

CS 5806 Machine Learning II

Lecture 10 - Classification with Mixture of Gaussians
September 27th, 2023
Hoda Eldardiry

Recommended reading: [6] Sec 4.2, [8] Sec 4.2

Classification using Mixtures of Gaussians

- A **statistical** classification model
- Based on **Mixtures of Gaussians**
- **Generative**
- **Probabilistic**
- **Linear**

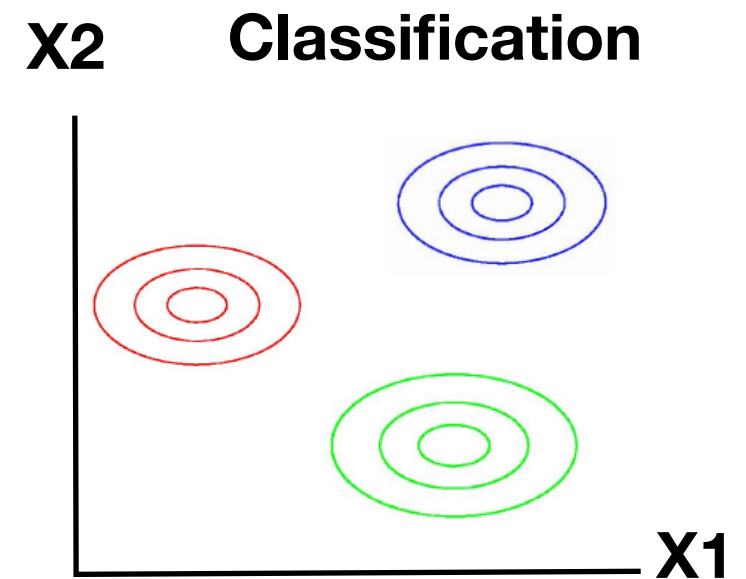
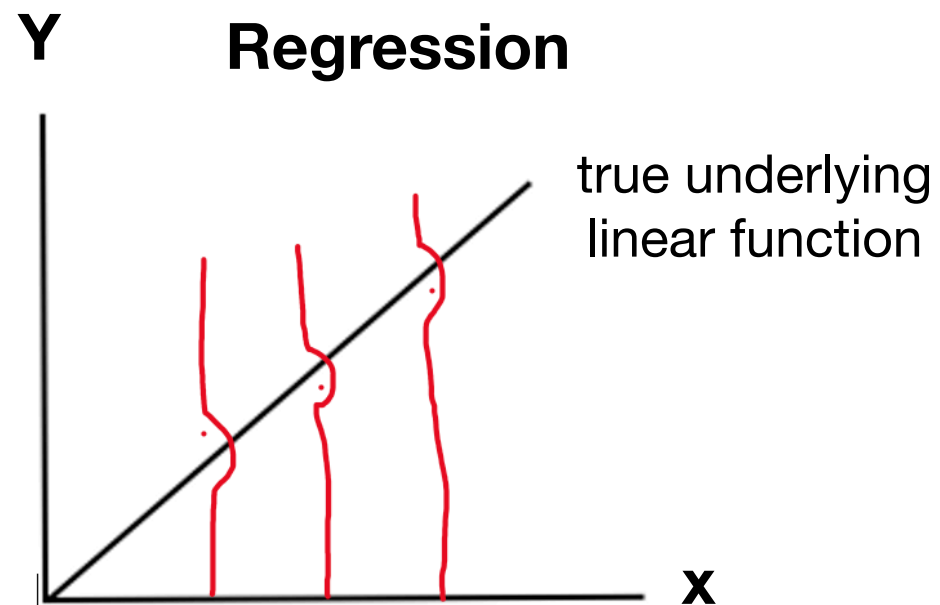
Generative Models

- **Probabilistic Generative Models**
 - Can be used for regression & classification
 - Can be used to simulate the creation of a dataset
 - Can generate data similar to training data
- Why do we need to generate data?

Data Generation Applications

- **Text generation**
 - Conversation agent
 - Goal: generate a response similar to responses we might find in a dataset of (message, response) pairs
- **Image generation**
 - Diagnosis from medical images
 - Goal: generate synthetic images to
 - (1) preserve patient privacy
 - (2) learn models to understand cases not in training data

Probabilistic Generative Models



- Generative models can be used for regression & classification

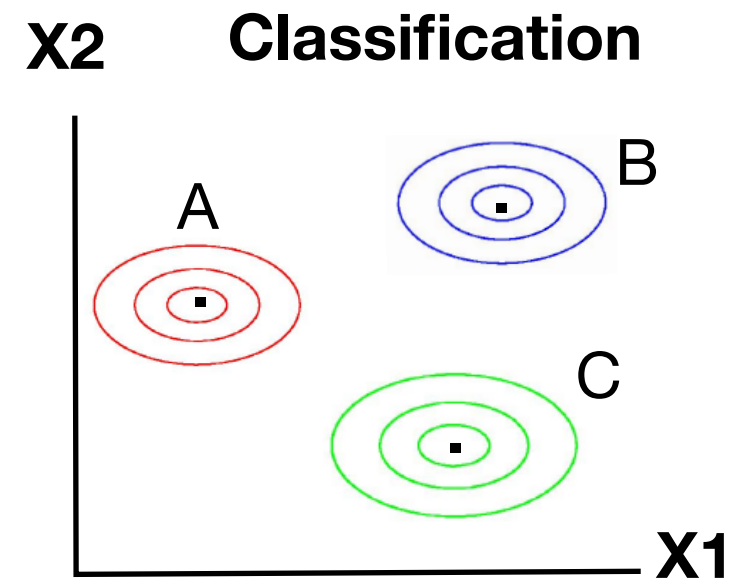
Generative Model for Regression



- Goal: fit a line/curve
Optimization perspective: minimize distance
Statistical perspective: perfect data lie on line
Imperfect data: assume noise has gaussian distribution
For each point x , assume output sampled from a Gaussian distn.
- **Data generation:**
For each given x , sample Y from a Gaussian distribution

Generative Model for Classification

- Assume in some domain:
 - 3 possible classes A, B, C
 - 2 input components x_1 , x_2
- Classification assumption:
 - Every data point belongs to 1 class
 - Every point in each class should be centered at class region
 - Input may not be accurate
 - Gaussian distribution capture noise in input measurements
 - Data follow a Gaussian distribution centered at each class
 - Circles: contour lines of 3D surfaces of Gaussian distributions
 - Distributions for where data in each class is located



Construct a Probabilistic Model

- For each class there's a distribution
Data can be in different regions
==> Construct a probabilistic model
- Define **$P(C)$** : **prior distribution of class C**
- $P(C)$ captures region including data points in C
- We call it a **prior** probability distribution
because we define it (or make an assumption about it)
without **prior knowledge** (a-priori)
before observing a single data point

Prior & Likelihood

- $P(C)$: prior distribution of class C
- Once we observe 1 data point
- We ask:
- What's the probability that if we pick a point from class C , that it would be at a certain location (x -value)?
- **$P(x|C)$:**
 - Class Conditional Distribution of x
 - Can be used to generate points from class C

Posterior Distribution for Classification

- Posterior $P(C|x)$ estimates probability of class C for a given data point x
 - Given prior $P(C)$ & likelihood $P(x|C)$
 - Compute posterior $P(C|x)$ using Bayes Theorem
 - Posterior = Prior * Likelihood
 - $P(C|x) = P(C) * P(x|C) / \sum_c P(C) * P(x|C)$
 - Denominator is a normalizing constant
 - Class Prediction of x : $P(C|x) = k * P(C) * P(x|C)$

