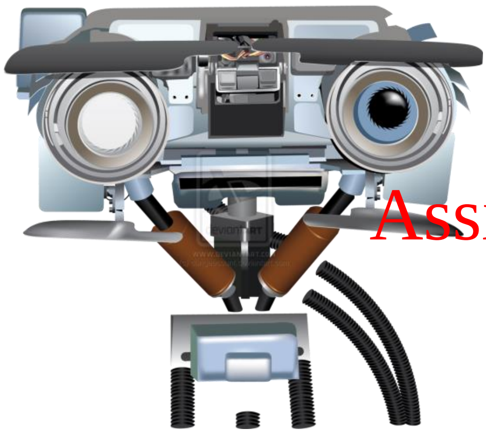
Announcements

Reminder: Midterm out on Thursday will be available on Blackboard (must be done by Friday)

90 minutes, must be completed after you start. Closed book.

Have scratch paper ready, some problems ask you to write out steps/show your work that can be shown on the scratch paper. Make sure to identify the problems you are showing your work for. Upload the scratch paper on the midterm form found on piazza right after you complete your test or it won't be counted!

- ps4 self-grading form out, due 10/30
- Lab this week – midterm review



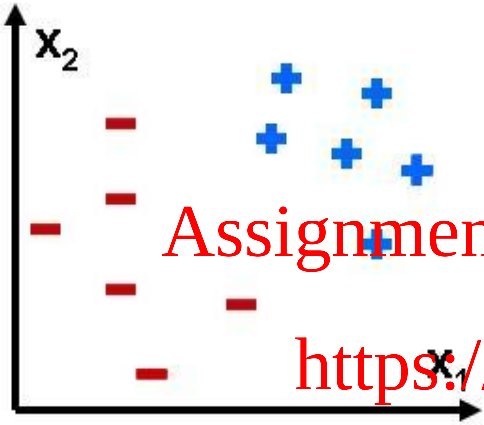
Probabilistic Generative Models

CS 542 Machine Learning

Today

- Probabilistic classification
- Linear Discriminant Analysis

Probabilistic Classification



$D = \left(x^{(i)}, y^{(i)}\right)$: data

$x \in \mathbb{R}^p$

$y \in \{k\}, k = 1, \dots, K$

- Can model output value directly, but having a probability is often more useful
- Bayes classifier: minimizes the probability of misclassification

$$y = \underset{k}{\operatorname{argmax}}\, p(Y = k | X = x)$$

- Want to model conditional distribution, $\mathrm{p}(\mathrm{Y} = \mathrm{y} | X = x)$, then assign label based on it

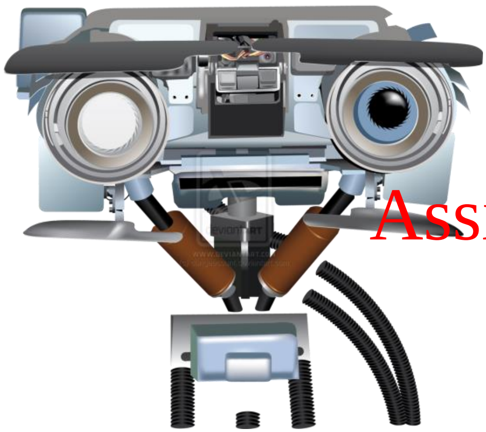
Two approaches to classification

- **Discriminative**: represent $\mathrm{p}(\mathrm{Y}|X)$ as function of parameters θ, then learn θ from training data

- **Generative**: use Bayes Rule to write

$$P\,(Y = k\,|\,X = x) = \frac{P(X = x\,|\,Y = k)P(Y = k)}{P(X = x)}$$

then learn parameters of class-conditional density $\mathrm{p}(\mathrm{X}|\mathrm{Y})$ and class prior $\mathrm{p}(\mathrm{Y})$ --- ignore $p(X)$



Generative vs Discriminative

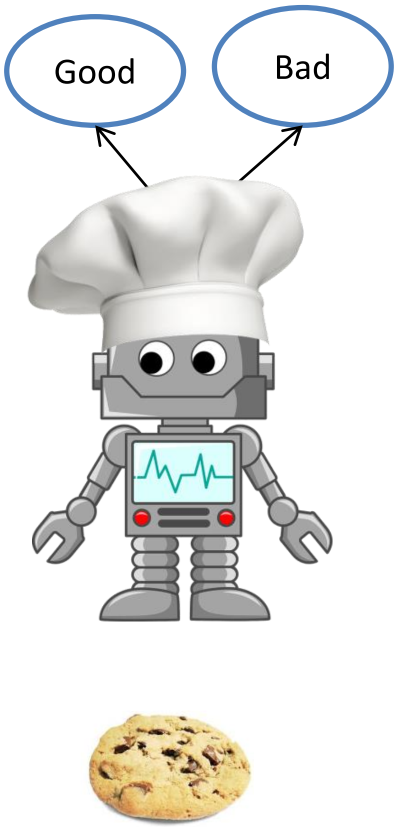
Intuition

Cookie Robots

- Suppose you own a cookie factory
- Want to detect bad cookies and discard them

Cookie Robots

$P(X|Y), P(Y)$ $\qquad\qquad$ $P(Y|X)$

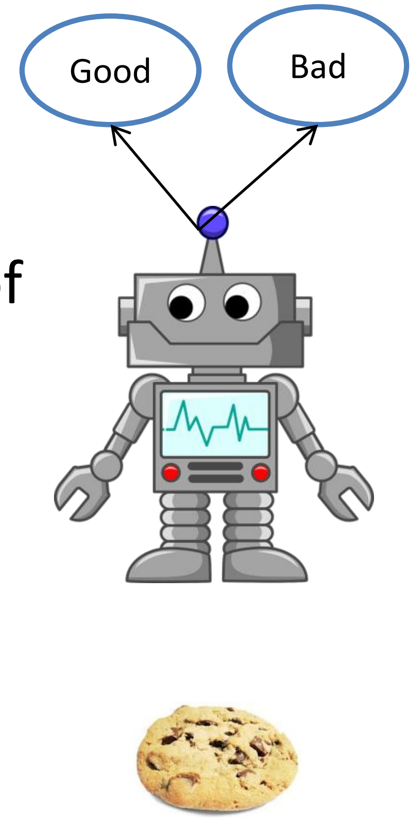


"The Chef"

- Can make good and bad cookies
- Compares new cookie to those
- Decides if it is good or bad

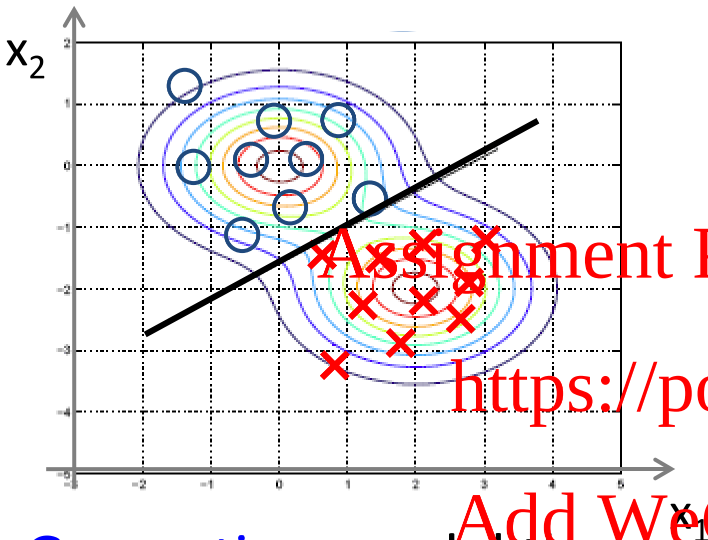
"The Critic"

