EECS 545: Machine Learning

Lecture 5. Classification 2

Honglak Lee

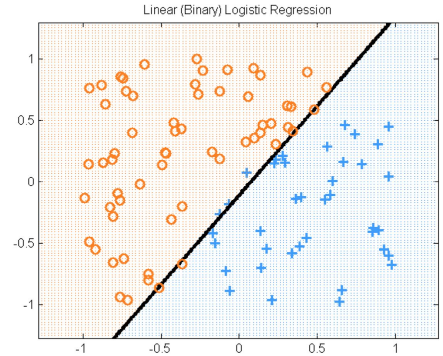
01/27/2020

Outline

- Softmax Regression
  - Multiclass extension of logistic regression
- Probabilistic generative models
  - Gaussian Discriminant Analysis

Softmax regression for multiclass classification

- For multiclass case, we can use softmax regression.
  - Softmax regression can be viewed as a generalization of logistic regression
- Recall that, logistic regression (binary classification) models class conditional probability as:



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

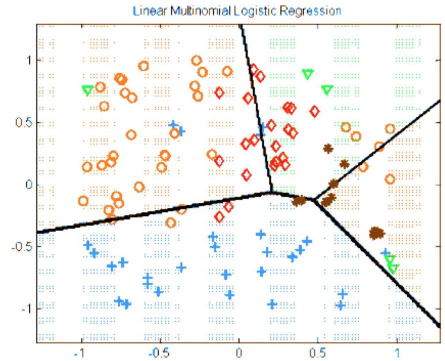
$$p(y=0|\mathbf{x};\mathbf{w}) = \frac{1}{1+\exp(\mathbf{w}^T\phi(\mathbf{x}))}$$

  - Note that these probability sum to 1.
- For multiclass classification (with K classes), we use the following model

$$p(y=k|\mathbf{x};\mathbf{w}) = \frac{\exp(\mathbf{w}_k^T\phi(\mathbf{x}))}{1+\sum_{j=1}^{K-1}\exp(\mathbf{w}_j^T\phi(\mathbf{x}))} \quad (fork = 1, \cdots, K-1)$$

$$p(y=K|\mathbf{x};\mathbf{w}) = \frac{1}{1+\sum_{j=1}^{K-1}\exp(\mathbf{w}_j^T\phi(\mathbf{x}))}$$

  - Note that these probability sum to 1.
  - This is equivalent when setting $\mathbf{w}_K = 0$



Softmax regression: Log-likelihood (objective function) and learning

- Defining $\mathbf{w}_K = 0$, we can write as:

$$p(y = k|\mathbf{x};\mathbf{w}) = \frac{\exp(\mathbf{w}_k^T \phi(\mathbf{x}))}{\sum_{j=1}^{K} \exp(\mathbf{w}_j^T \phi(\mathbf{x}))}$$

or $$p(y|\mathbf{x};\mathbf{w}) = \prod_{k=1}^{K} \left[\frac{\exp(\mathbf{w}_k^T \phi(\mathbf{x}))}{\sum_{j=1}^{K} \exp(\mathbf{w}_j^T \phi(\mathbf{x}))} \right]^{I(y=k)}$$

- Log-Likelihood

$$\log p(D|\mathbf{w}) = \sum_i \log p(y^{(i)}|\mathbf{x}^{(i)}, \mathbf{w})$$

$$= \sum_i \log \prod_{k=1}^{M} \left[\frac{\exp(\mathbf{w}_k^T \phi(\mathbf{x}^{(i)}))}{\sum_{j=1}^{M} \exp(\mathbf{w}_j^T \phi(\mathbf{x}^{(i)}))} \right]^{I(y^{(i)}=k)}$$

- We can learn parameters **w** by gradient ascent or Newton's method.

Probabilistic generative models

Learning the Classifier

- Goal: Learn the distributions $p(C_k\,|\mathbf{x})$.

(a) Discriminative models: Directly model $p(C_k|\mathbf{x})$ and learn parameters from the training set.

- Logistic regression
- Softmax regression

(b) Generative models: Learn class densities $p(\mathbf{x}|C_k)$ and priors $p(C_k)$

- Gaussian Discriminant Analysis
- Naive Bayes

Probabilistic Generative Models

- Bayes' theorem reduces the classification problem $p(C_k \mid \mathrm{x})$ to estimating the distribution of the data…

- Density estimation problems are easy to learn from labeled training data.
  - $p(C_k)$
  - $p(\mathbf{x} \mid C_k)$
- Maximum likelihood parameter estimation.