Bayesian Inference vs. Bayesian Learning

- This is not Bayesian Learning
- Bayesian Learning: use Bayes Theorem for learning
 - learn/estimate model parameters
- Bayes Inference: use Bayes Theorem for inference
 - infer class: estimate class probability for a given data point
 - compute posterior probability of a class
- We have not discussed model parameters yet

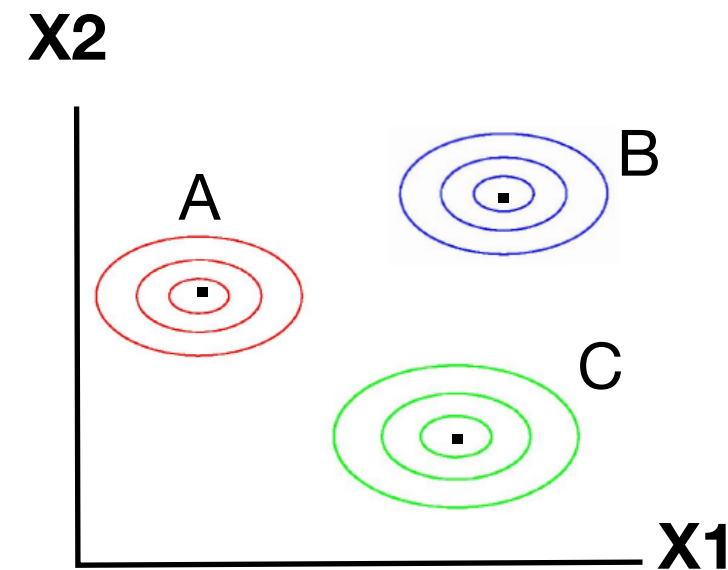
Model Assumptions

- Finite number of classes
- Classes are categorical
- A natural prior distribution in this case: Multinomial
- Multinomial distribution:
 - Distribution over finite number of categorical outcomes
- Coin: 2 categorical outcomes:
Bernoulli distribution (probability of head, tail)
- Dice: multiple categorical outcomes:
Multinomial distribution (probability of each of 6 faces)

Model Assumptions

- π_k : probability of class k
- π_k : a number between 0 & 1
- $x \in \mathbb{R}^d$
- Data defined by space of Real values & has d dimensions
- $P(x|C)$: class conditional distribution:
for a given class, where the data is likely to be in that space
- Assume $P(x|C)$ is a Gaussian distribution

Class Conditional Distribution: Gaussian



- Gaussian distributions are examples of class conditional distributions modeled by Gaussians
- Simplifying assumption:
Each Gaussian distribution has the same covariance matrix Σ
Same covariance matrix Σ for each class
- Class conditional distribution can be written as proportional to an exponential of the form:

$$\Pr(\mathbf{x}|c_k) \propto e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}$$

- More generally, we can consider different covariant matrices for different classes but we will stick to this for now

Posterior Distribution

- We defined the prior as multinomial π_k
- We defined the likelihood as a gaussian $e^{-\frac{1}{2}(x-\mu_k)^T \Sigma^{-1}(x-\mu_k)}$
- We can compute the posterior to do inference to estimate probability of each class given a data point
- Posterior = (normalizing constant) * prior * likelihood

$$\Pr(c_k | \mathbf{x}) = \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x}-\mu_k)^T \Sigma^{-1}(\mathbf{x}-\mu_k)}}{\sum_k \pi_k e^{-\frac{1}{2}(\mathbf{x}-\mu_k)^T \Sigma^{-1}(\mathbf{x}-\mu_k)}}$$

Posterior Distribution

- Rewrite exponent
- π_k * exponent with 3 terms
- Pull out 1st term (doesn't depend on class)
- Simplify other 2 terms in exponent
- Restrict to 2 classes
- Divide numerator & denominator by numerator

$$\begin{aligned}\Pr(c_k | \mathbf{x}) &= \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x} - \mu_k)^T \Sigma^{-1} (\mathbf{x} - \mu_k)}}{\sum_k \pi_k e^{-\frac{1}{2}(\mathbf{x} - \mu_k)^T \Sigma^{-1} (\mathbf{x} - \mu_k)}} \\ &= \frac{\pi_k e^{-\frac{1}{2}(\cancel{\mathbf{x}^T \Sigma^{-1} \mathbf{x}} - 2\mu_k^T \Sigma^{-1} \mathbf{x} + \mu_k^T \Sigma^{-1} \mu_k)}}{\sum_k \pi_k e^{-\frac{1}{2}(\cancel{\mathbf{x}^T \Sigma^{-1} \mathbf{x}} - 2\mu_k^T \Sigma^{-1} \mathbf{x} + \mu_k^T \Sigma^{-1} \mu_k)}}\end{aligned}$$

Consider two classes c_k and c_j

$$= \frac{1}{1 + \frac{\pi_j e^{\mu_j^T \Sigma^{-1} \mathbf{x} - \frac{1}{2} \mu_j^T \Sigma^{-1} \mu_j}}{\pi_k e^{\mu_k^T \Sigma^{-1} \mathbf{x} - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k}}}$$

Posterior Distribution

- Reorganize
- Exponent is linear in x , re-write as $(\mathbf{w}^T \mathbf{x} + w_0)$
- \mathbf{W} : coefficient of x

- w_0 : constant
doesn't depend on x

- Posterior function?

$$= \frac{1}{1 + e^{-\left(\mu_k^T - \mu_j^T\right) \Sigma^{-1} x + \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_j^T \Sigma^{-1} \mu_j - \ln \frac{\pi_k}{\pi_j}}}$$

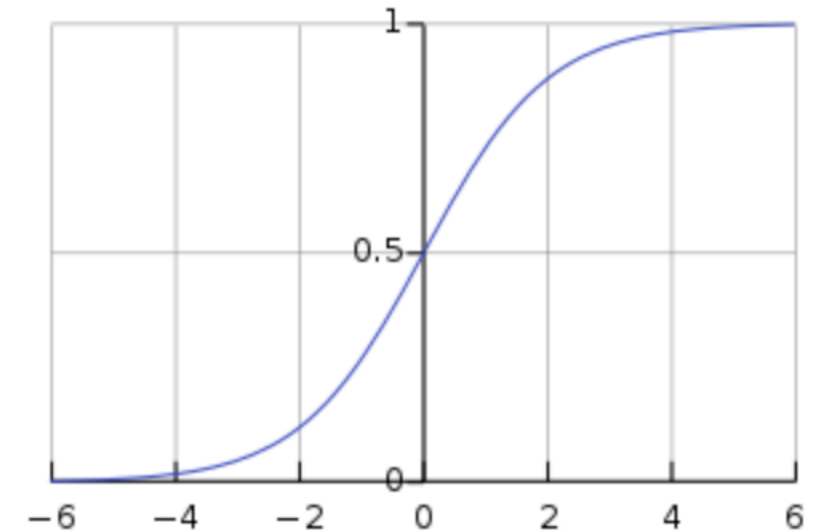
$$= \frac{1}{1 + e^{-(\mathbf{w}^T \mathbf{x} + w_0)}}$$

where $\mathbf{w} = \Sigma^{-1}(\mu_k - \mu_j)$

and $w_0 = -\frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \frac{1}{2} \mu_j^T \Sigma^{-1} \mu_j + \ln \frac{\pi_k}{\pi_j}$