- Cannot make cookies
- Has seen lots of good and bad cookies
- Decides if it is good or bad

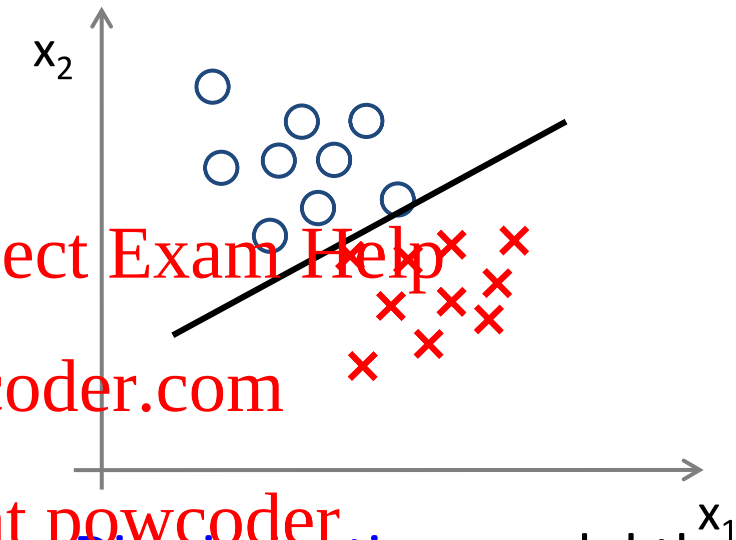


Generative vs Discriminative

$P(X|Y), P(Y)$



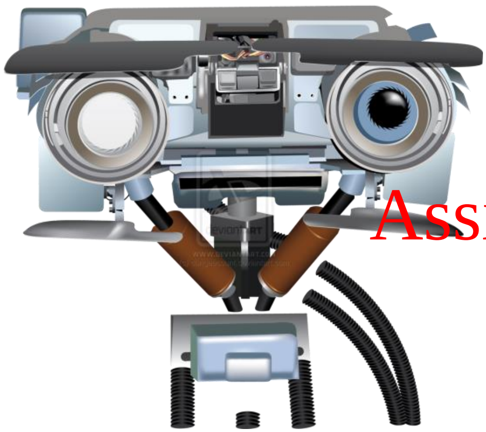
$P(Y|X)$



- Generative: model the class-conditional distribution of features, e.g. LDA, Naïve Bayes
- Discriminative: model the decision boundary directly, e.g. Logistic Regression, SVM

Can sample from distribution

Cannot sample from distribution



Linear Discriminant Analysis Derivation

Slide credits: Sergio Bacallado

Bayes Classifier

Find an estimate $\hat{P}(Y \mid X)$. Then, given an input x_0, we predict the output as in a Bayes classifier:

$$y_0 = argmax_y \hat{P}(Y = y_0 \mid X = x_0).$$

Generative Classifier

Instead of estimating $P(Y \mid X)$, we will estimate:

Generative Classifier

Instead of estimating $P(Y \mid X)$, we will estimate:

1. $P(X \mid Y)$: Given the category, what is the distribution of the inputs.

Generative Classifier

Instead of estimating $P(Y \mid X)$, we will estimate:

1. $P(X \mid Y)$: Given the category, what is the distribution of the inputs.
2. $P(Y)$: How likely are each of the categories.

Generative Classifier

Instead of estimating $P(Y \mid X)$, we will estimate:

1. $P(X \mid Y)$: Given the category, what is the distribution of the inputs.

2. $P(Y)$: How likely are each of the categories.

Then, we use *Bayes rule* to obtain the estimate:

$$P(Y = k \mid X = x) = \frac{P(X = x \mid Y = k)P(Y = k)}{P(X = x)}$$

Generative Classifier

Instead of estimating $P\,(Y \mid X)$, we will estimate:

1. $P\,(X \mid Y\,)$: Given the category, what is the distribution of the inputs.
2. $P\,(Y\,)$: How likely are each of the categories.

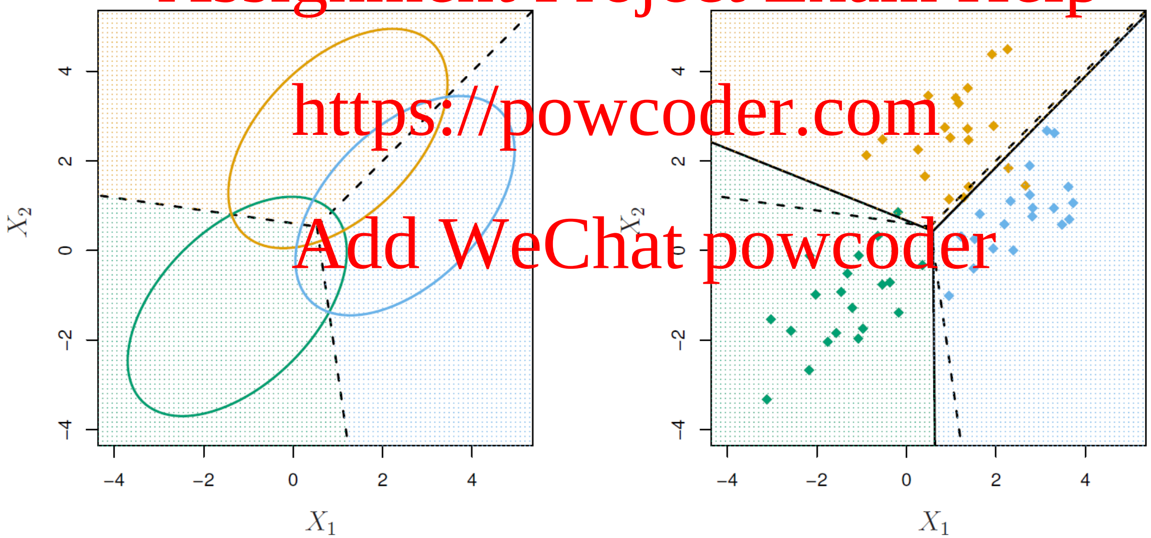
Then, we use *Bayes rule* to obtain the estimate:

$$P\,(Y = k \mid X = x) = \frac{P(X = x \mid Y = k)P(Y = k)}{\sum_j P(X = x \mid Y = j)P(Y = j)}$$

Linear Discriminant Analysis (LDA)

Instead of estimating $P\,(Y \mid X)$, we will estimate:

1. We model $P(X = x|Y = k) = f_k(x)$ as a *Multivariate Normal Distribution*:



2. $P(Y = k) = \pi_k$ is estimated by the fraction of training samples of class k.

LDA prior and class-conditional density

Suppose that:

LDA prior and class-conditional density

Suppose that:

- We know $P(Y = k) = \pi_k$ exactly.

LDA prior and class-conditional density

Suppose that:

- We know $P(Y = k) = \pi_k$ exactly.

- $P(X = x|Y = k)$ is Mutivariate Normal with density:

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \boldsymbol{\Sigma}^{-1}(x-\mu_k)}$$

LDA prior and class-conditional density

Suppose that:

- We know $P(Y = k) = \pi_k$ exactly.

- $P(X = x|Y = k)$ is Mutivariate Normal with density:

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

μ_k : Mean of the inputs for category k.

$\mathbf{\Sigma}$: Covariance matrix (common to all categories).