Probabilistic Generative Models

- For two classes, Bayes' theorem says:

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

- Use *log odds*:

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

- Then we can define the posterior via the *sigmoid*:

$$p(C_1|\mathbf{x}) = \frac{1}{1 + \exp(-a)} = \sigma(a)$$

Comparing the approaches: Discriminative vs. Generative

- The *generative* approach is typically model-based, and makes it possible to generate synthetic data from $p(\mathbf{x} \mid C_k)$.
  - By comparing the synthetic data and real data, we get a sense of how good the generative model is.

- The *discriminative* approach will typically have fewer parameters to estimate and have less assumptions about data distribution.
  - Linear (e.g., logistic regression) versus quadratic (e.g., Gaussian discriminant analysis) in the dimension of the input.
  - Less generative assumptions about the data (however, constructing the features may need prior knowledge)

Gaussian Discriminant Analysis

Gaussian Discriminant Analysis

- Prior distribution
  - $p(C_k)$: Constant (e.g., Bernoulli)
- Likelihood
  - $P(x|C_k)$: Gaussian distribution

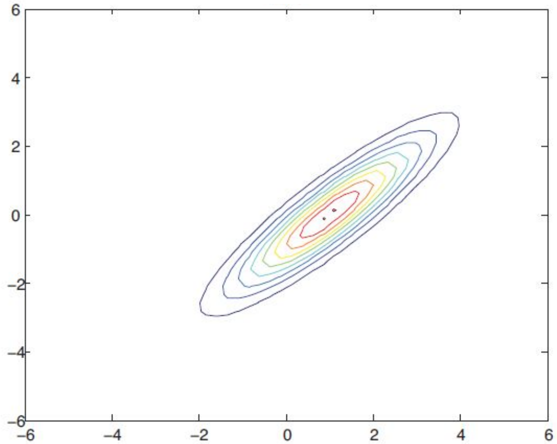
$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{ -\frac{1}{2}(\mathbf{x} - \mu_k)^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \mu_k) \right\}$$

- Classification: use Bayes rule (previous slide)

