$P(y \mid \mathbf{x})$$

- Generative models aim to estimate joint distribution

$$P(y, \mathbf{x})$$

- Can derive conditional from joint distribution, but not vice versa!

$$P(y, x) \rightsquigarrow P(y \mid x) = \frac{P(x, y)}{P(x)}$$

$$P(x) \to \sum_{y'} P(x, y')$$

Typical approach to generative modeling

$P(x, y) = P(x \mid y)\, P(y)$
(chain rule)
$= P(y \mid x)\, P(x)$

1. Estimate prior on labels $P(y)$

2. Estimate conditional distribution for each class *y* $P(\mathbf{x} \mid y)$

3. Obtain predictive distribution using Bayes' rule:

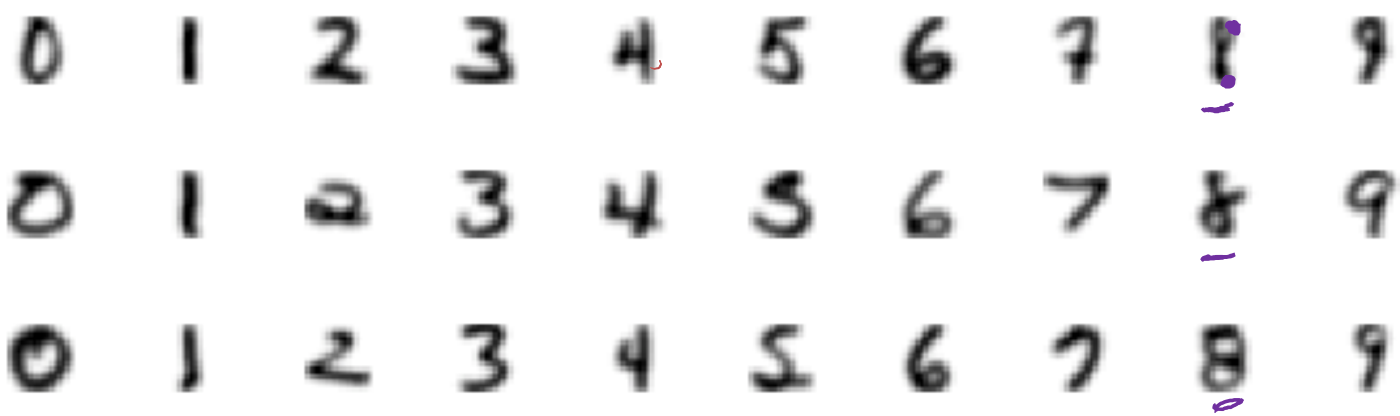
$$P(y \mid \mathbf{x}) = \frac{1}{Z} \underbrace{P(y) P(\mathbf{x} \mid y)}_{P(y, x)}$$

$P(x)$

A note on generative modeling

- Generative modeling attempts to infer the process, according to which examples are generated $P(x, y)$
- First generate class label $P(y)$
- Then, generate features given class $P(\mathbf{x} \mid y)$

y (label) 0 1 2 3 4 5 6 7 8 9



x (vector of pixel intensities)

Example: Naive Bayes Model

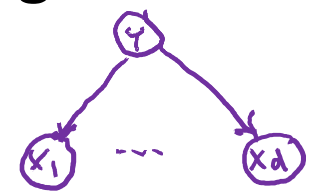
- Model class label as generated from categorical variable

$$P(Y = y) = p_y \qquad y \in \mathcal{Y} = \{1, \ldots, c\}$$

$$\forall y = 1, \ldots, c \quad p_y \geq 0 , \quad \sum_{y=1}^{c} p_y = 1$$

- Model features as conditionally independent given Y

$$P(X_1, \ldots, X_d \mid Y) = \prod_{i=1}^{d} P(X_i \mid Y)$$



$$P(X_1 = x_1, \ldots, X_d = x_d \mid Y = y) = \prod_{i=1}^{d} P(X_i = x_i \mid Y = y)$$

- I.e., given class label, each feature is „generated" independently of the other features.
- Need to still specify feature distributions $P(X_i \mid Y)$

Example: Gaussian *Naive* Bayes classifiers

- Model class label as generated from categorical variable

$$P(Y = y) = p_y \qquad\qquad y \in \mathcal{Y} = \{1, \ldots, c\}$$

- Model features by (conditionally) independent Gaussians

$$P(x_i \mid y) = \mathcal{N}(x_i \mid \mu_{y,i}, \sigma^2_{y,i})$$

depend on class y and feature i

$i \in \{1, \ldots, d\}$

- How do we estimate the parameters?