Logistic Sigmoid

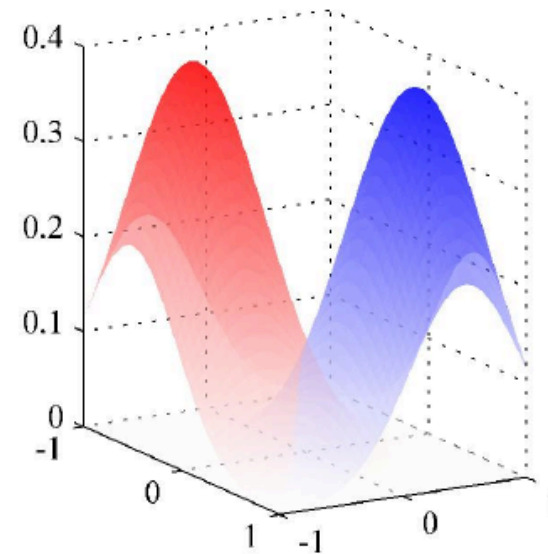
- Let $\sigma(a) = \frac{1}{1+e^{-a}}$
└──────────┘ Logistic sigmoid
- Then $\Pr(c_k|\mathbf{x}) = \sigma(\mathbf{w}^T \mathbf{x} + w_0)$
- Derived from computing posterior in context of mixtures of Gaussians
- Starts at 0, asymptotically reaches 1
- Input a: any real number, Output: between 0 & 1
- Popular in neural networks. Used as last layer to compute a probability output
- Coefficients of x are treated as parameters W, the constant part is denoted as w_0
- In the context of Mixture of Gaussians: mean, covariance matrix & class probabilities can be combined to give parameters W and w_0



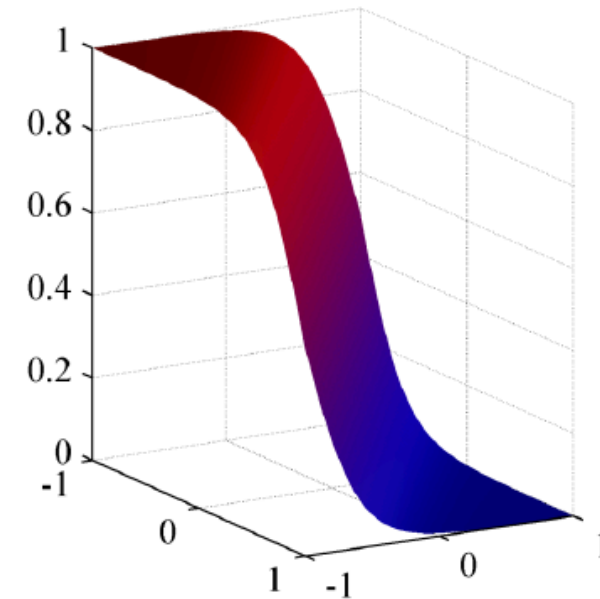
Logistic Sigmoid

- Classification:
find class of each data point
- Class conditionals (3D):
red & blue bell shape Gaussians
same as the contours 2D figure
- Compute posterior for a given point:
point on left: higher probability for red
point on right: higher probability for blue
- Posterior for red class: from 1 to 0 (from left to right)
- Posterior 3D surface: same as 2D curve
- 3D: generalization in higher dimension
- Both figures (conditionals & posteriors):
 - same x-axis: input data points
 - different y-axes
- Posterior curve y-axis: probability of red class (from 0 to 1)
- Conditional curve y-axis: probability density for each class (doesn't have to be from 0 to 1)
- Assuming Gaussian variances are the same (same covariance matrix) (they don't have to be), but the means are different
- Even if class conditional is not at the peak of a class, the class posterior can still be much higher than the other class (even close to 1)

class conditionals



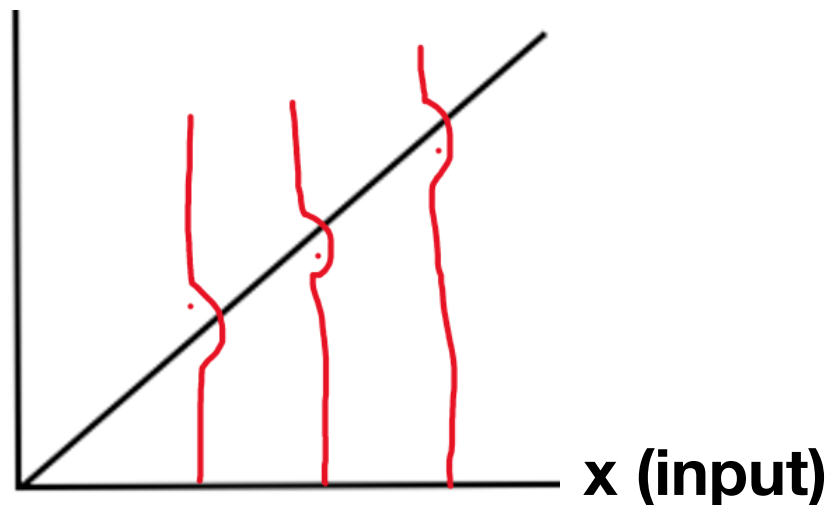
posterior



Mixture of Gaussians

Regression

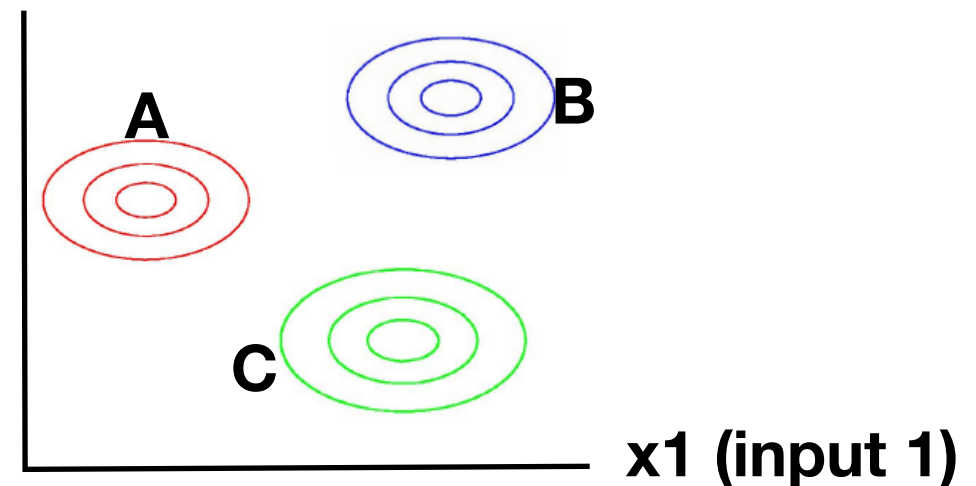
y (output)



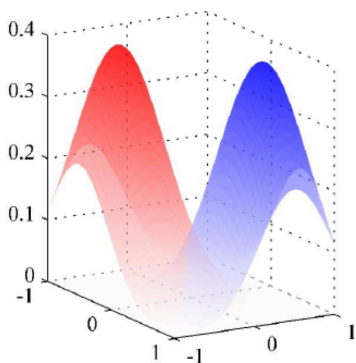
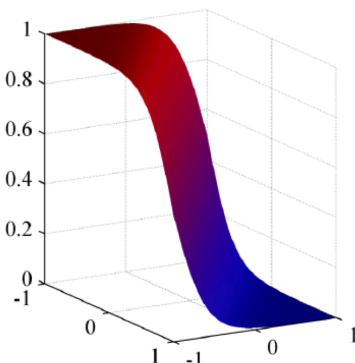
- **Regression:** fit curve/line to given points
 - Optimization: minimize distance of line to each point
 - Statistical assumptions:
 - true function is linear
 - noise in data follows a Gaussian distribution
 - i.e., for a given point,
 - output is sampled from a Gaussian distribution describe by the red curves
- **Data generation:** given input x,
 - Sample possible y from Gaussian distribution

