Winter 2019

CSEN1022: Machine Learning

Probabilistic Classifiers

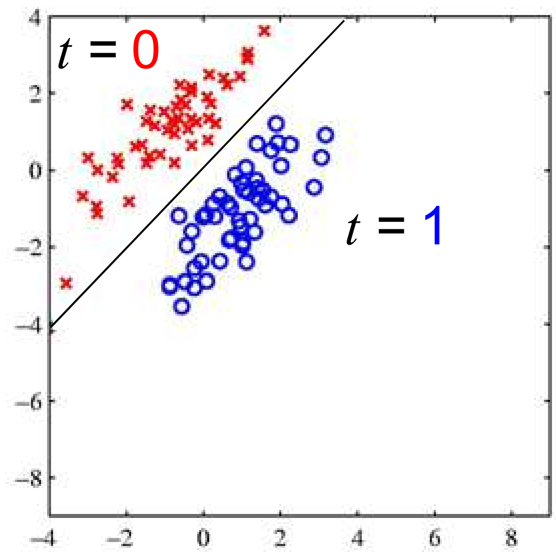
Seif Eldawlatly

Linear Classification

- Classification Problem

  Input vector: $\mathbf{x}$

  Target variable (vector): t



- Discriminant Functions
  - Least Squares
  - Fisher's Linear Discriminant
  - Perceptron

- Probabilistic Generative Models

- Probabilistic Discriminative Models

Linear Probabilistic Models for Classification

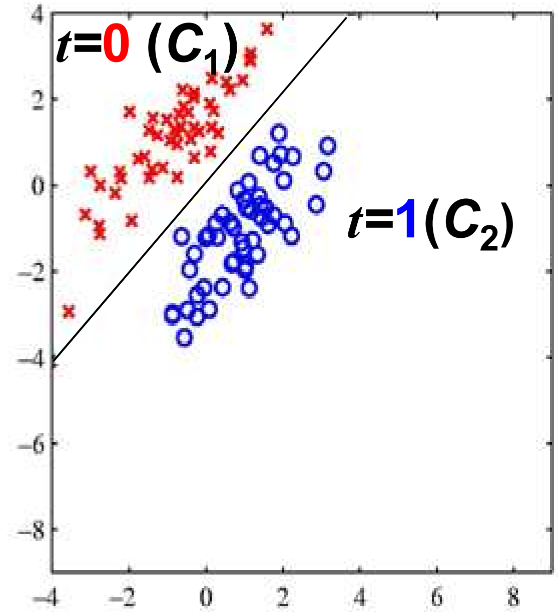
- Bayes' Theorem

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}$$

$p(C_k)$: Prior probability

$p(\mathbf{x}|C_k)$: Likelihood

$p(C_k|\mathbf{x})$: Posterior probability (Knowing $\mathbf{x}$, what is the probability of C_k?)



Linear Probabilistic Models for Classification

- Example

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)\,p(C_k)}{p(\mathbf{x})}$$

$\mathbf{x}$

C_k

	Non-smoker	Smoker	Total
Not cancer	40	3	43
Cancer	7	10	17
Total	47	13	60

- If a patient is a smoker, would he have cancer?

p(Smoker|Cancer)=10/17, p(Cancer)=17/60, p(Smoker)=13/60

p(Cancer|Smoker)=(10/17).(17/60)/(13/60)=10/13

max C = Cancer

p(Not cancer|Smoker)=3/13

Linear Probabilistic Models for Classification

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}$$

- Inference Stage: Find $p(C_k|\mathbf{x})$
- Decision Stage: $k^* = \arg\max_k p(C_k|\mathbf{x})$

- Probabilistic Generative Model: Learns $p(\mathbf{x}|C_k)$ and $p(C_k)$
- Probabilistic Discriminative Model: Learns $p(C_k|\mathbf{x})$ directly

Naïve Bayes Classifier

- Sometimes it is difficult to estimate $p(\mathbf{x}|C_k)$ for high dimensional data
- A simple solution is to assume independence between the features
- Naïve Bayes approximation

$$p(\mathbf{x}|C_k) \approx \prod_{j=1}^{D} p(x_j|C_k)$$

where D is the dimensionality of the input data

Naïve Bayes Classifier

- Example:

  Consider the data about car theft given in the table below

Example No.	Color	Type	Origin	**Stolen?**
1	Red	Sports	Domestic	**Yes**
2	Red	Sports	Domestic	**No**
3	Red	Sports	Domestic	**Yes**
4	Black	Sports	Domestic	**No**
5	Black	Sports	Imported	**Yes**
6	Black	SUV	Imported	**No**
7	Black	SUV	Imported	**Yes**
8	Black	SUV	Domestic	**No**
9	Red	SUV	Imported	**No**
10	Red	Sports	Imported	**Yes**

Using Naïve Bayes classifier, predict whether a Red Domestic SUV is stolen or not. Note that since the data is discrete, you can use the frequentist statistics to compute the needed probabilities.