Maximum Likelihood Estimation for P(y)

$y = \{-1, +1\}$ $\quad P(Y=+1) = p \Rightarrow P(Y=-1) = 1-p$

$D = \{(x_1, y_1), \cdots, (x_n, y_n)\}$

Estimate p using D via MLE:

$$\max_{p'} P(D \mid p') = \prod_{i=1}^{n} p'^{[y_i=1]} (1-p')^{[y_i=0]}$$

$$= p'^{(n_+)} (1-p')^{n_-} \quad \text{where} \begin{cases} n_+ & \text{\# pos. instances in } D \\ n_- & \text{-- " -- negative} \end{cases}$$

$$= \max_{p'} \ell(p') \quad \text{where} \quad \ell(p') = (n_+)\log p' + (n_-)\log(1-p')$$

$$\frac{\partial \ell}{\partial p} = \frac{n_+}{p'} + \frac{n_-}{1-p'} = 0 \Rightarrow$$

$$p' = \frac{n_+}{(n_+) + (n_-)}$$

Maximum Likelihood Estimation for P(x|y)

$$P(x_i \mid y) = \mathcal{N}(x_i ; \mu_{i,y}, \sigma_{i,y})$$

$$D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$$

MLE for $\mu_{i,y}$ and $\sigma_{i,y}$ ...

Deriving decision rules

- Estimate $\hat{P}(y)$ and $\hat{P}(\mathbf{x} \mid y)$
- In order to predict label *y* for new point x, use

$$P(y \mid \mathbf{x}) = \frac{1}{Z} P(y) P(\mathbf{x} \mid y) \qquad Z = \sum_{y} P(y) P(\mathbf{x} \mid y)$$

P(x)

- Predict using Bayesian decision theory.
- E.g., in order to minimize misclassification error, predict

$$y = \arg\max_{y'} P(y' \mid \mathbf{x})$$

$$= \arg\max_{y'} \frac{P(y')P(x \mid y')}{P(x)} = \arg\max_{y'} P(y')P(x \mid y')$$

$$(\log) = \arg\max_{y'}$$

$$NB \leftarrow \log P(y') + \sum_{i=1}^{d} \log(P(x_i \mid y'))$$

Gaussian Naive Bayes Classifiers

- Learning given data $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$
  - MLE for class prior: $\hat{P}(Y = y) = \hat{p}_y = \frac{\text{Count}(Y = y)}{n}$
  - MLE for feature distribution: $\hat{P}(x_i \mid y) = \mathcal{N}(x_i; \hat{\mu}_{y,i}, \sigma^2_{y,i})$

$$\hat{\mu}_{y,i} = \frac{1}{\text{Count(Y=y)}} \sum_{j:y_j=y} \underbrace{x_{j,i}}$$

the value of feature i for instance j (x_j, y_j)

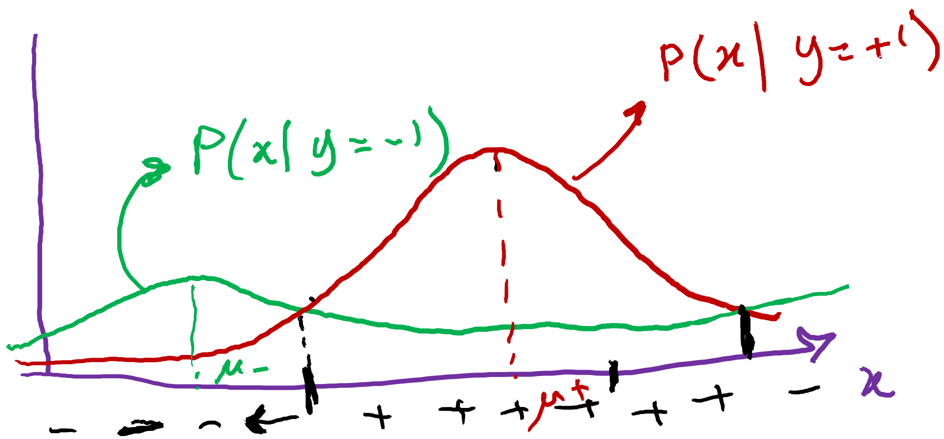
$$\sigma^2_{y,i} = \frac{1}{\text{Count(Y=y)}} \sum_{j:y_j=y} (x_{j,i} - \hat{\mu}_{y,i})^2$$