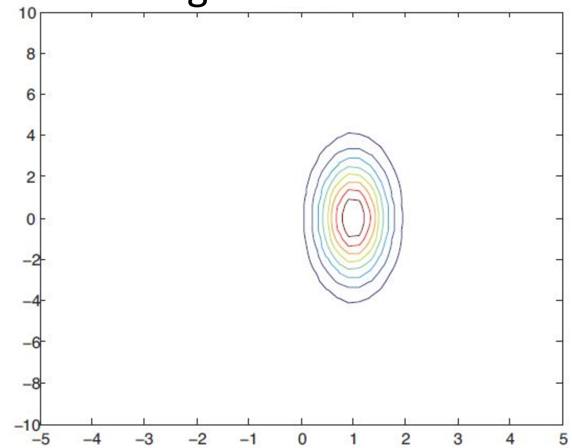
Examples of Gaussian Distributions

- Probability density p(x) for 2 dimensional case

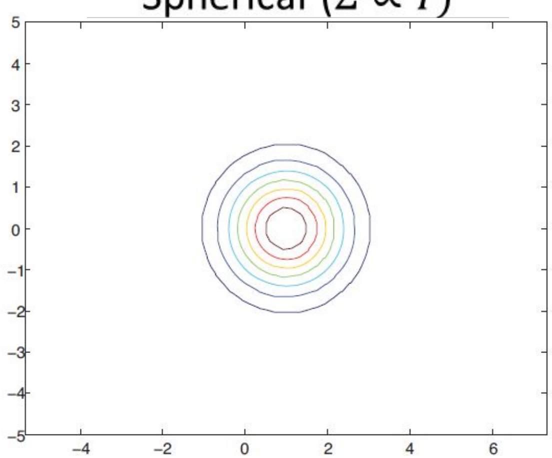
Full covariance



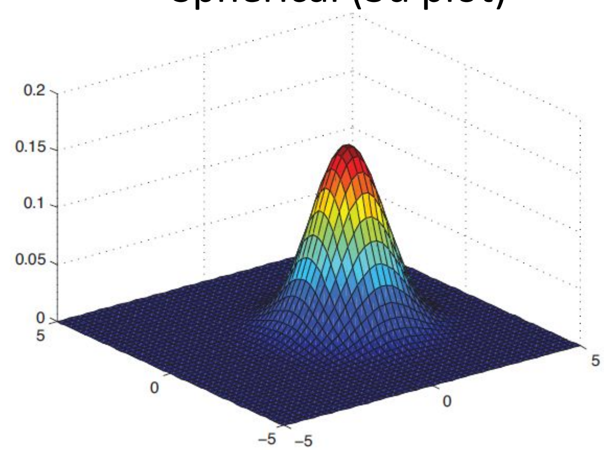
Diagonal covariance



Spherical ($\Sigma \propto I$)

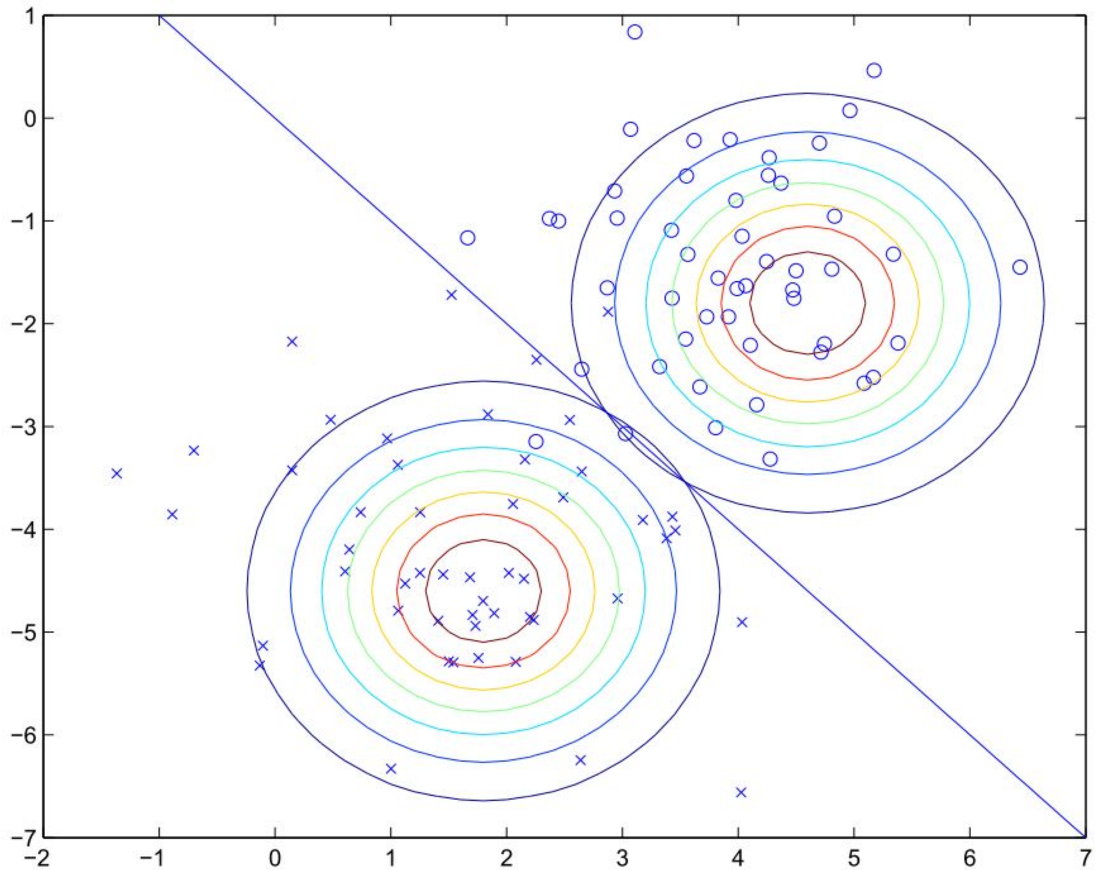


Spherical (3d plot)



Gaussian Discriminant Analysis

- Basic GDA assumes the same covariance for all classes
  - The below shows class-specific density and decision boundary
  - Note linear decision boundary!