LDA prior and class-conditional density

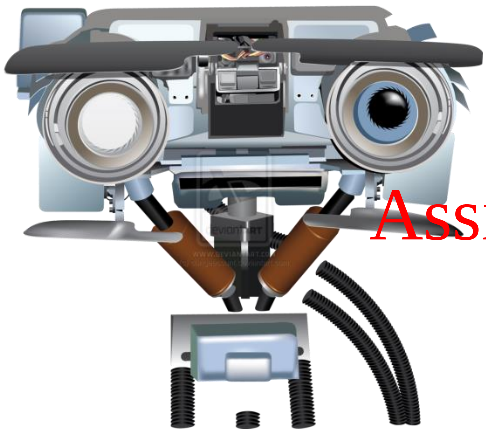
Suppose that:

- We know $P(Y = k) = \pi_k$ exactly.
- $P(X = x|Y = k)$ is Mutivariate Normal with density:

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\boldsymbol{\Sigma}^{-1}(x-\mu_k)}$$

μ_k : Mean of the inputs for category k.

$\boldsymbol{\Sigma}$: Covariance matrix (common to all categories).



LDA Solution

Slide credits: Sergio Bacallado

LDA has linear decision boundaries

By Bayes rule, the probability of category k, given the input x is:

$$P(Y = k \mid X = x) = \frac{f_k(x)\pi_k}{P(X = x)}$$

LDA has linear decision boundaries

By Bayes rule, the probability of category k, given the input x is:

$$P(Y = k \mid X = x) = \frac{f_k(x)\pi_k}{P(X = x)}$$

The denominator does not depend on the output k, so we can write it as a constant:

$$P(Y = k \mid X = x) = C \times f_k(x)\pi_k$$

LDA has linear decision boundaries

By Bayes rule, the probability of category k, given the input x is:

$$P(Y = k \mid X = x) = \frac{f_k(x)\pi_k}{P(X = x)}$$

The denominator does not depend on the output k, so we can write it as a constant:

$$P(Y = k \mid X = x) = C \times f_k(x)\pi_k$$

Now, expanding $f_k(x)$:

$$P(Y = k \mid X = x) = \frac{C\pi_k}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}^{-1}(x-\mu_k)}$$

LDA has linear decision boundaries

$$P(Y = k \mid X = x) = \frac{C\pi_k}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}^{-1}(x-\mu_k)}$$

LDA has linear decision boundaries

$$P(Y = k \mid X = x) = \frac{C\pi_k}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}^{-1}(x-\mu_k)}$$

Now, let us absorb everything that does not depend on k into a constant C':

$$P(Y = k \mid X = x) = C'\pi_k e^{-\frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}^{-1}(x-\mu_k)}$$

LDA has linear decision boundaries

$$P(Y = k \mid X = x) = \frac{C\pi_k}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

Now, let us absorb everything that does not depend on k into a constant C':

$$P(Y = k \mid X = x) = C'\pi_k e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

and take the logarithm of both sides:

$$\log P(Y = k \mid X = x) = \log C' + \log \pi_k - \frac{1}{2}(x - \mu_k)^T \mathbf{\Sigma}^{-1}(x - \mu_k).$$

LDA has linear decision boundaries

$$P(Y = k \mid X = x) = \frac{C\pi_k}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

Now, let us absorb everything that does not depend on k into a constant C':

$$P(Y = k \mid X = x) = C'\pi_k e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

and take the logarithm of both sides:

$$\log P(Y = k \mid X = x) = \log C' + \log \pi_k - \frac{1}{2}(x - \mu_k)^T \mathbf{\Sigma}^{-1}(x - \mu_k).$$

This is the same for every category, k.

LDA has linear decision boundaries

$$P(Y = k \mid X = x) = \frac{C\pi_k}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

Now, let us absorb everything that does not depend on k into a constant C':

$$P(Y = k \mid X = x) = C'\pi_k e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

and take the logarithm of both sides:

$$\log P(Y = k \mid X = x) = \log C' + \log \pi_k - \frac{1}{2}(x - \mu_k)^T \mathbf{\Sigma}^{-1}(x - \mu_k).$$

This is the same for every category, k.

So we want to find the maximum of this over k.

LDA has linear decision boundaries

Goal, maximize the following over k:

$$\log \pi_k - \frac{1}{2}(x - \mu_k)^T \boldsymbol{\Sigma}^{-1}(x - \mu_k).$$

LDA has linear decision boundaries

Goal, maximize the following over k:

$$\log \pi_k - \frac{1}{2}(x - \mu_k)^T \boldsymbol{\Sigma}^{-1}(x - \mu_k).$$

$$= \log \pi_k - \frac{1}{2}\left[x^T \boldsymbol{\Sigma}^{-1} x + \mu_k^T \boldsymbol{\Sigma}^{-1} \mu_k \right] + x^T \boldsymbol{\Sigma}^{-1} \mu_k$$

LDA has linear decision boundaries

Goal, maximize the following over k:

$$
\begin{aligned}
&\log \pi_k - \frac{1}{2}(x - \mu_k)^T \boldsymbol{\Sigma}^{-1}(x - \mu_k). \\
&= \log \pi_k - \frac{1}{2}\left[x^T \boldsymbol{\Sigma}^{-1} x + \mu_k^T \boldsymbol{\Sigma}^{-1} \mu_k \right] + x^T \boldsymbol{\Sigma}^{-1} \mu_k \\
&= C'' + \log \pi_k - \frac{1}{2}\mu_k^T \boldsymbol{\Sigma}^{-1} \mu_k + x^T \boldsymbol{\Sigma}^{-1} \mu_k
\end{aligned}
$$

LDA has linear decision boundaries

Goal, maximize the following over k:

$$\log \pi_k - \frac{1}{2}(x - \mu_k)^T \boldsymbol{\Sigma}^{-1}(x - \mu_k).$$

$$= \log \pi_k - \frac{1}{2}\left[x^T \boldsymbol{\Sigma}^{-1} x + \mu_k^T \boldsymbol{\Sigma}^{-1} \mu_k \right] + x^T \boldsymbol{\Sigma}^{-1} \mu_k$$

$$= C'' + \log \pi_k - \frac{1}{2} \mu_k^T \boldsymbol{\Sigma}^{-1} \mu_k + x^T \boldsymbol{\Sigma}^{-1} \mu_k$$

We define the objective:

$$\delta_k(x) = \log \pi_k - \frac{1}{2} \mu_k^T \boldsymbol{\Sigma}^{-1} \mu_k + x^T \boldsymbol{\Sigma}^{-1} \mu_k$$

At an input x, we predict the output with the highest $\delta_k(x)$.