Classification

x2 (input 2)



- **Classification:** input(x1, x2) \rightarrow output class (A, B or C)
 - Statistical assumptions:
 - noise in data follows a Gaussian distribution
 - i.e., for a given point,
 - output is sampled from a Gaussian distribution described by the 3D contours
 - **Data generation:** given input x,
 - Sample possible y from Gaussian distribution

Model Components	1) Prior P(Ci) “Learning”	2) Likelihood P(x C) “Learning”	3) Posterior P(C x) “Inference”
Definition	Prior distribution assumed; without prior knowledge of data	Class conditional distribution of x given class, likelihood to observe x?	Posterior probability of class C given an input x
Computation			Bayes: posterior=prior * likelihood $\Pr(C \mathbf{x}) = \frac{\Pr(\mathbf{x} C) \Pr(C)}{\sum_C \Pr(\mathbf{x} C) \Pr(C)}$
Assumptions & Derivation	prior: multinomial distribution $\Pr(C = c_k) = \pi_k$ classes: finite, categorical	likelihood: Gaussian distribution $\Pr(\mathbf{x} c_k) \propto e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}$ data: $\mathbf{x} \in \mathbb{R}^d$ covariance matrix $\boldsymbol{\Sigma}$ same for c1, c2	$\Pr(c_k \mathbf{x}) = \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}}{\sum_k \pi_k e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}}$ $= \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x}^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - 2 \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k)}}{\sum_k \pi_k e^{-\frac{1}{2}(\mathbf{x}^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - 2 \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k)}}$ Consider two classes c_k and c_j $= \frac{1}{1 + \frac{\pi_j e^{\boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j}}{\pi_k e^{\boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k}}}}$ $= \frac{1}{1 + e^{-\left(\boldsymbol{\mu}_k^T - \boldsymbol{\mu}_j^T\right) \boldsymbol{\Sigma}^{-1} \mathbf{x} + \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j - \ln \frac{\pi_k}{\pi_j}}}}$
Inference: Binary Classification			Posterior: Logistic Sigmoid $\Pr(c_k \mathbf{x}) = \frac{1}{1 + e^{-(\mathbf{w}^T \mathbf{x} + w_0)}}$ $\mathbf{w} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_j)$ $w_0 = -\frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \frac{1}{2} \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j + \ln \frac{\pi_k}{\pi_j}$
Plot			

Binary Prediction

- **Posterior (logistic sigmoid) function**
=> compute probability distribution over classes
=> make prediction
- For some applications
 - we don't want class probability
 - we want the predicted class
- Given class probabilities
=> choose class of highest probability
- Binary classification
=> choose class of probability ≥ 0.5

$$\begin{aligned} \text{best class} &= \operatorname{argmax}_k \Pr(c_k | \mathbf{x}) \\ &= \begin{cases} c_1 & \sigma(\mathbf{w}^T \mathbf{x} + w_0) \geq 0.5 \\ c_2 & \text{otherwise} \end{cases} \end{aligned}$$

Red

Blue

$$\text{Class boundary: } \sigma(\mathbf{w}_k^T \bar{\mathbf{x}}) = 0.5$$

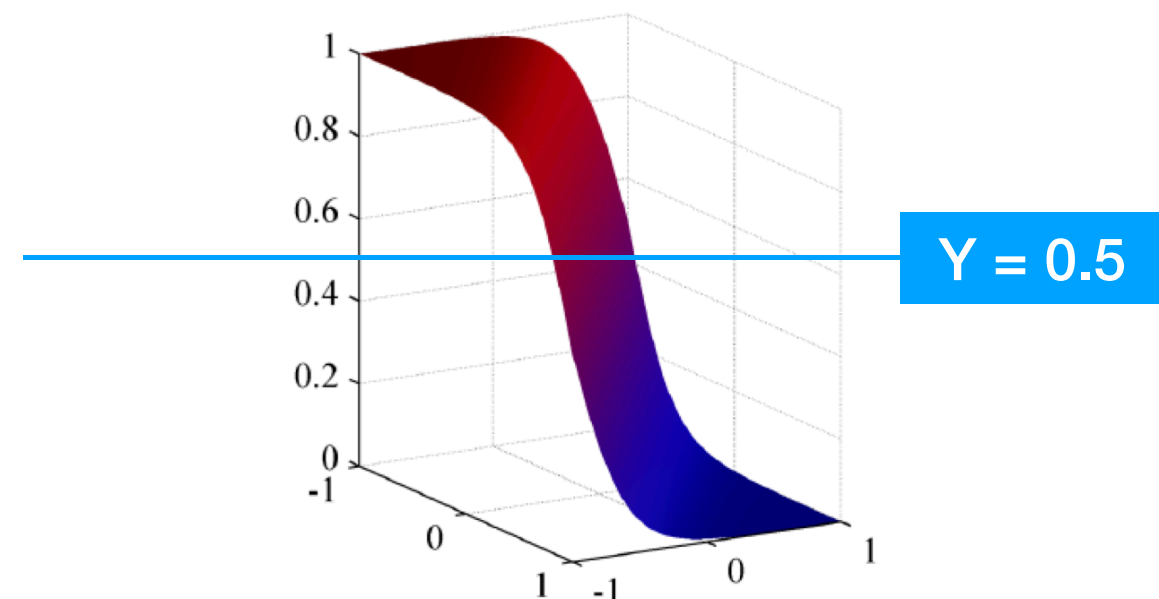
$$\Rightarrow \frac{1}{1 + e^{-(\mathbf{w}_k^T \bar{\mathbf{x}})}} = 0.5$$

$$\Rightarrow \mathbf{w}_k^T \bar{\mathbf{x}} = 0$$

Sigmoid input = 0

\therefore linear separator

Simple model



Multi-class Problems

- Consider Gaussian conditional distributions with identical Σ

$$\Pr(c_k | \mathbf{x}) = \frac{\Pr(c_k) \Pr(\mathbf{x} | c_k)}{\sum_j \Pr(c_j) \Pr(\mathbf{x} | c_j)}$$

Same expression as before

Covariance matrix

$$= \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x} - \mu_k)^T \Sigma^{-1} (\mathbf{x} - \mu_k)}}{\sum_j \pi_j e^{-\frac{1}{2}(\mathbf{x} - \mu_j)^T \Sigma^{-1} (\mathbf{x} - \mu_j)}}$$