Class-Conditional Densities

- Suppose we model $p(\mathrm{x} \mid C_k)$ as Gaussians with the <u>same covariance</u> matrix.

$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\boldsymbol{\Sigma}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu_k)^T \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\mu_k)\right\}$$

- This gives us $p(C_1|\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x} + w_0)$
  - where $\mathbf{w} = \boldsymbol{\Sigma}^{-1}(\mu_1 - \mu_2)$

    and

$$w_0 = -\frac{1}{2}\mu_1^T \boldsymbol{\Sigma}^{-1}\mu_1 + \frac{1}{2}\mu_2^T \boldsymbol{\Sigma}^{-1}\mu_2 + \ln\frac{p(C_1)}{p(C_2)}$$

Derivation

$$
\begin{aligned}
P(x, C_1) &= P(x|C_1)P(C_1) \\
&= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right\} P(C_1) \\
P(x, C_2) &= P(x|C_2)P(C_2) \\
&= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_2)^T\Sigma^{-1}(x-\mu_2)\right\} P(C_2)
\end{aligned}
$$

$$
\log\frac{P(C_1|x)}{P(C_2|x)} = \log\frac{P(C_1|x)}{1-P(C_1|x)}
$$

“Log-odds”

Derivation

$$\begin{aligned}
P(x, C_1) &= P(x|C_1)P(C_1) \\
&= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)\right\} P(C_1) \\
P(x, C_2) &= P(x|C_2)P(C_2) \\
&= \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_2)^T \Sigma^{-1} (x-\mu_2)\right\} P(C_2)
\end{aligned}$$

$$\log \frac{P(C_1|x)}{P(C_2|x)} = \log \frac{P(C_1|x)}{1 - P(C_1|x)}$$

"Log-odds"

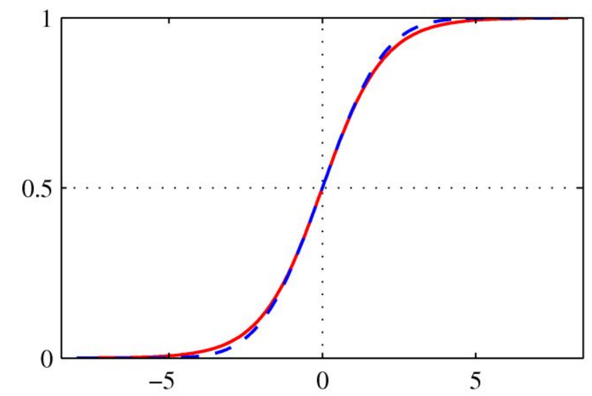
$$\begin{aligned}
&= \log \frac{\exp\left\{-\frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)\right\}}{\exp\left\{-\frac{1}{2}(x-\mu_2)^T \Sigma^{-1} (x-\mu_2)\right\}} + \log \frac{P(C_1)}{P(C_2)} \\
&= \left\{-\frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)\right\} - \left\{-\frac{1}{2}(x-\mu_2)^T \Sigma^{-1} (x-\mu_2)\right\} + \log \frac{P(C_1)}{P(C_2)} \\
&= (\mu_1 - \mu_2)^T \Sigma^{-1} x - \frac{1}{2}\mu_1^T \Sigma^{-1} \mu_1 + \frac{1}{2}\mu_2^T \Sigma^{-1} \mu_2 + \log \frac{P(C_1)}{P(C_2)} \\
&= \left(\Sigma^{-1}(\mu_1 - \mu_2)\right)^T x + w_0
\end{aligned}$$

$$\text{where } w_0 = -\tfrac{1}{2}\mu_1^T \Sigma^{-1} \mu_1 + \tfrac{1}{2}\mu_2^T \Sigma^{-1} \mu_2 + \log \tfrac{P(C_1)}{P(C_2)}$$

Class-Conditional Densities (for shared covariances)

- $P(C_k|x)$ is a sigmoid function:

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



  – with log-odds (*logit* function):

$$a = \log\left(\frac{\sigma}{1-\sigma}\right) = \left(\Sigma^{-1}(\mu_1 - \mu_2)\right)^T x + w_0$$