- Prediction given new point ***x:***

$$y = \arg\max_{y'} \hat{P}(y' \mid \mathbf{x}) = \arg\max_{y'} \hat{P}(y') \prod_{i=1}^{d} \hat{P}(x_i \mid y')$$

Decision boundaries (1D)

$$\begin{cases} d=1, \ 1 \text{ feature } x \\ Y = \{-1, +1\} \\ P(Y=+1) = P(Y=-1) = 0.5 \\ \mu_+ > \mu_-, \quad \sigma_+^2 < \sigma_-^2 \end{cases}$$



$$y = \arg\max_{y'} P(y')P(x|y')$$

$$= \arg\max_{y'} P(x|y')$$

Decision rules for binary classification

- Want to predict $y = \arg\max_{y'} P(y' \mid \mathbf{x})$

- For binary tasks (i.e., c=2, $y \in \{+1, -1\}$), this is equivalent to

$$y = \operatorname{sign}\left(\underbrace{\log \frac{\overbrace{P(Y = 1 \mid \mathbf{x})}^{p}}{P(Y = -1 \mid \mathbf{x})}}_{f(x)}\right)$$

easy to verify that the above gives you $\begin{cases} +1 & \text{if } p > 0.5 \\ -1 & \text{o.w.} \end{cases}$

- The function $f(\mathbf{x}) = \log \frac{P(Y = 1 \mid \mathbf{x})}{P(Y = -1 \mid \mathbf{x})}$

  is called discriminant function

Special case: Gaussian Naive Bayes (c=2)

- Given: $\underbrace{P(Y=1)=.5}_{\text{Assumption 1}}$ and $P(\mathbf{x}\mid y)=\prod_i \mathcal{N}(x_i;\mu_{y,i},\sigma_i^2)$ $\forall y \in Y.\ \sigma_{i,y}^2 = \sigma_i^2$

  (i.e., assume equal class prob., class indep. variance)

- Want: $$f(\mathbf{x})=\log\frac{P(Y=1\mid\mathbf{x})}{P(Y=-1\mid\mathbf{x})}$$

$$f(x) = \log \frac{\left(\cancel{P(Y=1)}\prod_{i=1}^d P(x_i\mid Y=1)\right)/\cancel{P(x)}}{\left(\cancel{P(Y=-1)}\prod_{i=1}^d P(x_i\mid Y=-1)\right)/\cancel{P(x)}} = \log\frac{\prod_{i=1}^d P(x_i\mid Y=1)}{\prod_{i=1}^d P(x_i\mid Y=-1)}$$

$$= \log \frac{\prod_{i=1}^d \cancel{\frac{1}{\sqrt{2\pi\sigma_i^2}}}\exp\left(\frac{-1}{2\sigma_i^2}(x_i-\mu_{+1,i})^2\right)}{\prod_{i=1}^d \cancel{\frac{1}{\sqrt{2\pi\sigma_i^2}}}\exp\left(\frac{-1}{2\sigma_i^2}(x_i-\mu_{-1,i})^2\right)} = -\sum_{i=1}^d \frac{1}{2\sigma_i^2}\left(\quad\right)$$

$$(x_i-\mu_{+1,i})^2-(x_i-\mu_{-1,i})^2$$

$$= \sum_{i=1}^d \underbrace{\frac{1}{\sigma_i^2}(\mu_{+1,i}-\mu_{-1,i})}_{w_i} x_i + \underbrace{\sum_i \frac{1}{2\sigma_i^2}(\mu_{-1,i}^2-\mu_{+1,i}^2)}_{w_0}$$

Special case: GNB (c=2), constant variance

- In case of shared variance, Gaussian Naive Bayes produces linear classifier

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$

Recall $y = \text{sign}(f(x)) \Rightarrow \text{sign}(w^T x + w_0)$

- Hereby: $$w_0 = \log \frac{\hat{p}_+}{1 - \hat{p}_+} + \sum_{i=1}^{d} \frac{\hat{\mu}_{-,i}^2 - \hat{\mu}_{+,i}^2}{2\hat{\sigma}_i^2}$$

$$w_i = \frac{\mu_{+,i} - \mu_{-,i}}{\sigma_i^2}$$

Discriminant function vs class probability

$$f(\mathbf{x}) = \log \frac{P(Y = 1 \mid \mathbf{x})}{P(Y = -1 \mid \mathbf{x})}$$