Not dividing D & N by N (to derive sigmoid)
Just simplify D&N

N: linear expression in x

D: linear expression in x & summation over all classes j

We get the softmax function (popular in NNs)

$$= \frac{\pi_k e^{-\frac{1}{2}(-2\mu_k^T \Sigma^{-1} \mathbf{x} + \mu_k^T \Sigma^{-1} \mu_k)}}{\sum_j \pi_j e^{-\frac{1}{2}(-2\mu_j^T \Sigma^{-1} \mathbf{x} + \mu_j^T \Sigma^{-1} \mu_j)}}$$

$$= \frac{e^{\mu_k^T \Sigma^{-1} \mathbf{x} - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \ln \pi_k}}{\sum_j e^{\mu_j^T \Sigma^{-1} \mathbf{x} - \frac{1}{2} \mu_j^T \Sigma^{-1} \mu_j + \ln \pi_j}}$$

$$= \frac{e^{w_k^T \bar{\mathbf{x}}}}{\sum_j e^{w_j^T \bar{\mathbf{x}}}}$$

\Rightarrow softmax

Softmax: derived from
computing posterior distribution
for mixtures of Gaussians

$$\text{where } \mathbf{w}_k^T = \left(-\frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \ln \pi_k, \mu_k^T \Sigma^{-1} \right)$$

\mathbf{w}_0

Other entries in vector w:
Coefficients of x

Multi-class Prediction

- When there are several classes, the posterior is a **softmax** (generalization of the sigmoid)

- Softmax distribution: $\Pr(c_k | \mathbf{x}) = \frac{e^{f_k(\mathbf{x})}}{\sum_j e^{f_j(\mathbf{x})}}$

- Argmax distribution: Hence the name softMAX

hard/degenerate distribution:
1 class prob 1 & all others probability 0

SOFTmax softens argmax
(nonzero probability for all classes)
approximates argmax by rewriting it (take limit and push to infinity)

Then raise base to exponent of function of each class

$$\Pr(c_k | \mathbf{x}) = \begin{cases} 1 & \text{if } k = \text{argmax}_j f_j(\mathbf{x}) \\ 0 & \text{otherwise} \end{cases}$$

f: generalize to any function in x
not limited to linear

$$= \lim_{base \rightarrow \infty} \frac{base^{f_k(\mathbf{x})}}{\sum_j base^{f_j(\mathbf{x})}}$$

Function w/ highest value increase dramatically when taking exponent compared to all other smaller fns
In the limit, base \rightarrow infinity, converges to 1 for fn of highest value & 0 for all others

$$\approx \frac{e^{f_k(\mathbf{x})}}{\sum_j e^{f_j(\mathbf{x})}}$$

(softmax approximation)

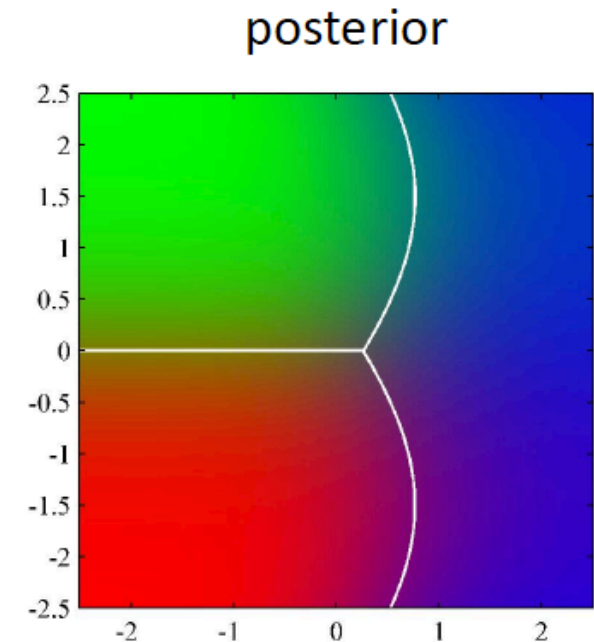
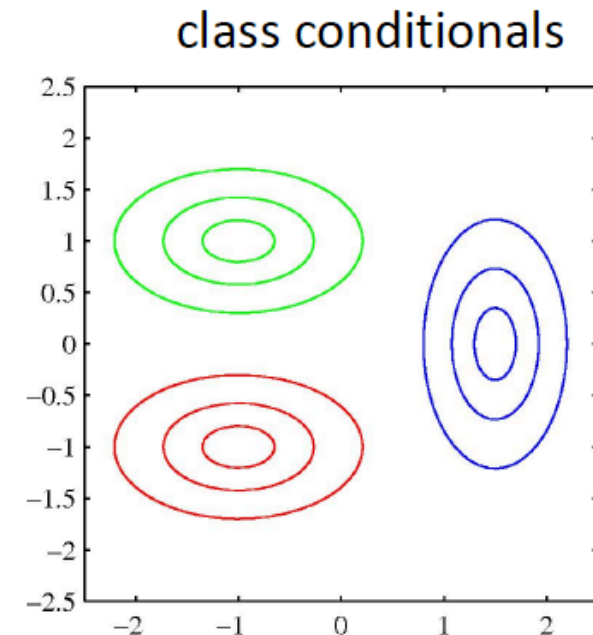
To soften (avoid 1 & 0, want something in-between (softer)), consider a finite base e.g., e, we get a spot version of argmax

NN: want to use a given function to determine output class (give highest probability to class of highest fn)
 \rightarrow use exponential trick

Consider Mixture of Gaussians \rightarrow computer posterior \rightarrow softmax

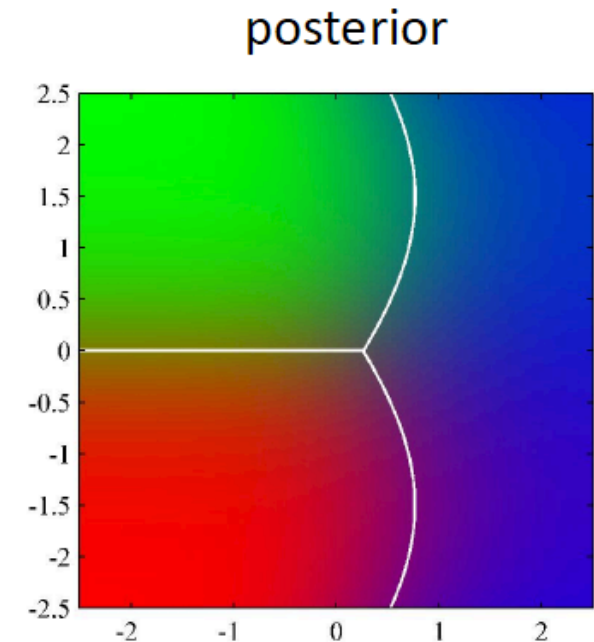
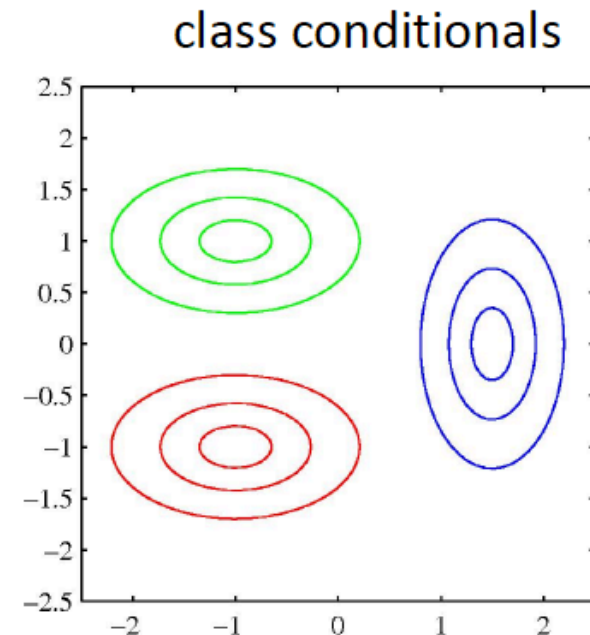
Softmax