LDA has linear decision boundaries

What is the decision boundary? It is the set of points in which 2 classes do just as well:

$$\delta_k(x) = \delta_l(x)$$

LDA has linear decision boundaries

What is the decision boundary? It is the set of points in which 2 classes do just as well:

$$\delta_k(x) = \delta_l(x)$$

$$\log \pi_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k + x^T \mathbf{\Sigma}^{-1}\mu_k = \log \pi_l - \frac{1}{2}\mu_l^T \mathbf{\Sigma}^{-1}\mu_l + x^T \mathbf{\Sigma}^{-1}\mu_l$$

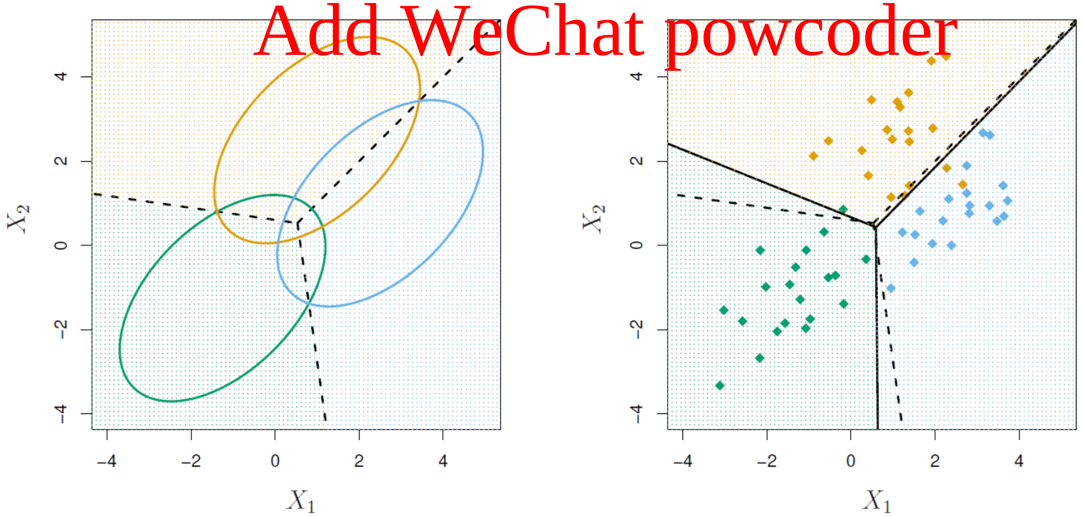
LDA has linear decision boundaries

What is the decision boundary? It is the set of points in which 2 classes do just as well:

$$\delta_k(x) = \delta_l(x)$$

$$\log \pi_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k + x^T \mathbf{\Sigma}^{-1}\mu_k = \log \pi_l - \frac{1}{2}\mu_l^T \mathbf{\Sigma}^{-1}\mu_l + x^T \mathbf{\Sigma}^{-1}\mu_l$$

This is a linear equation in x.



Estimating π_k

$$\pi_k = \frac{\#\{i \; ; y_i = k\}}{n}$$

In English, the fraction of training samples of class k.

Estimating the parameters of $f_k(x)$

Estimate the center of each class μ_k:

$$\mu_k = \frac{1}{\#\{i\,;\,y_i = k\}} \sum_{i\,;\,y_i=k} x_i$$

Estimating the parameters of $f_k(x)$

Estimate the center of each class μ_k:

$$\mu_k = \frac{1}{\#\{i; y_i = k\}} \sum_{i; y_i = k} x_i$$

Estimate the common covariance matrix $\mathbf{\Sigma}$:

Estimating the parameters of $f_k(x)$

Estimate the center of each class μ_k:

$$\mu_k = \frac{1}{\#\{i; y_i = k\}} \sum_{i; y_i = k} x_i$$

Estimate the common covariance matrix $\mathbf{\Sigma}$:

- One dimension ($p = 1$):

$$\sigma^2 = \frac{1}{n - K} \sum_{k=1}^{K} \sum_{i; y_i = k} (x_i - \mu_k)^2$$

Estimating the parameters of $f_k(x)$

Estimate the center of each class μ_k:

$$\mu_k = \frac{1}{\#\{i; y_i = k\}} \sum_{i; y_i = k} x_i$$

Estimate the common covariance matrix $\mathbf{\Sigma}$:

- One dimension ($p = 1$):

$$\sigma^2 = \frac{1}{n - K} \sum_{k=1}^{K} \sum_{i;\, y_i = k} (x_i - \mu_k)^2$$

- Many dimensions ($p > 1$): Compute the vectors of deviations $(x_1 - \mu_{y_1}), (x_2 - \mu_{y_2}), \ldots, (x_n - \mu_{y_n})$ and use an estimate of its covariance matrix, $\mathbf{\Sigma}$.

LDA prediction

For an input x, predict the class with the largest:

$$\delta_k(x) = \log \pi_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k + x^T \mathbf{\Sigma}^{-1}\mu_k$$

LDA prediction

For an input x, predict the class with the largest:

$$\delta_k(x) = \log \pi_k - \frac{1}{2}\mu_k^T \boldsymbol{\Sigma}^{-1}\mu_k + x^T \boldsymbol{\Sigma}^{-1}\mu_k$$

The decision boundaries are defined by:

$$\log \pi_k - \frac{1}{2}\mu_k^T \boldsymbol{\Sigma}^{-1}\mu_k + x^T \boldsymbol{\Sigma}^{-1}\mu_k = \log \pi_l - \frac{1}{2}\mu_l^T \boldsymbol{\Sigma}^{-1}\mu_l + x^T \boldsymbol{\Sigma}^{-1}\mu_l$$

LDA prediction

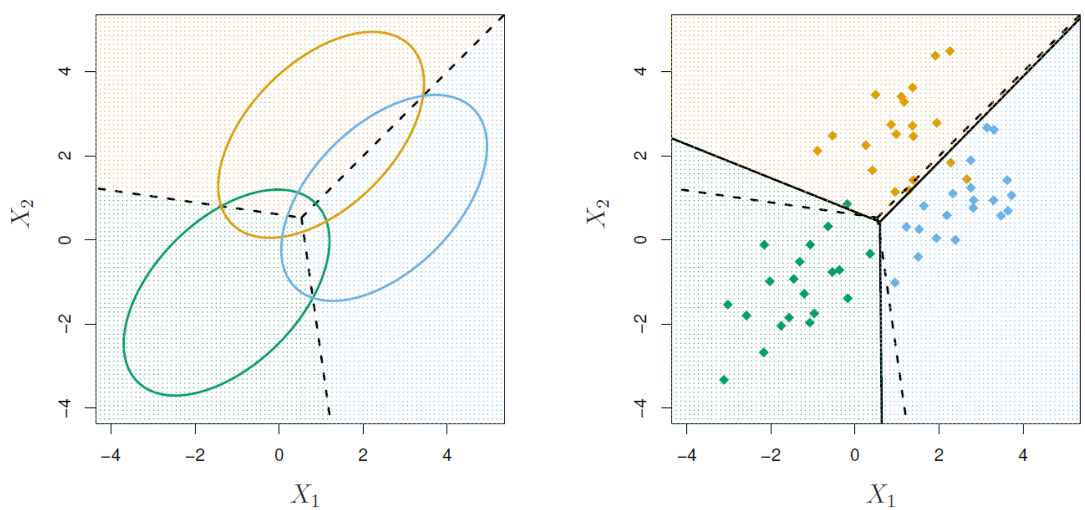
For an input x, predict the class with the largest:

$$\delta_k(x) = \log \pi_k - \frac{1}{2}\mu_k^T \boldsymbol{\Sigma}^{-1}\mu_k + x^T \boldsymbol{\Sigma}^{-1}\mu_k$$

The decision boundaries are defined by:

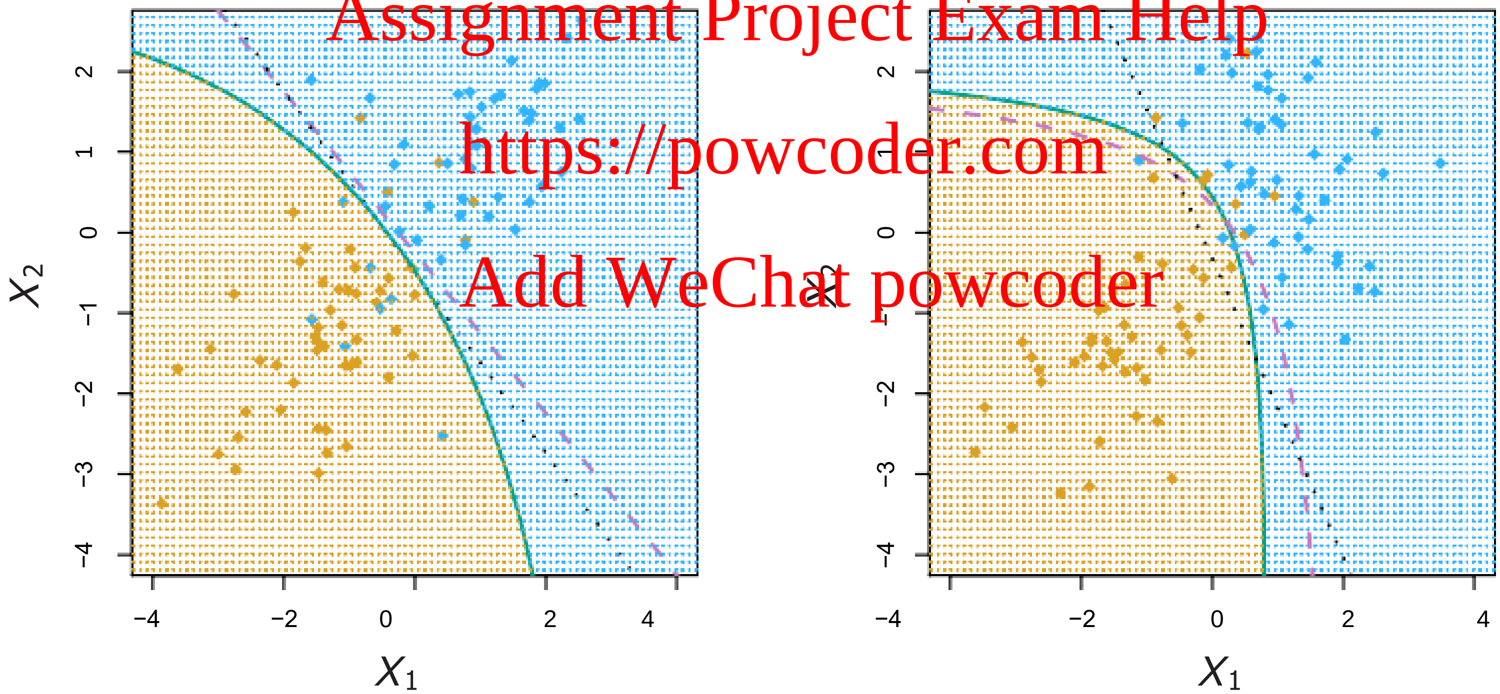
$$\log \pi_k - \frac{1}{2}\mu_k^T \boldsymbol{\Sigma}^{-1}\mu_k + x^T \boldsymbol{\Sigma}^{-1}\mu_k = \log \pi_l - \frac{1}{2}\mu_l^T \boldsymbol{\Sigma}^{-1}\mu_l + x^T \boldsymbol{\Sigma}^{-1}\mu_l$$

Solid lines in:



Quadratic discriminant analysis (QDA)

The assumption that the inputs of every class have the same covariance $\mathbf{\Sigma}$ can be quite restrictive:



Quadratic discriminant analysis (QDA)

In **quadratic discriminant analysis** we estimate a mean μ_k and a covariance matrix $\boldsymbol{\Sigma}_k$ for each class separately.

Quadratic discriminant analysis (QDA)

In **quadratic discriminant analysis** we estimate a mean μ_k and a covariance matrix $\boldsymbol{\Sigma}_k$ for each class separately.

Given an input, it is easy to derive an objective function:

$$\delta_k(x) = \log \pi_k - \frac{1}{2}\mu_k^T \boldsymbol{\Sigma}_k^{-1}\mu_k + x^T \boldsymbol{\Sigma}_k^{-1}\mu_k - \frac{1}{2}x^T \boldsymbol{\Sigma}_k^{-1} x - \frac{1}{2}\log|\boldsymbol{\Sigma}_k|$$

Quadratic discriminant analysis (QDA)

In **quadratic discriminant analysis** we estimate a mean μ_k and a covariance matrix $\boldsymbol{\Sigma}_k$ for each class separately.

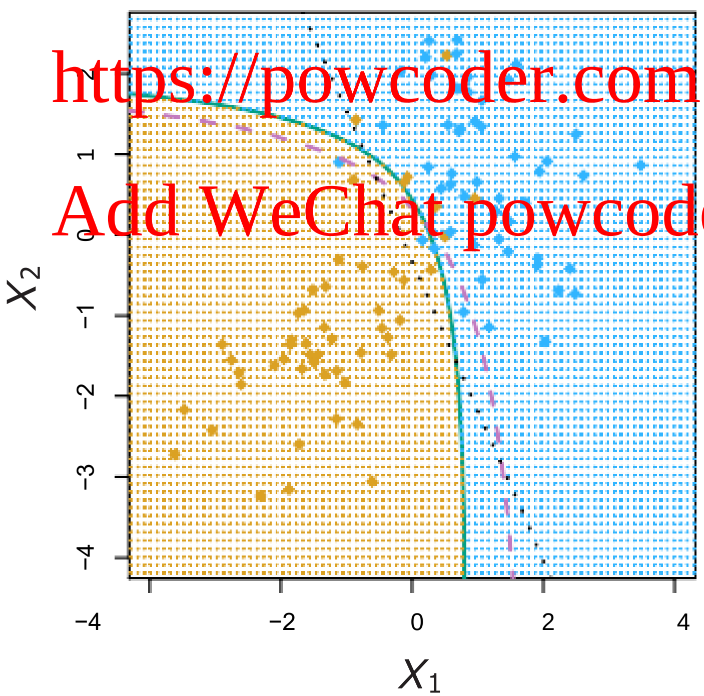
Given an input, it is easy to derive an objective function:

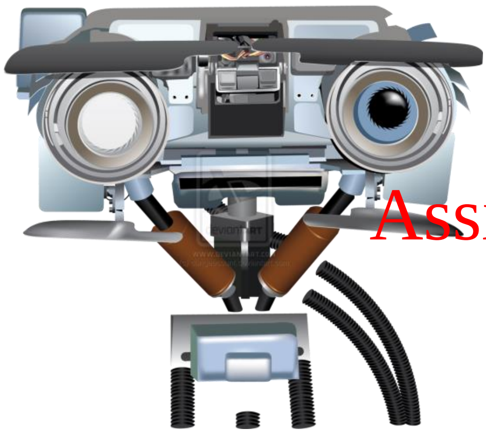
$$\delta_k(x) = \log \pi_k - \frac{1}{2}\mu_k^T \boldsymbol{\Sigma}_k^{-1}\mu_k + x^T \boldsymbol{\Sigma}_k^{-1}\mu_k - \frac{1}{2}x^T \boldsymbol{\Sigma}_k^{-1} x - \frac{1}{2}\log|\boldsymbol{\Sigma}_k|$$

This objective is now quadratic in x and so are the decision boundaries.

Quadratic discriminant analysis (QDA)

- Bayes boundary (– – –)
- LDA (· · · · · ·)
- QDA (——).