Define $P(Y=+1 \mid x) := p(x)$

$$\Rightarrow \quad f(x) = \log \frac{p(x)}{1 - p(x)}$$

$$\Rightarrow \exp(f(x)) = \frac{p(x)}{1 - p(x)}$$

$$\Rightarrow \quad p(x) = \frac{\exp(f(x))}{1 + \exp(f(x))} = \frac{1}{1 + \exp(-f(x))} = \sigma(f(x))$$

Demo: Gaussian Naive Bayes

Gaussian NB vs. Logistic regression

- Gaussian NB with shared variance uses discriminant

$$f(\mathbf{x}) = \log \frac{P(Y = 1 \mid \mathbf{x})}{P(Y = -1 \mid \mathbf{x})}$$

where $f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$ and

$$w_0 = \log \frac{\hat{p}_+}{1 - \hat{p}_+} + \sum_{i=1}^{d} \frac{\hat{\mu}_{-,i}^2 - \hat{\mu}_{+,i}^2}{2\hat{\sigma}_i^2}$$

$$w_i = \frac{\mu_{+,i} - \mu_{-,1}}{\sigma_i^2}$$

- The corresponding class distribution

$$P(Y = 1 \mid \mathbf{x}) = \frac{1}{1 + \exp(-f(\mathbf{x}))} = \sigma(\mathbf{w}^T \mathbf{x} + w_0)$$

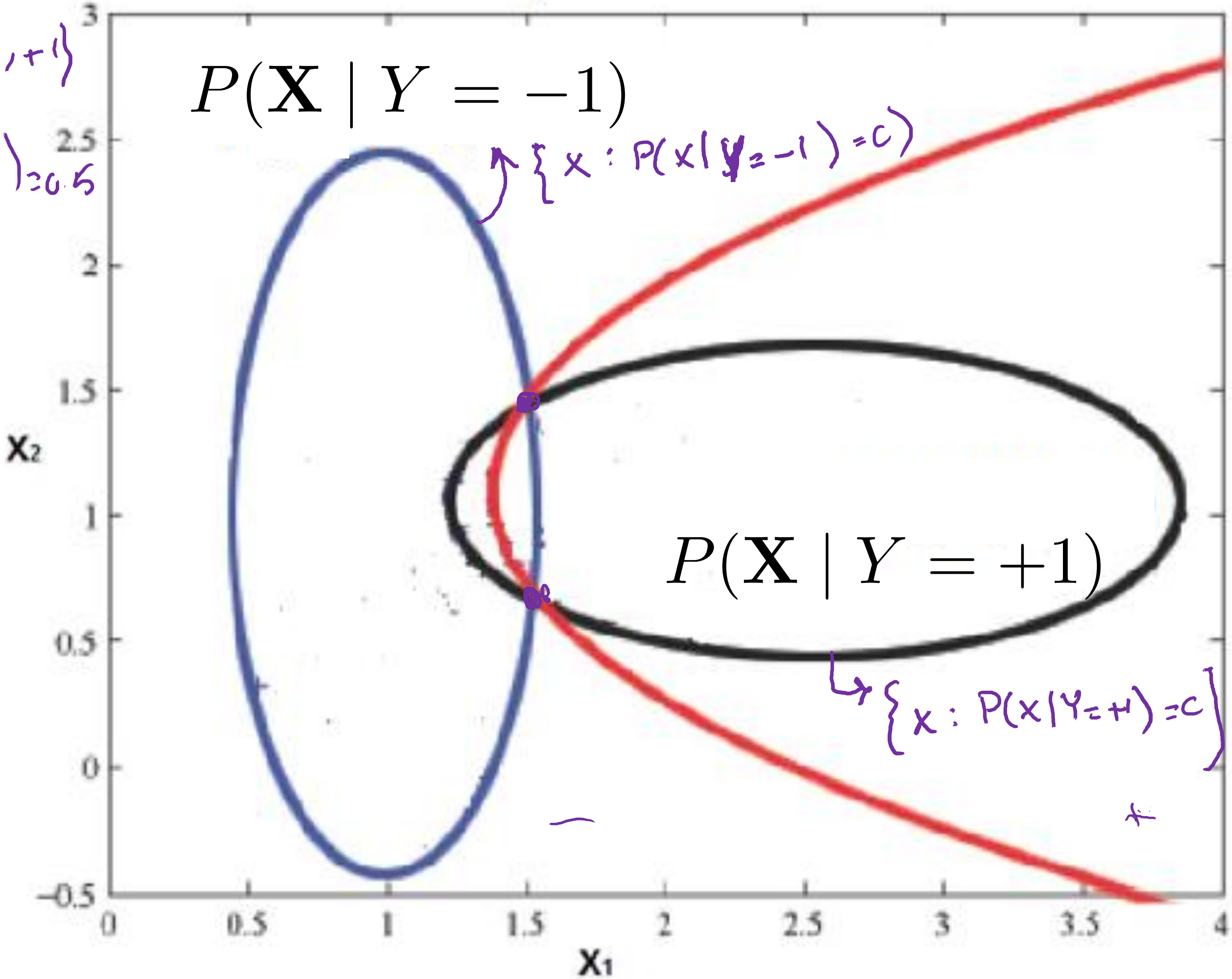
is of the same form as logistic regression!