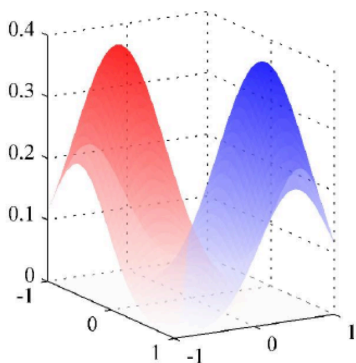
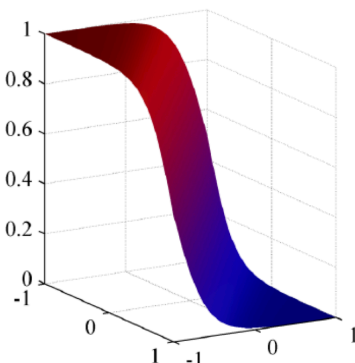
- Boundaries in posterior for 3 classes
- Green & red: same covariance matrix
- Blue: different covariance matrix
- Covariance matrix determines shape of gaussian
- To compute boundary line / curve
- 2 classes will dominate with similar probability
- Figure shows regions where each class dominates
- Similar covariance matrix \rightarrow linear separator
- Different covariance matrix \rightarrow nonlinear separator
- Why? (Hint: derivation!)

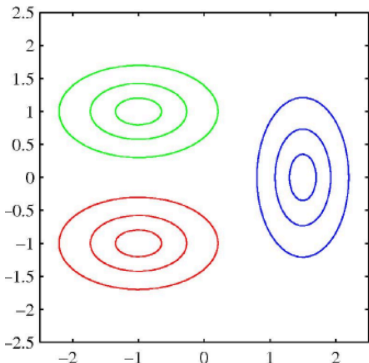
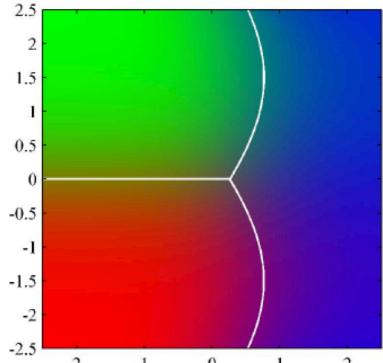


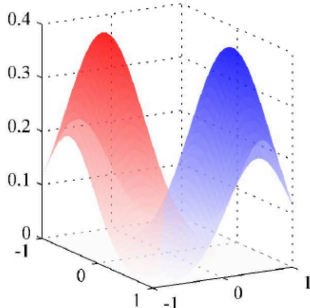
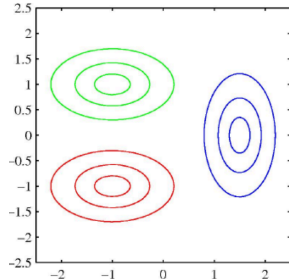
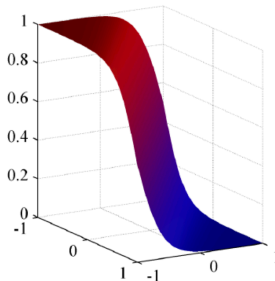
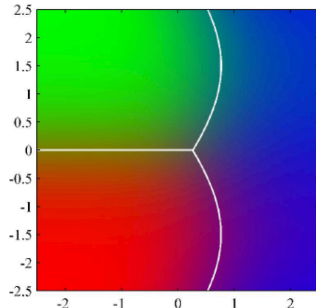
Softmax

- Boundaries in posterior for 3 classes
- Green & red: same covariance matrix
- Blue: different covariance matrix
- Covariance matrix determines shape of gaussian
- Similar covariance matrix \rightarrow linear separator
- Different covariance matrix \rightarrow nonlinear separator
- Derivation:
 - Same covariance matrix for all classes \rightarrow allowed simplifying expression to linear
 - Different covariance matrix \rightarrow expression is quadratic in x
- For a linear separator \rightarrow assume same covariance matrix for all classes

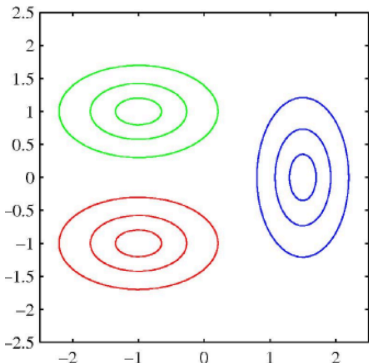
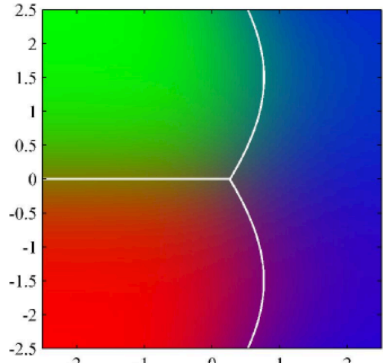


Model Components	1) Prior P(Ci) “Learning”	2) Likelihood P(x C) “Learning”	3) Posterior P(C x) “Inference”
Definition	Prior distribution assumed; without prior knowledge of data	Class conditional distribution of x given class, likelihood to observe x?	Posterior probability of class C given an input x
Computation			Bayes: posterior=prior x likelihood $\Pr(C \mathbf{x}) = \frac{\Pr(\mathbf{x} C) \Pr(C)}{\sum_C \Pr(\mathbf{x} C) \Pr(C)}$
Assumptions & Derivation	prior: multinomial distribution $\Pr(C = c_k) = \pi_k$ classes: finite, categorical	likelihood: Gaussian distribution $\Pr(\mathbf{x} c_k) \propto e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}$ data: $\mathbf{x} \in \mathbb{R}^d$ covariance matrix $\boldsymbol{\Sigma}$ same for c1, c2	$\Pr(c_k \mathbf{x}) = \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}}{\sum_k \pi_k e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}}$ $= \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x}^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - 2 \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k)}}{\sum_k \pi_k e^{-\frac{1}{2}(\mathbf{x}^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - 2 \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k)}}$ Consider two classes c_k and c_j $= \frac{1}{1 + \frac{\pi_j e^{\boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j}}{\pi_k e^{\boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k}}}}$ $= \frac{1}{1 + e^{-\left(\boldsymbol{\mu}_k^T - \boldsymbol{\mu}_j^T\right) \boldsymbol{\Sigma}^{-1} \mathbf{x} + \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j - \ln \frac{\pi_k}{\pi_j}}}}$
Inference: Binary Classification			Posterior: Logistic Sigmoid $\Pr(c_k \mathbf{x}) = \frac{1}{1 + e^{-(\mathbf{w}^T \mathbf{x} + w_0)}}$ $\mathbf{w} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_j)$ $w_0 = -\frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \frac{1}{2} \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j + \ln \frac{\pi_k}{\pi_j}$
Plot			