Linear Discriminant Analysis

Assignment Project Exam Help

Add WeChat powcoder

More intuition

Illustration of Decision Boundary

$$\log\frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k + \mu_l)^T \Sigma^{-1}(\mu_k - \mu_l) + x^T \Sigma^{-1}(\mu_k - \mu_l) = 0$$

class prior log-ratio

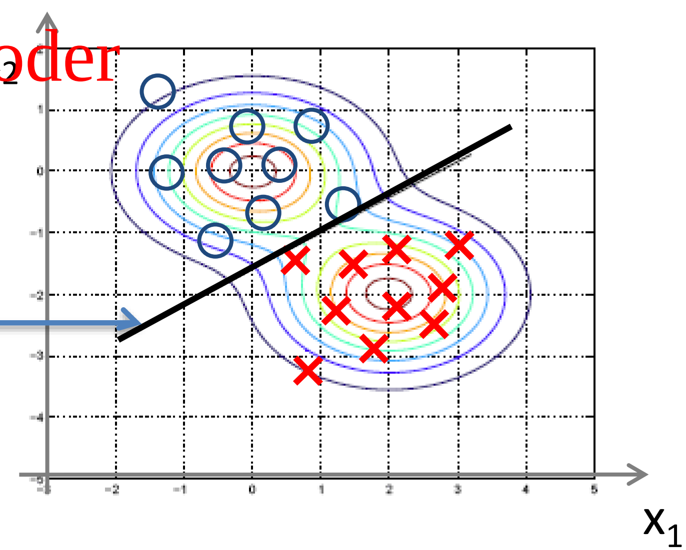
constant

input

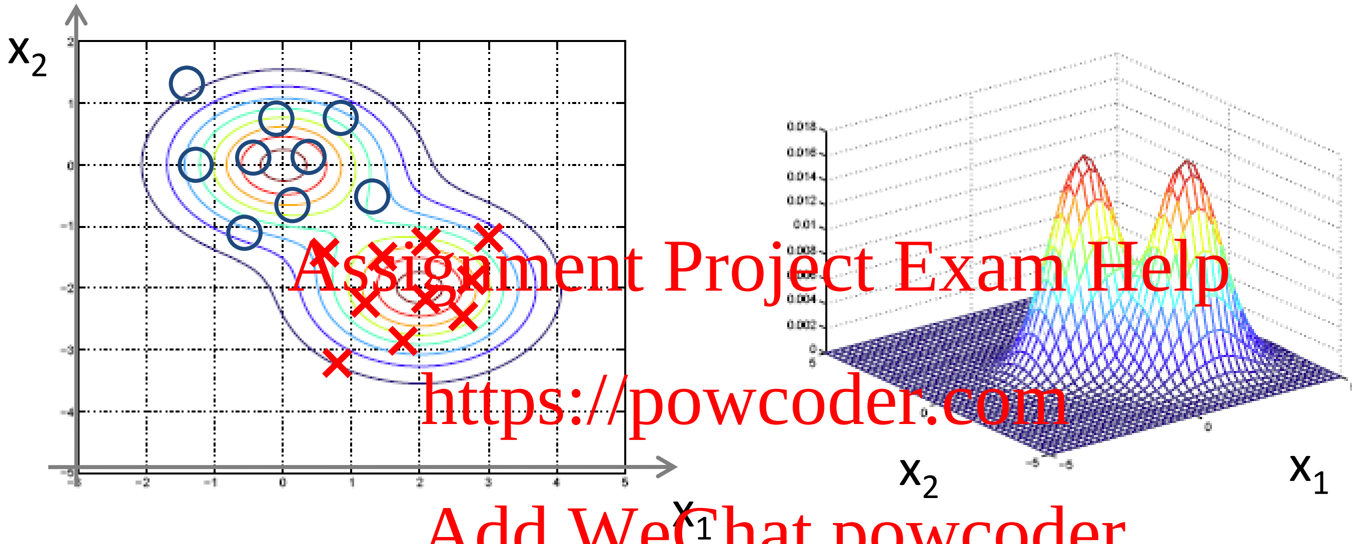
covariance

diff. in class means

Can re-write as $\theta_0 + x^T\theta = 0$



Effect of Covariance Matrix



- covariance matrix determines the shape of the Gaussian density, so
- in LDA, the Gaussian densities for different classes have the same shape, but are shifted versions of each other (different mean vectors).

Effect of Class Prior

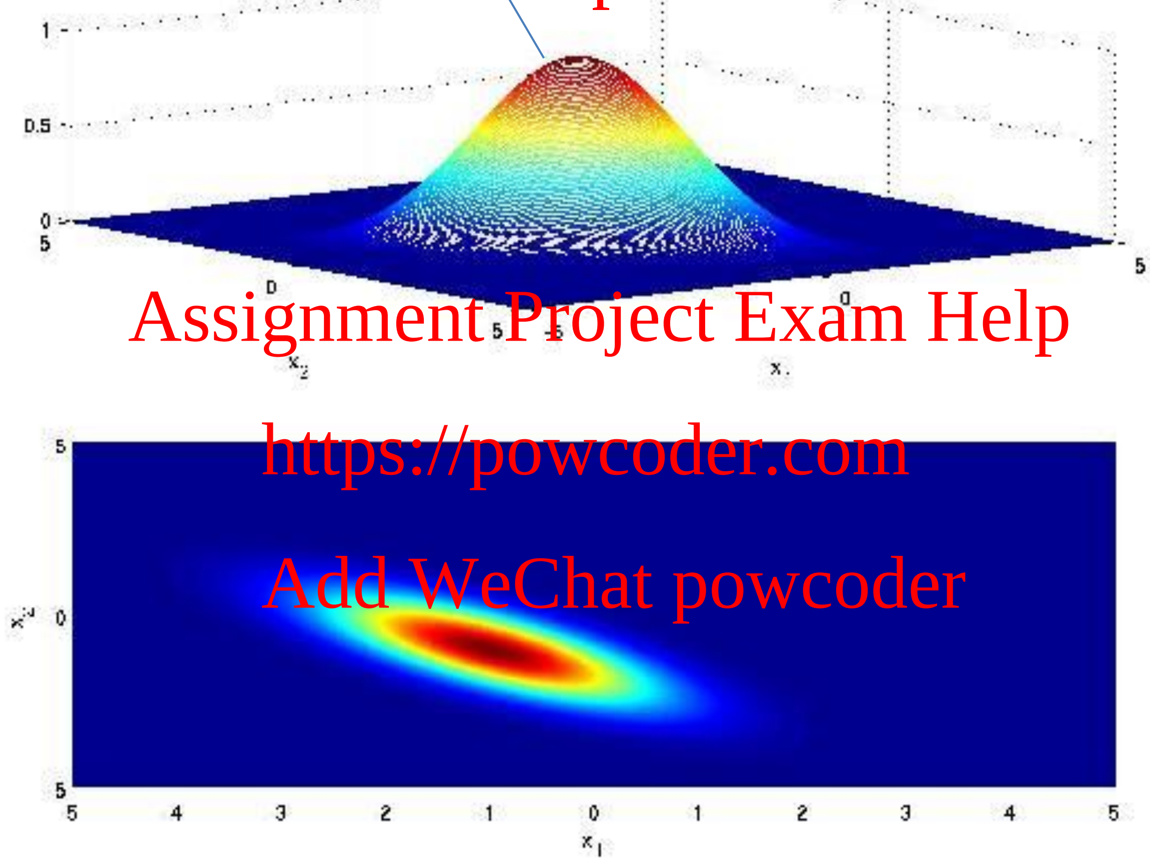
- What effect does the prior p(class), or π_k, have?