- If model assumptions are met, GNB will make same predictions as Logistic Regression!

Illustration

$Y = \{-1, +1\}$

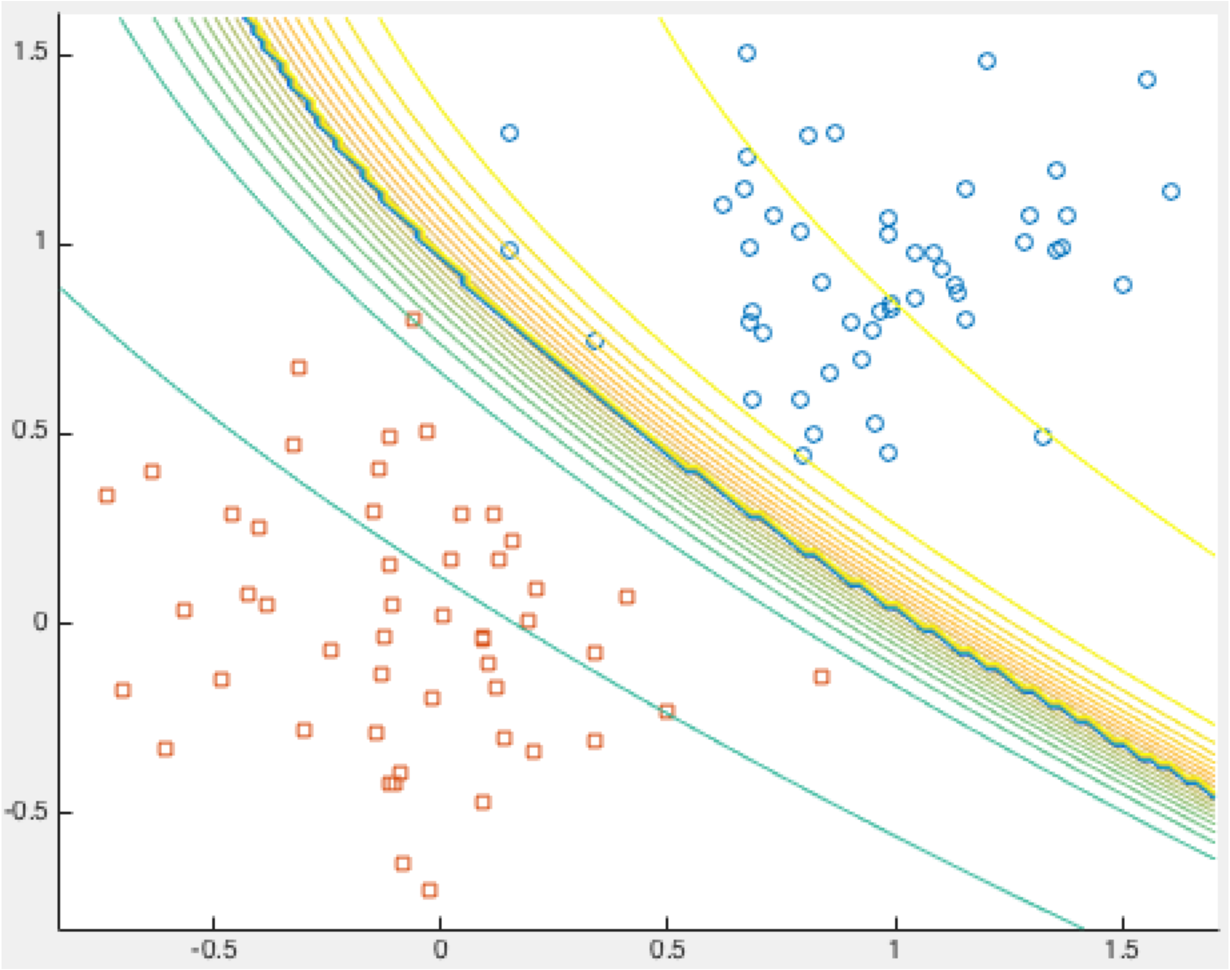
$P(Y=+1) = 0.5$



$f(x) = \log \frac{P(x \mid Y=+1)}{P(x \mid Y=-1)}$

[Zhang] 41

Demo: Gaussian Naive Bayes



Issue with Naive Bayes models

- Conditional independence assumption means that features are generated independently given class label
- If there is (conditional) correlation between class labels, then this assumption is violated

Suppose $P(Y=+1) = P(Y=-1) = 0.5$ $P(X_1 = x \mid Y = y) = \mathcal{N}(x; y, 1)$

$X_2 = X_3 = X_4 = \cdots = X_d = X_1$ (duplicates)

1) GNB that only uses X_1: $f_1(x) = \log \dfrac{P(Y=+1 \mid X_1 = x)}{P(Y=-1 \mid X_1 = x)}$

2) GNB that use $X_1, \ldots, X_d$: $f_2(\vec{x}) = \log \dfrac{P(Y=+1 \mid X_1 = x_1, \ldots, X_d = x_d)}{P(Y=-1 \mid X_1 = x_1, \ldots, X_d = x_d)}$

$= \log \dfrac{P(X_1 = x_1, \ldots, X_d = x_d \mid Y = +1)}{P(X_1 = x_1, \ldots, X_d = x_d \mid Y = -1)}$

$= \log \dfrac{\prod_{i=1}^{d} P(X_i = x_i \mid Y = +1)}{\ldots \mid Y = -1)} = d \cdot f_1(x)$

Issue with Naive Bayes models

- Due to conditional independence assumption, predictions can become overconfident (very close to 1 or 0)
- This might be fine if we care about most likely class only, but not if we want to use probilities for making decisions (e.g., asymmetric losses etc.)

More general: Gaussian Bayes classifiers

- Model class label as generated from categorical variable