Naïve Bayes Classifier

- Solution:

  Since the goal is to classify a Red Domestic SUV as stolen or not, we first define two classes C_1 and C_2, corresponding to Stolen = Yes and Stolen = No, respectively.

  To classify the given car with attributes $\mathbf{x}$, we need to compute $p(C_1|\mathbf{x})$:

  p(Stolen = Yes | Color = Red, Type = SUV, Origin = Domestic)

  and $p(C_2|\mathbf{x})$:

  p(Stolen = No | Color = Red, Type = SUV, Origin = Domestic)

  and find which conditional probability is larger. If the first one is larger, then our prediction is Stolen = Yes. If the second one is larger, then our prediction is Stolen = No. Note that $\mathbf{x}$ here is 3 dimensional corresponding to Color, Type and Origin.

Naïve Bayes Classifier

Since $$p(C_1|\mathbf{x}) = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x})}$$

We need to compute $p(\mathbf{x}|C_1)$ = *p*(Color = Red, Type = SUV, Origin = Domestic | Stolen = Yes).

Using the Naïve Bayes assumption which assumes that the dimensions of the input data (the attributes of the car) are independent, we can re-write $p(\mathbf{x}|C_1)$ as

$$p(\mathbf{x}|C_1) = \prod_{i=1}^{D} p(x_i|C_1)$$

= *p*(Color = Red | Stolen = Yes) *p*(Type = SUV | Stolen = Yes) *p*(Origin = Domestic | Stolen = Yes)

Similarly, $p(\mathbf{x}|C_2)$ can be re-written as $p(\mathbf{x}|C_2) = \prod_{i=1}^{D} p(x_i|C_2)$

= *p*(Color = Red | Stolen = No) *p*(Type = SUV | Stolen = No) *p*(Origin = Domestic | Stolen = No)

Naïve Bayes Classifier

- From the available data in the table and using frequentist statistics:

p(Color = Red | Stolen = Yes) = 3/5 (out of the 5 stolen cars, 3 were red)
p(Color = Red | Stolen = No) = 2/5 (out of the 5 non-stolen cars, 2 were red)
p(Type = SUV | Stolen = Yes) = 1/5 (out of the 5 stolen cars, 1 was SUV)
p(Type = SUV | Stolen = No) = 3/5
p(Origin = Domestic | Stolen = Yes) = 2/5
p(Origin = Domestic | Stolen = No) = 3/5

Therefore,
p(Color = Red | Stolen = Yes) p(Type = SUV | Stolen = Yes) p(Origin = Domestic | Stolen = Yes) = (3/5) x (1/5) x (2/5) = 0.048

And
p(Color = Red | Stolen = No) p(Type = SUV | Stolen = No) p(Origin = Domestic | Stolen = No) = (2/5) x (3/5) x (3/5) = 0.144

Also p(Stolen = Yes) = 5/10 and p(Stolen = No) = 5/10

Naïve Bayes Classifier

To classify the given car, we need to compare $p(C_1|\mathbf{x})$ to $p(C_2|\mathbf{x})$ such that

If $\frac{p(C_1|\mathbf{x})}{p(C_2|\mathbf{x})} > 1$ then $\mathbf{x} \in C_1$, otherwise $\mathbf{x} \in C_2$

$$\because \frac{p(C_1|\mathbf{x})}{p(C_2|\mathbf{x})} = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_2)p(C_2)}$$