Model Components	1) Prior P(Ci) “Learning”	2) Likelihood P(x C) “Learning”	3) Posterior P(C x) “Inference”
Definition	Prior distribution assumed; without prior knowledge of data	Class conditional distribution of x given class, likelihood to observe x?	Posterior probability of class C given an input x
Computation			Bayes: posterior=prior x likelihood $\Pr(C \mathbf{x}) = \frac{\Pr(\mathbf{x} C) \Pr(C)}{\sum_C \Pr(\mathbf{x} C) \Pr(C)}$
Assumptions & Derivation	prior: multinomial distribution $\Pr(C = c_k) = \pi_k$ classes: finite, categorical	likelihood: Gaussian distribution $\Pr(\mathbf{x} c_k) \propto e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}$ data: $\mathbf{x} \in \mathbb{R}^d$ covariance matrix $\boldsymbol{\Sigma}$: similar c1, c2, different for c3	$ \begin{aligned} &= \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}}{\sum_j \pi_j e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)}} \\ &= \frac{\pi_k e^{-\frac{1}{2}(-2\boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k)}}{\sum_j \pi_j e^{-\frac{1}{2}(-2\boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j)}} \\ &= \frac{e^{\boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \ln \pi_k}}{\sum_j e^{\boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j + \ln \pi_j}} \end{aligned} $
Inference: Multi-class Classification			Posterior: Softmax $\Pr(c_k \mathbf{x}) = \frac{e^{\mathbf{w}_k^T \bar{\mathbf{x}}}}{\sum_j e^{\mathbf{w}_j^T \bar{\mathbf{x}}}}$ $\mathbf{w}_k^T = \left(-\frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \ln \pi_k, \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1}\right)$
Plot			

Model Components	Binary Classification	Multi-class Classification
Class Conditional Distribution	<p>2 similar conditional distribution covariance matrix</p> 	<p>1 different conditional distribution covariance matrix</p> 
Posterior Distribution	<p>Linear separator between classes of similar covariance matrix</p> 	<p>Linear separator between classes of similar covariance matrix</p> <p>Quadratic separator between classes of different covariance matrix</p> 
Posterior Distribution	<p>Logistic Sigmoid</p> $\Pr(c_k \mathbf{x}) = \frac{1}{1+e^{-(\mathbf{w}^T \mathbf{x} + w_0)}}$ $\mathbf{w} = \mathbf{\Sigma}^{-1}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_j)$ $w_0 = -\frac{1}{2}\boldsymbol{\mu}_k^T \mathbf{\Sigma}^{-1} \boldsymbol{\mu}_k + \frac{1}{2}\boldsymbol{\mu}_j^T \mathbf{\Sigma}^{-1} \boldsymbol{\mu}_j + \ln \frac{\pi_k}{\pi_j}$	<p>Softmax Distribution</p> $\Pr(c_k \mathbf{x}) = \frac{e^{f_k(\mathbf{x})}}{\sum_j e^{f_j(\mathbf{x})}}$
Argmax Distribution	$\Pr(c_k \mathbf{x}) = \begin{cases} c_1 & \sigma(\mathbf{w}^T \mathbf{x} + w_0) \geq 0.5 \\ c_2 & \text{otherwise} \end{cases}$	$\Pr(c_k \mathbf{x}) = \begin{cases} 1 & \text{if } k = \operatorname{argmax}_j f_j(\mathbf{x}) \\ 0 & \text{otherwise} \end{cases}$ $= \lim_{base \rightarrow \infty} \frac{base^{f_k(\mathbf{x})}}{\sum_j base^{f_j(\mathbf{x})}}$ $\approx \frac{e^{f_k(\mathbf{x})}}{\sum_j e^{f_j(\mathbf{x})}}$

Parameter Estimation

Model Components	1) Prior P(Ci) “Learning”	2) Likelihood P(x C) “Learning”	3) Posterior P(C x) “Inference”
Definition	Prior distribution assumed; without prior knowledge of data	Class conditional distribution of x given class, likelihood to observe x?	Posterior probability of class C given an input x
Computation			Bayes: posterior=prior x likelihood $\Pr(C \mathbf{x}) = \frac{\Pr(\mathbf{x} C) \Pr(C)}{\sum_C \Pr(\mathbf{x} C) \Pr(C)}$
Assumptions & Derivation	prior: multinomial distribution $\Pr(C = c_k) = \pi_k$ classes: finite, categorical	likelihood: Gaussian distribution $\Pr(\mathbf{x} c_k) \propto e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}$ data: $\mathbf{x} \in \mathbb{R}^d$ covariance matrix $\boldsymbol{\Sigma}$: similar c1, c2, different for c3	$= \frac{\pi_k e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)}}{\sum_j \pi_j e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_j)}}$ $= \frac{\pi_k e^{-\frac{1}{2}(-2\boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k)}}{\sum_j \pi_j e^{-\frac{1}{2}(-2\boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \mathbf{x} + \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j)}}$ $= \frac{e^{\boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \ln \pi_k}}{\sum_j e^{\boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \mathbf{x} - \frac{1}{2} \boldsymbol{\mu}_j^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_j + \ln \pi_j}}$
Inference: Multi-class Classification			Posterior: Softmax $\Pr(c_k \mathbf{x}) = \frac{e^{\mathbf{w}_k^T \bar{\mathbf{x}}}}{\sum_j e^{\mathbf{w}_j^T \bar{\mathbf{x}}}}$ $\mathbf{w}_k^T = (-\frac{1}{2} \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_k + \ln \pi_k, \boldsymbol{\mu}_k^T \boldsymbol{\Sigma}^{-1})$
Plot			

Classification Model: Mixture of Gaussian

- **Inference**

Compute posterior $\Pr(c_k|\mathbf{x})$ for 2 or multiple classes

- **Learning**

Estimate model parameters for:

- Prior probability $\Pr(c_k)$
- Conditional distribution parameters $\Pr(\mathbf{x}|c_k)$

Model Parameters

Learning	Prior Distribution Prior probability for each class	Likelihood/Conditional Distribution
Probability	$\Pr(c_k)$	$\Pr(\mathbf{x} c_k)$
Distribution (binary classification) Probability for each class	Multinomial (Binomial) $\Pr(c_1) = \pi,$ $\Pr(c_2) = 1 - \pi$	Gaussian $\Pr(\mathbf{x} c_1) \propto e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_1)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_1)}$ $\Pr(\mathbf{x} c_2) \propto e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu}_2)}$
Parameters	Probability of class 1 π	Mean $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$ Covariance matrix $\boldsymbol{\Sigma}$ Assume same

Estimate model parameters: GIVEN A DATASET, estimate π $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$ $\boldsymbol{\Sigma}$

Learning: Parameter Estimation

- Estimate parameters by
 - **Maximum likelihood**
 - Maximum a posteriori
 - Bayesian learning
- **Maximum likelihood:**
 - Simplest, most commonly used, especially with Mixtures of Gaussians

Maximum Likelihood Solution

Likelihood:

$$L(\mathbf{X}, \mathbf{y}) = \Pr(\mathbf{X}, \mathbf{y} | \pi, \mu_1, \mu_2, \Sigma) = \prod_n [\pi N(\mathbf{x}_n | \mu_1, \Sigma)]^{y_n} [(1 - \pi) N(\mathbf{x}_n | \mu_2, \Sigma)]^{1-y_n}$$

$y_n \in \{0, 1\}$

- Problem becomes: find parameters that maximize likelihood of data
- Define Likelihood function for dataset \mathbf{X}, \mathbf{y} given model parameters $\pi, \mu_1, \mu_2, \Sigma$
- For a given data point x ,
 - [1] If x belongs to c_1 , class conditional is Gaussian1 (mean 1)
 - [2] If x belongs to c_2 , class conditional is Gaussian2 (mean 2)
- Without knowledge of c , Likelihood should still use correct Gaussian (mean) in each c case
 - Define a single expression that combines [1] & [2]
 - Define Likelihood to use correct Gaussian expression
 - Label classes as 0 and 1
 - Use class as exponent
 - The correct Gaussian term is activated with the corresponding class
- This expression represents the probability of obtaining a point with a certain class
 - Given n data points in the given dataset, the likelihood is the product of all data points