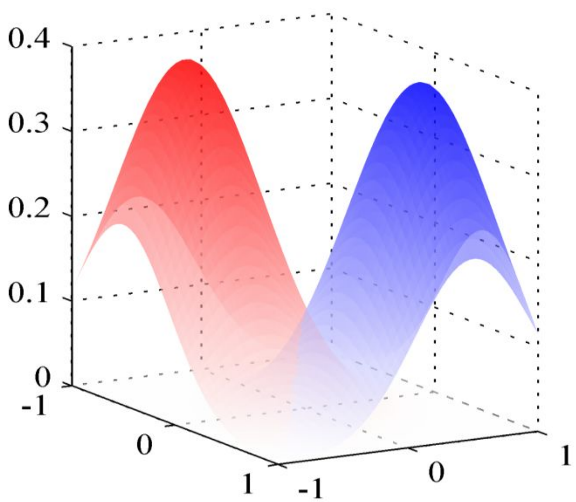
$$\text{where } w_0 = -\tfrac{1}{2}\mu_1\Sigma^{-1}\mu_1 + \tfrac{1}{2}\mu_2\Sigma^{-1}\mu_2 + \log\frac{P(C_1)}{P(C_2)}$$

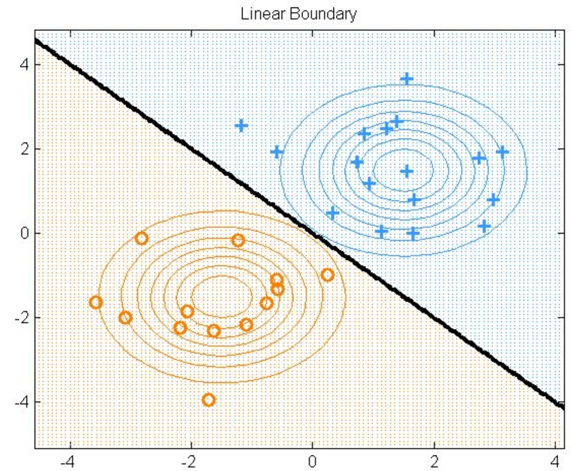
- Generalizes to *normalized exponential*, or *softmax*.

$$p_i = \frac{\exp(q_i)}{\sum_j \exp(q_j)}$$

Linear Decision Boundaries

- At decision boundary, we have $p(C_1 \mid x) = p(C_2 \mid x)$
- With the same covariance matrices, the boundary $p(C_1 \mid x) = p(C_2 \mid x)$ is linear.
  - Different priors $p(C_1)$, $p(C_2)$ just shift it around.





Learning parameters via maximum likelihood

- Given training data $\left\{(x^{(1)}, y^{(1)}), \cdots, (x^{(N)}, y^{(N)})\right\}$ and a generative model ("shared covariance")

$$p(y) = \phi^y (1-\phi)^{1-y}$$

$$p(\mathbf{x}|y=0) = \frac{1}{\sqrt{2\pi}\,|\Sigma|^{\frac{1}{2}}} \exp(-\frac{1}{2}(\mathbf{x}-\mu_0)^T \Sigma^{-1} (\mathbf{x}-\mu_0))$$

$$p(\mathbf{x}|y=1) = \frac{1}{\sqrt{2\pi}\,|\Sigma|^{\frac{1}{2}}} \exp(-\frac{1}{2}(\mathbf{x}-\mu_1)^T \Sigma^{-1} (\mathbf{x}-\mu_1))$$

Learning via maximum likelihood

- Maximum likelihood estimation (homework):