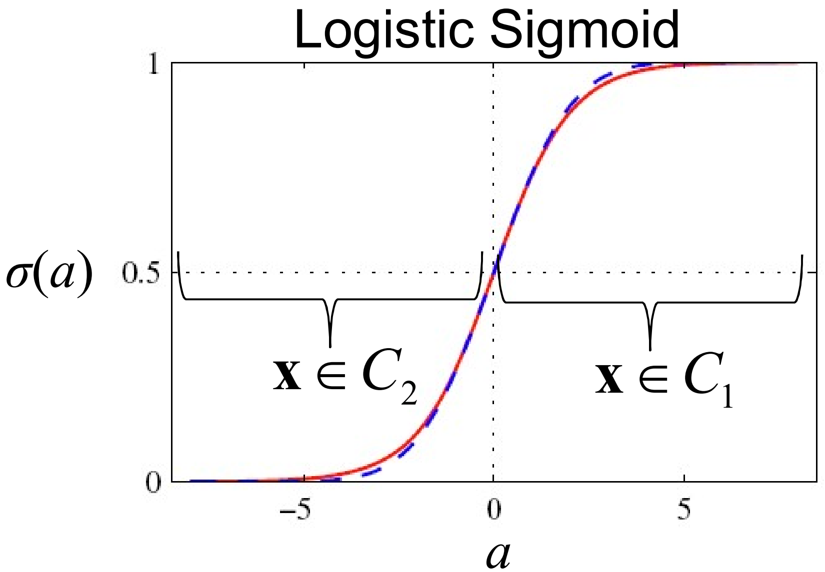
Therefore, for the this problem

$$\frac{p(C_1|\mathbf{x})}{p(C_2|\mathbf{x})} = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_2)p(C_2)} = \frac{0.048 \times 0.5}{0.144 \times 0.5} = 0.333$$

Therefore, our prediction is C_2 which is that the car is not stolen.

Probabilistic Generative Models

- We first represent the posterior probability as



$$
\begin{aligned}
p(\mathcal{C}_1|\mathrm{x}) &= \frac{p(\mathrm{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathrm{x}|\mathcal{C}_1)p(\mathcal{C}_1) + p(\mathrm{x}|\mathcal{C}_2)p(\mathcal{C}_2)} \\
&= \frac{1}{1+\exp(-a)} = \sigma(a)
\end{aligned}
$$

where

$$a = \ln \frac{p(\mathrm{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathrm{x}|\mathcal{C}_2)p(\mathcal{C}_2)}$$

If a > 0 → $\frac{p(\mathrm{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathrm{x}|\mathcal{C}_2)p(\mathcal{C}_2)}$ >1 → $p(C_1|\mathbf{x}) > p(C_2|\mathbf{x})$ → $\mathbf{x} \in C_1$

If a < 0 → $\frac{p(\mathrm{x}|\mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathrm{x}|\mathcal{C}_2)p(\mathcal{C}_2)}$ <1 → $p(C_1|\mathbf{x}) < p(C_2|\mathbf{x})$ → $\mathbf{x} \in C_2$

Probabilistic Generative Models

- Assume Gaussian distribution for class-conditional densities $p(\mathbf{x}|\mathcal{C}_k)$

$$p(\mathbf{x}|\mathcal{C}_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\mathbf{\Sigma}|^{1/2}} \exp\left\{ -\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^{\mathrm{T}} \mathbf{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu}_k) \right\}$$

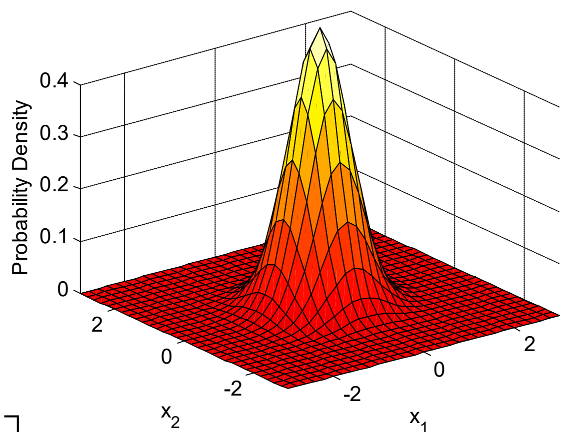
$\boldsymbol{\mu}_k$: Mean

$\mathbf{\Sigma}$: Covariance Matrix

(Common for both classes)