- Lets look at an example for 2 classes…

$$\log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k + \mu_l)^T \Sigma^{-1} (\mu_k - \mu_l) + x^T \Sigma^{-1} (\mu_k - \mu_l) = 0$$

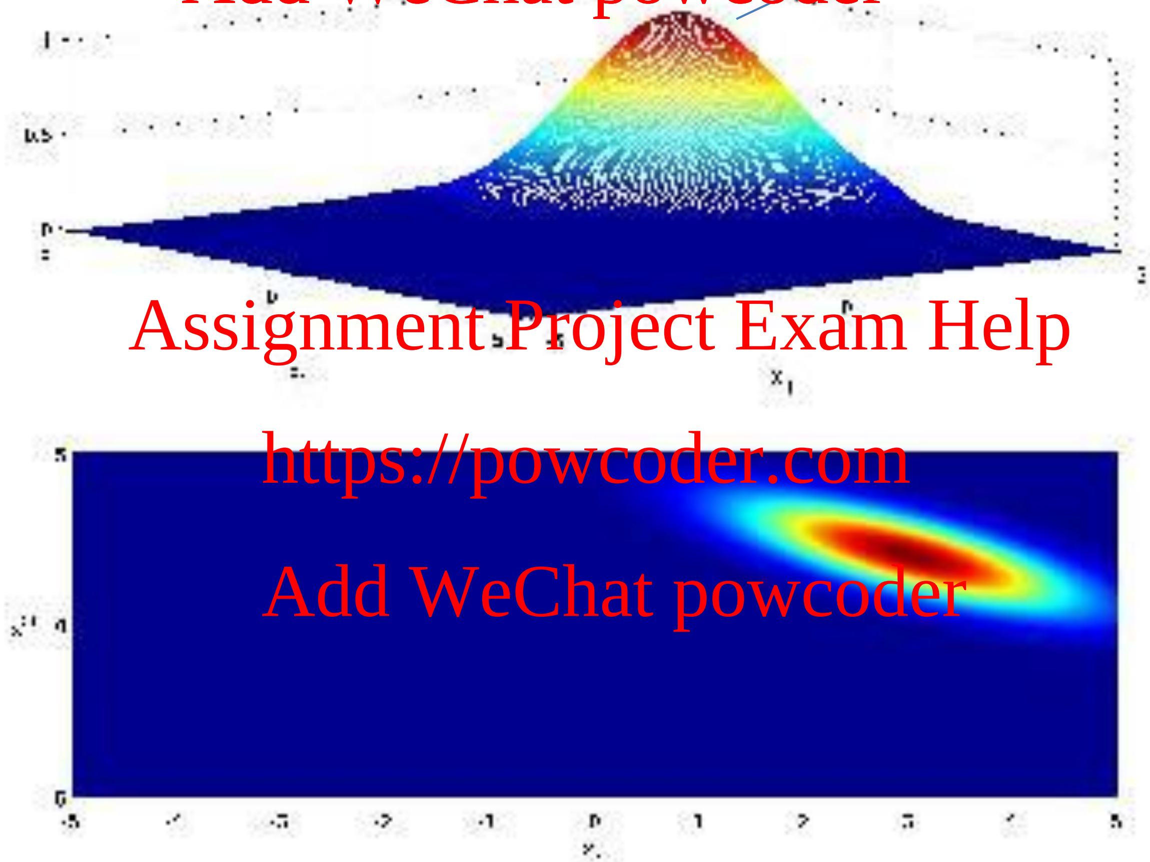
class prior
log-ratio

$p(C_1|\boldsymbol{x}) \propto p(\boldsymbol{x}|C_1)p(C_1)$



Model class-conditional probability of a 2D feature vector for class 1 as a multivariate Gaussian density.

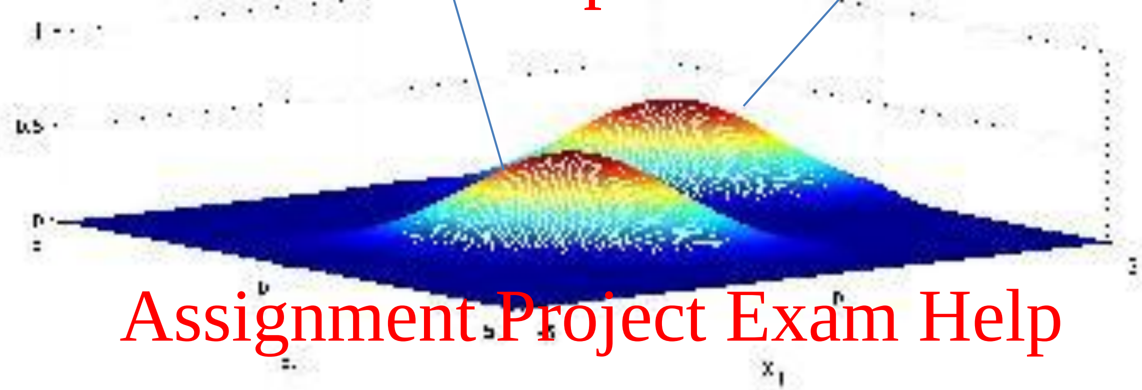
$$p(C_2|\boldsymbol{x}) \propto p(\boldsymbol{x}|C_2)p(C_2)$$



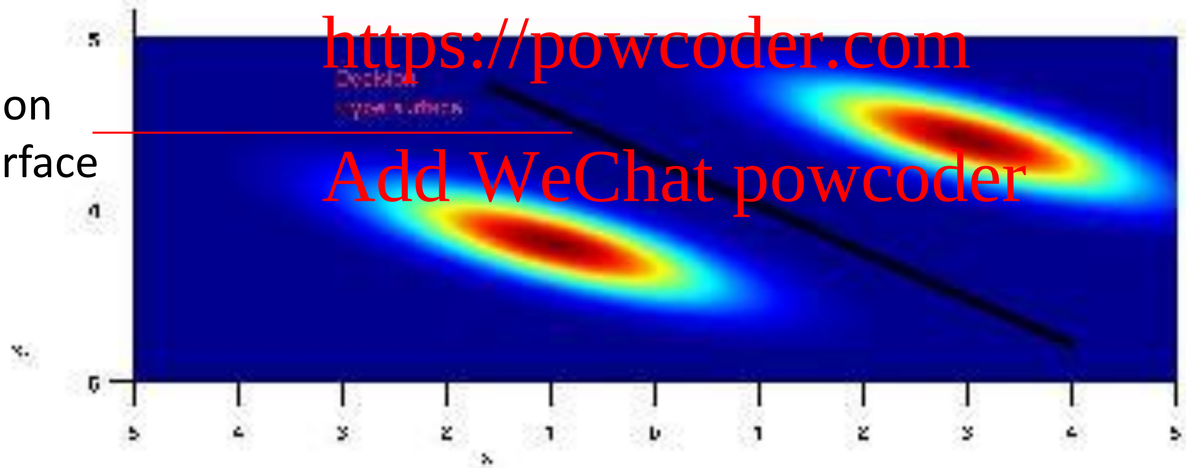
Now consider class 2 with a similar Gaussian conditional density, which has the same covariance but a different mean