$$P(Y = y) = p_y \qquad\qquad y \in \mathcal{Y} = \{1, \ldots, c\}$$

- Model features as generated by multivariate Gaussian

$$P(\mathbf{x} \mid y) = \mathcal{N}(\mathbf{x}; \underbrace{\mu_y}_{\text{mean}}, \underbrace{\Sigma_y}_{\text{covariate matrix}})$$

G Naive B we assumed $\Sigma_y = \begin{pmatrix} \sigma_{y,1}^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_{y,d}^2 \end{pmatrix}$

- How do we estimate the parameters?

MLE for Gaussian Bayes Classifier

- Given data set $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$
- MLE for class label distribution $\hat{P}(Y = y) = \hat{p}_y$

$$\hat{p}_y = \frac{\text{Count}(Y = y)}{n}$$

- MLE for feature distribution $\hat{P}(\mathbf{x} \mid y) = \mathcal{N}(\mathbf{x}; \hat{\mu}_y, \hat{\Sigma}_y)$

$$\hat{\mu}_y = \frac{1}{\text{Count(Y=y)}} \sum_{i: y_i = y} \mathbf{x}_i$$

$$\hat{\Sigma}_y = \frac{1}{\text{Count(Y=y)}} \sum_{i: y_i = y} (\mathbf{x}_i - \hat{\mu}_y)(\mathbf{x}_i - \hat{\mu}_y)^T$$

Discriminant functions for GBCs

- Given: $P(Y=1)=p$ and $P(\mathbf{x}\mid y)=\mathcal{N}(\mathbf{x};\mu_y,\Sigma_y)$
- Want:

$$f(\mathbf{x})=\log\frac{P(Y=1\mid\mathbf{x})}{P(Y=-1\mid\mathbf{x})}$$

- This discriminant function is given by

$$f(\mathbf{x})=\log\frac{p}{1-p}+\frac{1}{2}\left[\log\frac{|\hat{\Sigma}_-|}{|\hat{\Sigma}_+|}+\left((\mathbf{x}-\hat{\mu}_-)^T\hat{\Sigma}_-^{-1}(\mathbf{x}-\hat{\mu}_-)\right)-\left((\mathbf{x}-\hat{\mu}_+)^T\hat{\Sigma}_+^{-1}(\mathbf{x}-\hat{\mu}_+)\right)\right]$$