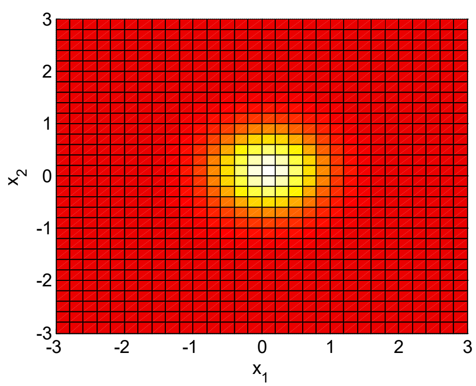
- Example

$p(\mathbf{x}|C_k)$

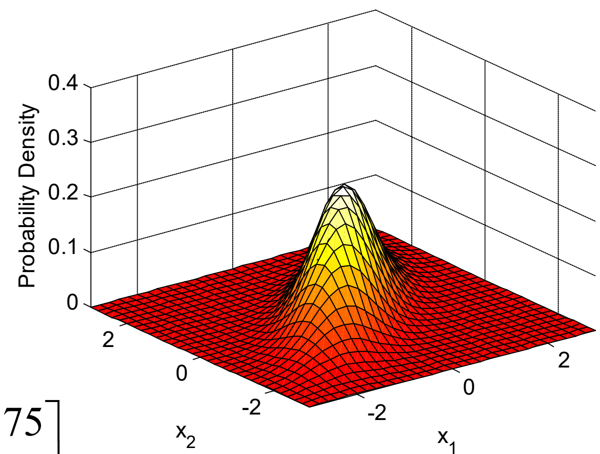


$$\Sigma = \begin{bmatrix} 0.3 & 0 \\ 0 & 0.3 \end{bmatrix}$$

Diagonal Covariance

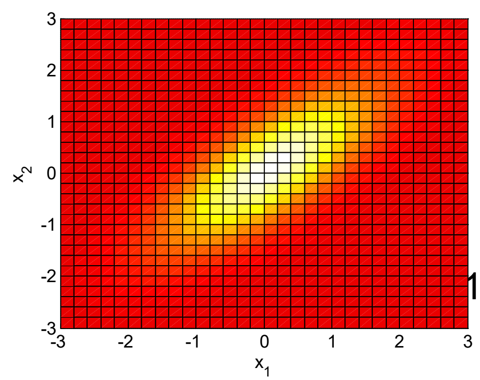


$p(\mathbf{x}|C_k)$



$$\Sigma = \begin{bmatrix} 1 & 0.75 \\ 0.75 & 1 \end{bmatrix}$$

Non-diagonal Covariance



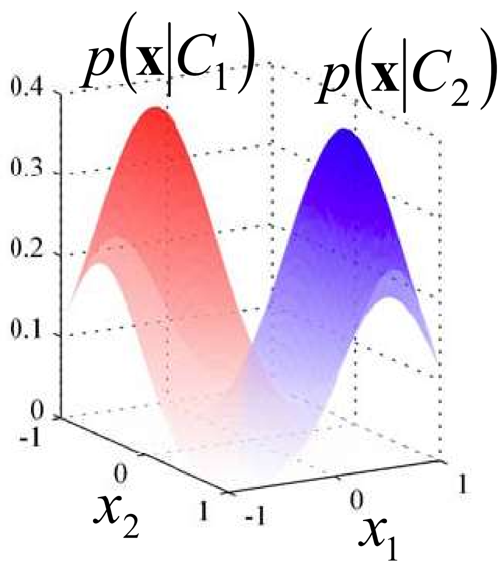
Probabilistic Generative Models

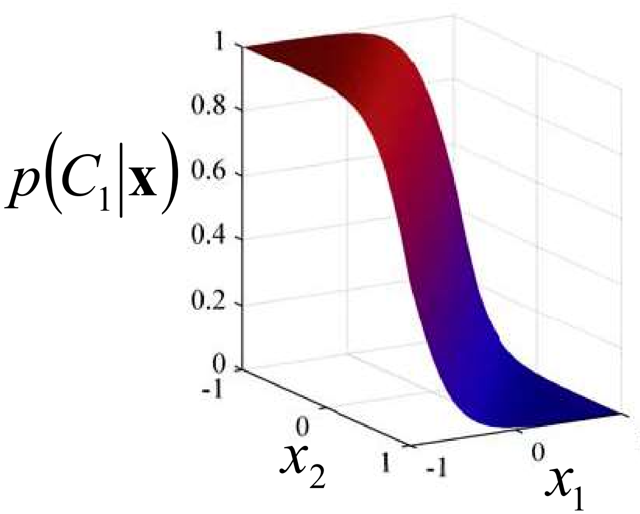
- Using Gaussian assumption and logistic sigmoid representation, we can show that

$$p(\mathcal{C}_1|\mathbf{x}) = \sigma(\mathbf{w}^{\mathrm{T}}\mathbf{x} + w_0)$$

where

$$\begin{aligned} \mathbf{w} &= \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \\ w_0 &= -\frac{1}{2}\boldsymbol{\mu}_1^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_2^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_2 + \ln\frac{p(\mathcal{C}_1)}{p(\mathcal{C}_2)} \end{aligned}$$