$$\phi = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}\{y^{(i)} = 1\}$$

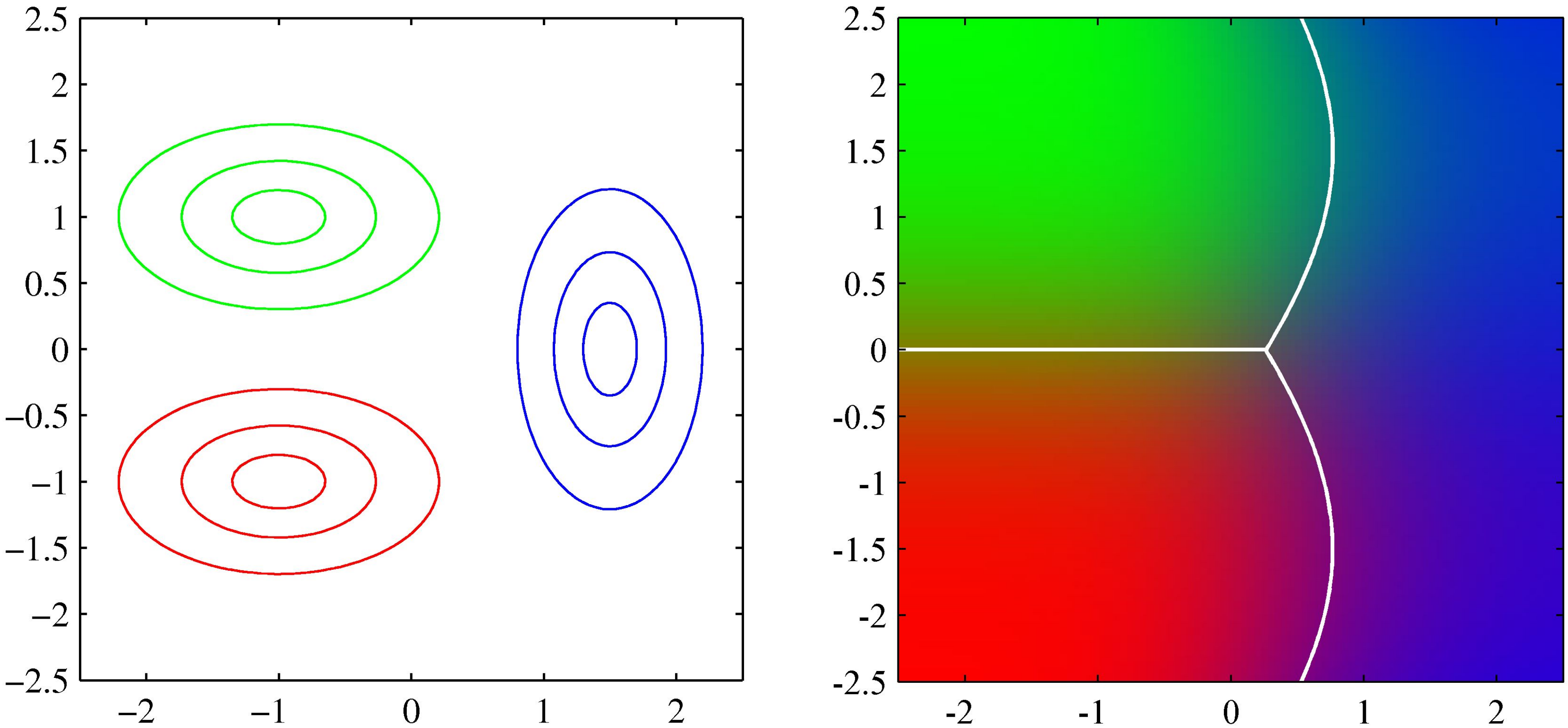
$$\mu_0 = \frac{\sum_{i=1}^{N} \mathbf{1}\{y^{(i)} = 0\}\mathbf{x}^{(i)}}{\sum_{i=1}^{N} \mathbf{1}\{y^{(i)} = 0\}}$$

$$\mu_1 = \frac{\sum_{i=1}^{N} \mathbf{1}\{y^{(i)} = 1\}\mathbf{x}^{(\mathrm{i})}}{\sum_{i=1}^{N} \mathbf{1}\{y^{(i)} = 1\}}$$

$$\sum = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}^{(i)} - \mu_{y^{(i)}})(\mathbf{x}^{(i)} - \mu_{y^{(i)}})^T$$

Different Covariance

- Decision boundaries can be quadratic when each class has different covariance.



Comparison between GDA and Logistic regression

- Logistic regression:
  - For an M-dimensional feature space, this model has M parameters to fit.
- Gaussian Discriminative Analysis
  - 2M parameters for the means of $p(\mathbf{x} \mid C_1)$ and $p(\mathbf{x} \mid C_2)$
  - M(M+1)/2 parameters for the shared covariance matrix
- Logistic regression has less parameters and is more flexible about data distribution.
- GDA has a stronger modeling assumption, and works well when the distribution follows the assumption.