$Y=\{-1,+1\}\rightarrow$ predict sign $(f(x))$

GBC Demo

Fisher's linear discriminant analysis LDA (c=2)

$y = \{-1, +1\}$

- Suppose we fix *p=.5*
- Further, assume covariances are equal: $\boxed{\hat{\Sigma}_- = \hat{\Sigma}_+ = \hat{\Sigma}}$
- What happens with the discriminant function:

$$f(\mathbf{x}) = \cancel{\log \frac{p}{1-p}} + \frac{1}{2}\left[\cancel{\log \frac{|\hat{\Sigma}_-|}{|\hat{\Sigma}_+|}} + \left((\mathbf{x}-\hat{\mu}_-)^T \hat{\Sigma}_-^{-1} (\mathbf{x}-\hat{\mu}_-)\right) - \left((\mathbf{x}-\hat{\mu}_+)^T \hat{\Sigma}_+^{-1} (\mathbf{x}-\hat{\mu}_+)\right)\right]$$

0, $p=0.5$ $\qquad$ $\hat{\Sigma}_- = \hat{\Sigma}_+$

$$= \frac{1}{2}\left[\left((x-\hat{\mu}_-)^T \hat{\Sigma}^{-1} (x-\hat{\mu}_-)\right) - \left((x-\hat{\mu}_+)^T \hat{\Sigma}^{-1} (x-\hat{\mu}_+)\right)\right]$$

$$= \frac{1}{2}\left[\left(\cancel{x^T \hat{\Sigma}^{-1} x} - 2x^T \Sigma^{-1} \mu_- + \hat{\mu}_- \Sigma^{-1} \mu_-\right) - \left(\cancel{x^T \hat{\Sigma}^{-1} x} - 2x^T \hat{\Sigma}^{-1} \hat{\mu}_+ + \hat{\mu}_+ \Sigma^{-1} \mu_+\right)\right]$$

$$\underbrace{(\hat{\mu}_+ - \hat{\mu}_-)^T \hat{\Sigma}^{-1}}_{w^T} x \underbrace{+ \frac{1}{2}\hat{\mu}_-^T \hat{\Sigma}^{-1} \hat{\mu}_- + \frac{1}{2}\hat{\mu}_+^T \Sigma^{-1} \mu_+}_{w_0}$$

Fisher's linear discriminant analysis LDA (c=2)

- Suppose we fix *p=.5*
- Further, assume covariances are equal: $\hat{\Sigma}_- = \hat{\Sigma}_+ = \hat{\Sigma}$
- Then the discriminant function

$$f(\mathbf{x}) = \log \frac{p}{1-p} + \frac{1}{2}\left[\log \frac{|\hat{\Sigma}_-|}{|\hat{\Sigma}_+|} + \Big((\mathbf{x}-\hat{\mu}_-)^T \hat{\Sigma}_-^{-1}(\mathbf{x}-\hat{\mu}_-)\Big) - \Big((\mathbf{x}-\hat{\mu}_+)^T \hat{\Sigma}_+^{-1}(\mathbf{x}-\hat{\mu}_+)\Big)\right]$$

simplifies: $f(\mathbf{x}) = \mathbf{x}^T \hat{\Sigma}^{-1}(\hat{\mu}_+ - \hat{\mu}_-) + \frac{1}{2}(\hat{\mu}_-^T \Sigma^{-1}\hat{\mu}_- - \hat{\mu}_+^T \Sigma^{-1}\hat{\mu}_+)$

- Under these assumptions, we predict

$$y = \text{sign}(f(\mathbf{x})) = \text{sign}(\mathbf{w}^T \mathbf{x} + w_0) \quad \mathbf{w} = \hat{\Sigma}^{-1}(\hat{\mu}_+ - \hat{\mu}_-)$$

$$w_0 = \frac{1}{2}(\hat{\mu}_-^T \Sigma^{-1}\hat{\mu}_- - \hat{\mu}_+^T \Sigma^{-1}\hat{\mu}_+)$$

- This linear classifier is called
  Fisher's linear discriminant analysis