Probabilistic Generative Models

- Using the previous assumptions and a training dataset $\{\mathbf{x}_n, t_n\}$, we can estimate the values of the parameters

$$\boldsymbol{\mu}_k \quad \boldsymbol{\Sigma} \quad p(\mathcal{C}_k)$$

- Maximum Likelihood Estimation (MLE)
  - Let t_n = 1 denote class C_1 and t_n = 0 denote class C_2
  - Let $p(\mathcal{C}_1) = \pi$ → $p(\mathcal{C}_2) = 1 - \pi$
  - For $\mathbf{x}_n \in C_1$

$$p(\mathbf{x}_n, \mathcal{C}_1) = p(\mathcal{C}_1)p(\mathbf{x}_n|\mathcal{C}_1) = \pi\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$$

  - For $\mathbf{x}_n \in C_2$

$$p(\mathbf{x}_n, \mathcal{C}_2) = p(\mathcal{C}_2)p(\mathbf{x}_n|\mathcal{C}_2) = (1-\pi)\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \boldsymbol{\Sigma})$$

Probabilistic Generative Models

- Maximum Likelihood Estimation (MLE)
  - We define the likelihood of the data as the probability of observing the available data
  - Let $\mathbf{D}$ denote the available data, where $\mathbf{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$
  - The probability of observing the data $p(\mathbf{D})$ (Given the parameters)

$$p(\mathbf{D}) = p(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N)$$

  - Assuming independence between the input vectors

$$p(\mathbf{D}) = p(\mathbf{x}_1)p(\mathbf{x}_2) \ldots p(\mathbf{x}_N) = \prod_{n=1}^{N} p(\mathbf{x}_n) = \prod_{i=1}^{N_1} p(\mathbf{x}_i, C_1) \prod_{j=1}^{N_2} p(\mathbf{x}_j, C_2)$$

  - Given the assumption of the previous slide, the likelihood can be modeled as

$$p(\mathbf{t}|\pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \prod_{n=1}^{N} \left[\pi \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \boldsymbol{\Sigma})\right]^{t_n} \left[(1-\pi)\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \boldsymbol{\Sigma})\right]^{1-t_n}$$

Maximum Likelihood Estimation

- MLE Solution:
  - Find the parameters that maximize the likelihood of the data
  - Given that the maximum of this function is achieved at the same value that maximizes the log of the function, we use the Log-likelihood of the function for convenience
  - Remember that $\ln(AB) = \ln(A) + \ln(B)$, $\ln(A^m) = m \ln(A)$

$$\ln p(\mathbf{t}|\pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma) = \ln \prod_{n=1}^{N} \left[\pi \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \Sigma)\right]^{t_n} \left[(1-\pi)\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \Sigma)\right]^{1-t_n}$$

$$= \sum_{n=1}^{N} \ln\left(\left[\pi \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \Sigma)\right]^{t_n} \left[(1-\pi)\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \Sigma)\right]^{1-t_n}\right)$$

$$= \sum_{n=1}^{N} t_n \ln\left(\pi \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \Sigma)\right) + (1-t_n)\ln\left((1-\pi)\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \Sigma)\right)$$

$$= \sum_{n=1}^{N} t_n \ln(\pi) + t_n \ln\left(\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1, \Sigma)\right) + (1-t_n)\ln\left((1-\pi)\right) + (1-t_n)\ln\left(\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2, \Sigma)\right)$$

Maximum Likelihood Estimation

$$\ln p(\mathbf{t}|\pi,\boldsymbol{\mu}_1,\boldsymbol{\mu}_2,\Sigma)$$

$$=\sum_{n=1}^{N} t_n \ln(\pi) + t_n \ln\left(\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_1,\Sigma)\right) + (1-t_n)\ln((1-\pi)) + (1-t_n)\ln\left(\mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_2,\Sigma)\right)$$