Maximum Likelihood Solution

Likelihood:

$$L(\mathbf{X}, \mathbf{y}) = \Pr(\mathbf{X}, \mathbf{y} | \pi, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}) = \prod_n [\pi N(\mathbf{x}_n | \boldsymbol{\mu}_1, \boldsymbol{\Sigma})]^{y_n} [(1 - \pi) N(\mathbf{x}_n | \boldsymbol{\mu}_2, \boldsymbol{\Sigma})]^{1-y_n}$$

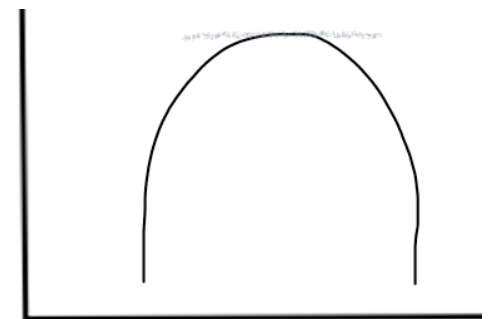
$y_n \in \{0, 1\}$

- Find parameters that maximize this likelihood
 - => optimization: maximize $L(\mathbf{X}, \mathbf{y})$
 - => maximizing a product (non-linear)
 - => complex optimization
 - => simplify
 - => log (expression)
 - => log does not change the maximum
 - => product becomes a sum

Maximum Likelihood Solution

$$\begin{aligned} &\text{ML hypothesis:} \\ &< \pi^*, \mu_1^*, \mu_2^*, \Sigma^* > = \\ &\operatorname{argmax}_{\pi, \mu_1, \mu_2, \Sigma} \sum_n y_n \left[\ln \pi - \frac{1}{2} (\mathbf{x}_n - \mu_1)^T \Sigma^{-1} (\mathbf{x}_n - \mu_1) \right] \\ &\quad + (1 - y_n) \left[\ln(1 - \pi) - \frac{1}{2} (\mathbf{x}_n - \mu_2)^T \Sigma^{-1} (\mathbf{x}_n - \mu_2) \right] \end{aligned}$$

- Optimization reduced Log-likelihood maximization
 - Log cancels the exponential
 - Resulting expression has a form that is concave
 - Concave optimization has a single global optimum
- Maximizing log-likelihood corresponds to a curve that has a concave shape
 - Log-likelihood maximization of concave shape
 - Easy optimization function
 - Set derivative to zero & isolate parameters
- Solve once for each parameter
 - Take derivative w.r.t. parameter
 - Set derivative to zero
 - Isolate parameter



Solve for π

ML hypothesis:

$$\langle \pi^*, \mu_1^*, \mu_2^*, \Sigma^* \rangle =$$

$$\operatorname{argmax}_{\pi, \mu_1, \mu_2, \Sigma} \sum_n y_n \left[\ln \pi - \frac{1}{2} (\mathbf{x}_n - \mu_1)^T \Sigma^{-1} (\mathbf{x}_n - \mu_1) \right] \\ + (1 - y_n) \left[\ln(1 - \pi) - \frac{1}{2} (\mathbf{x}_n - \mu_2)^T \Sigma^{-1} (\mathbf{x}_n - \mu_2) \right]$$

- Set derivative to 0

$$0 = \frac{\partial \ln L(\mathbf{X}, \mathbf{y})}{\partial \pi}$$

$$\Rightarrow 0 = \sum_n y_n \left[\frac{1}{\pi} \right] + (1 - y_n) \left[-\frac{1}{1 - \pi} \right]$$

$$\Rightarrow 0 = \sum_n y_n (1 - \pi) + (1 - y_n) (-\pi)$$

$$\Rightarrow \sum_n y_n = \pi (\sum_n y_n + \sum_n (1 - y_n))$$

$$\Rightarrow \sum_n y_n = \pi N \quad (\text{where } N \text{ is the \# of training points})$$

$$\therefore \frac{\sum_n y_n}{N} = \pi$$

← ? **Sample Mean**
(Empirical Mean)

Isolate π

π : probability of class 1

π : fraction of data of class 1

Without derivation, probability of a class corresponds to fraction of data belonging to the class
Which is what maximum likelihood estimator computes
Formal derivation confirms an intuitive approach
Not necessarily best estimate
But an estimate we can justify

Solve for μ_1, μ_2

ML hypothesis:
 $\langle \pi^*, \mu_1^*, \mu_2^*, \Sigma^* \rangle =$
$$\operatorname{argmax}_{\pi, \mu_1, \mu_2, \Sigma} \sum_n y_n \left[\ln \pi - \frac{1}{2} (\mathbf{x}_n - \mu_1)^T \Sigma^{-1} (\mathbf{x}_n - \mu_1) \right]$$
$$+ (1 - y_n) \left[\ln(1 - \pi) - \frac{1}{2} (\mathbf{x}_n - \mu_2)^T \Sigma^{-1} (\mathbf{x}_n - \mu_2) \right]$$

Y : target label given in training data

$$0 = \partial \ln L(\mathbf{X}, \mathbf{y}) / \partial \mu_1$$

$$\Rightarrow 0 = \sum_n y_n [-\Sigma^{-1} (\mathbf{x}_n - \mu_1)]$$

$$\Rightarrow \sum_n y_n \mathbf{x}_n = \sum_n y_n \mu_1$$

$$\Rightarrow \sum_n y_n \mathbf{x}_n = N_1 \mu_1$$

Isolate μ_1

Y_n : Label 1 \Rightarrow class 1

$$\therefore \frac{\sum_n y_n \mathbf{x}_n}{N_1} = \mu_1$$

Similarly:
$$\frac{\sum_n (1 - y_n) \mathbf{x}_n}{N_2} = \mu_2$$

**Sample Mean
(Empirical Mean)**

where N_1 is the # of data points in class 1

N_2 is the # of data points in class 2

This finds best values for both means that jointly maximizes likelihood
In other settings, we both means may be in the same term, then we must solve a system of linear equations

Solve for Σ

ML hypothesis:

$$< \pi^*, \mu_1^*, \mu_2^*, \Sigma^* > =$$

$$\operatorname{argmax}_{\pi, \mu_1, \mu_2, \Sigma} \sum_n y_n \left[\ln \pi - \frac{1}{2} (\mathbf{x}_n - \mu_1)^T \Sigma^{-1} (\mathbf{x}_n - \mu_1) \right] \\ + (1 - y_n) \left[\ln(1 - \pi) - \frac{1}{2} (\mathbf{x}_n - \mu_2)^T \Sigma^{-1} (\mathbf{x}_n - \mu_2) \right]$$

$$\frac{\partial \ln L(\mathbf{X}, \mathbf{y})}{\partial \Sigma} = 0$$

$\Rightarrow \dots$

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

where $\mathbf{S}_1 = \frac{1}{N_1} \sum_{n \in c_1} (\mathbf{x}_n - \mu_1)(\mathbf{x}_n - \mu_1)^T$

$$\mathbf{S}_2 = \frac{1}{N_2} \sum_{n \in c_2} (\mathbf{x}_n - \mu_2)(\mathbf{x}_n - \mu_2)^T$$

(\mathbf{S}_k is the empirical covariance matrix of class k)

Linear convex combination of empirical covariance matrices for each class weighted by number of points in each class