- Derivative w.r.t. parameters and equate with zero

π $$\pi = \frac{1}{N}\sum_{n=1}^{N} t_n = \frac{N_1}{N} = \frac{N_1}{N_1+N_2}$$

$\boldsymbol{\mu}_1$ $$\boldsymbol{\mu}_1 = \frac{1}{N_1}\sum_{n=1}^{N} t_n \mathbf{x}_n$$

$\boldsymbol{\mu}_2$ $$\boldsymbol{\mu}_2 = \frac{1}{N_2}\sum_{n=1}^{N} (1-t_n)\mathbf{x}_n$$

$\boldsymbol{\Sigma}$ $$\boldsymbol{\Sigma} = \frac{N_1}{N}\mathbf{S}_1 + \frac{N_2}{N}\mathbf{S}_2$$

$$\mathbf{S}_1 = \frac{1}{N_1}\sum_{n\in\mathcal{C}_1}(\mathbf{x}_n-\boldsymbol{\mu}_1)(\mathbf{x}_n-\boldsymbol{\mu}_1)^{\mathrm{T}}$$

$$\mathbf{S}_2 = \frac{1}{N_2}\sum_{n\in\mathcal{C}_2}(\mathbf{x}_n-\boldsymbol{\mu}_2)(\mathbf{x}_n-\boldsymbol{\mu}_2)^{\mathrm{T}}$$

- Using the expression in slide 8, $p(\mathcal{C}_1|\mathbf{x}) = \sigma(\mathbf{w}^{\mathrm{T}}\mathbf{x}+w_0)$ can be obtained

Linear Probabilistic Models for Classification

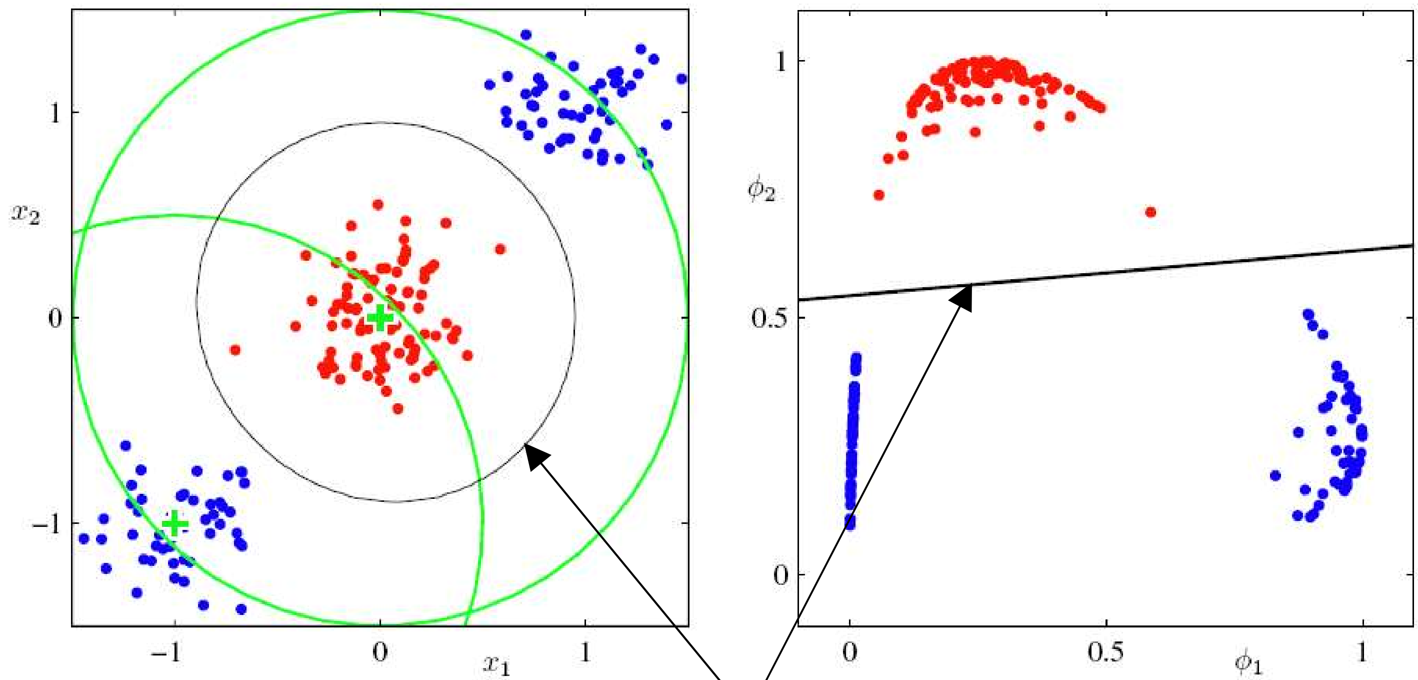
$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}$$

- Inference Stage: Find $p(C_k|\mathbf{x})$
- Decision Stage: $k^* = \max_k p(C_k|\mathbf{x})$

- Probabilistic Generative Model: Learns $p(\mathbf{x}|C_k)$ and $p(C_k)$
- Probabilistic Discriminative Model: Learns $p(C_k|\mathbf{x})$ directly

Probabilistic Discriminative Models

- Learn $p(C_k|\mathbf{x})$ directly
- Using a nonlinear transformation of the data $\phi(\mathbf{x})$ (basis function)

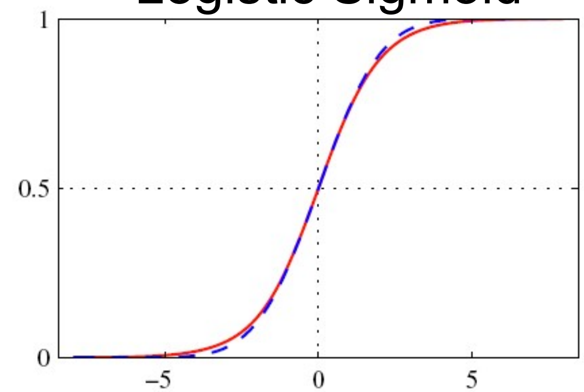


Decision Boundary

Logistic Regression

- Assumption

Logistic Sigmoid



$$p(\mathcal{C}_1|\phi) = y(\phi) = \sigma\left(\mathbf{w}^{\mathrm{T}}\phi\right)$$

where $$\sigma(a) = \frac{1}{1+\exp(-a)}$$

- Logistic Regression vs. MLE for Gaussian Generative Model (For 2 classes)
  - No. of Parameters for Logistic Regression: D parameters
  - No. of Parameters for MLE: $D(D+5)/2 + 1$ parameters

$$\mu_k \rightarrow 2D \qquad \Sigma \rightarrow D(D+1)/2 \qquad p(\mathcal{C}_k) \rightarrow 1$$

Logistic Regression

$$p(\mathcal{C}_1|\phi) = y(\phi) = \sigma\left(\mathbf{w}^{\mathrm{T}}\phi\right)$$

- Using maximum likelihood to estimate $\mathbf{w}$ from the training dataset $\{\boldsymbol{\varphi}_n, t_n\}$. Likelihood function

$$p(\mathbf{t}|\mathbf{w}) = \prod_{n=1}^{N} y_n^{t_n} \left\{1 - y_n\right\}^{1-t_n}$$

- Minimizing the negative of the log-likelihood is equivalent to maximizing the log-likelihood

$$E(\mathbf{w}) = -\ln p(\mathbf{t}|\mathbf{w}) = -\sum_{n=1}^{N} \left\{t_n \ln y_n + (1 - t_n)\ln(1 - y_n)\right\}$$

Logistic Regression

- We make use of the property

$$\frac{d\sigma}{da} = \sigma(1-\sigma)$$

- Take the derivative of $E(\mathbf{w})$ w.r.t. $\mathbf{w}$

$$\nabla E(\mathbf{w}) = -\sum_{n=1}^{N}\left\{t_n \frac{1}{\sigma_n}\sigma_n(1-\sigma_n)\phi_n + (1-t_n)\frac{1}{1-\sigma_n}(-\sigma_n(1-\sigma_n))\phi_n\right\}$$

$$= -\sum_{n=1}^{N}\{t_n(1-\sigma_n)\phi_n - (1-t_n)\sigma_n\phi_n\}$$

$$= \sum_{n=1}^{N}(y_n - t_n)\phi_n$$

- Optimal $\mathbf{w}$ can be found using gradient descent

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta\frac{\partial E}{\partial \mathbf{w}^{(\tau)}}$$