$p(C_1|\boldsymbol{x}) \propto p(\boldsymbol{x}|C_1)p(C_1)$

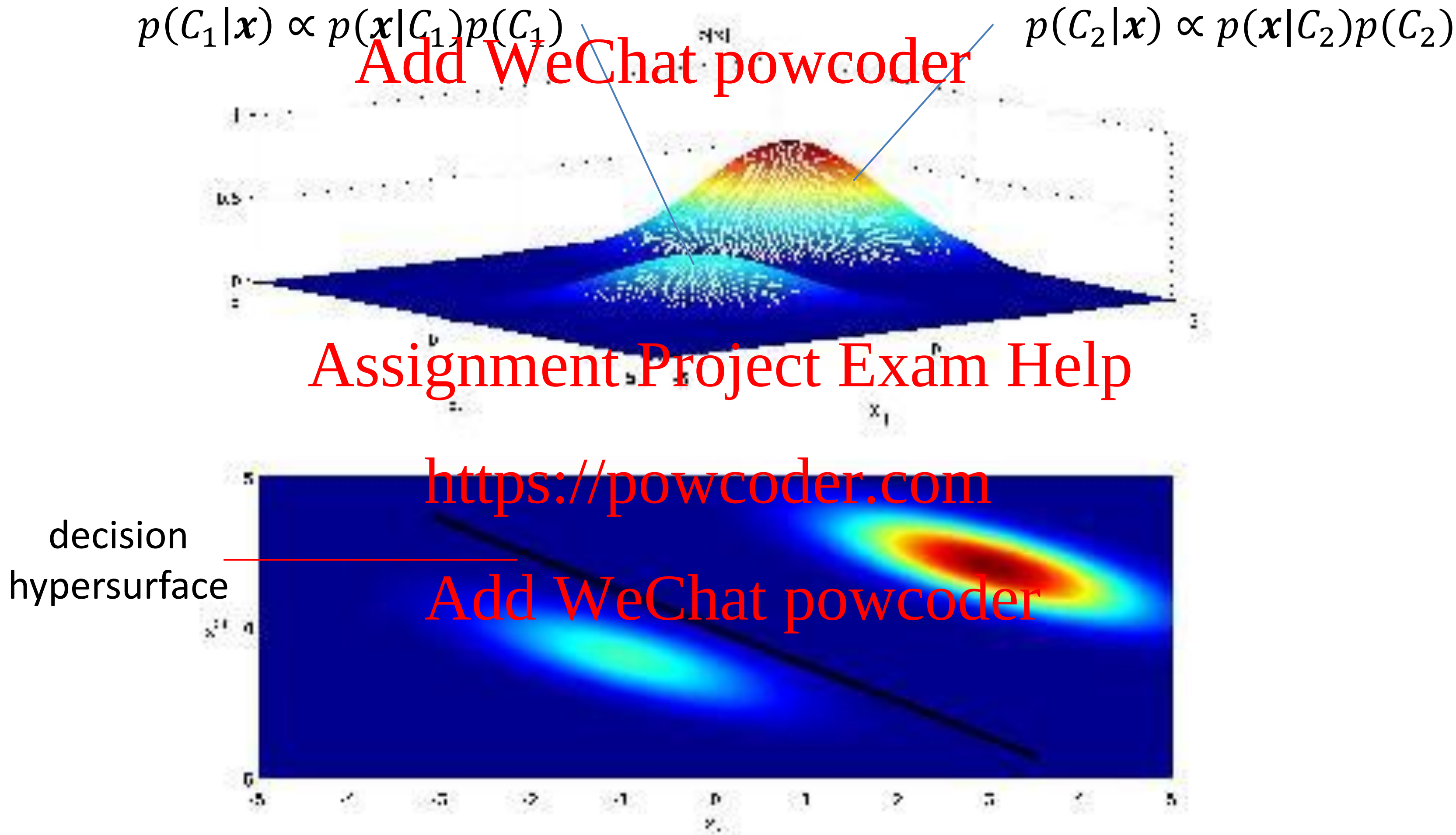
$p(C_2|\boldsymbol{x}) \propto p(\boldsymbol{x}|C_2)p(C_2)$



decision hypersurface

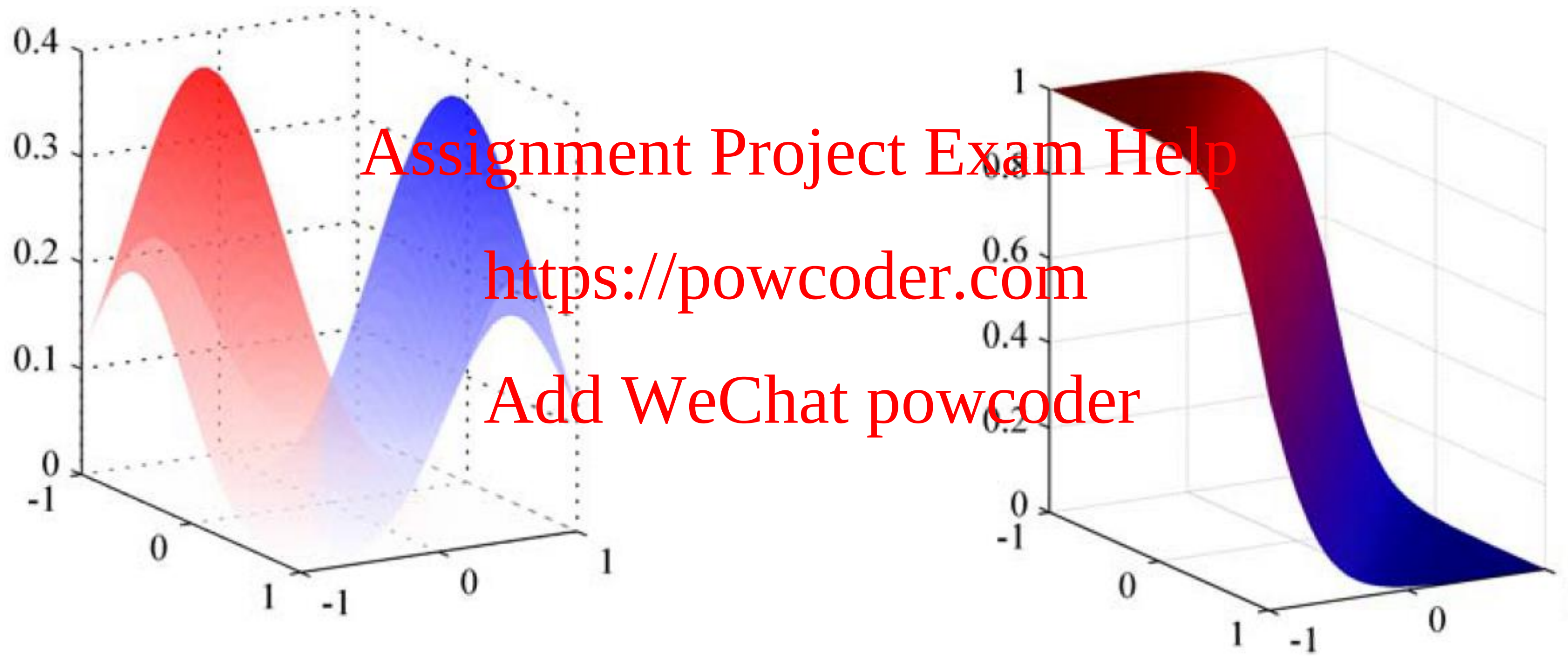


If the priors for each class are the same (i.e. 0.5), then the decision hypersurface cuts directly between the two means, with a direction parallel to the elliptical shape of the modes of the Gaussian densities shaped by their (identical) covariance matrices.



Now if the priors for each class are unequal, the decision hypersurface cuts between the two means with a direction as before, but now will be located further from the more likely class. This biases the predictor in favor of the more likely class.

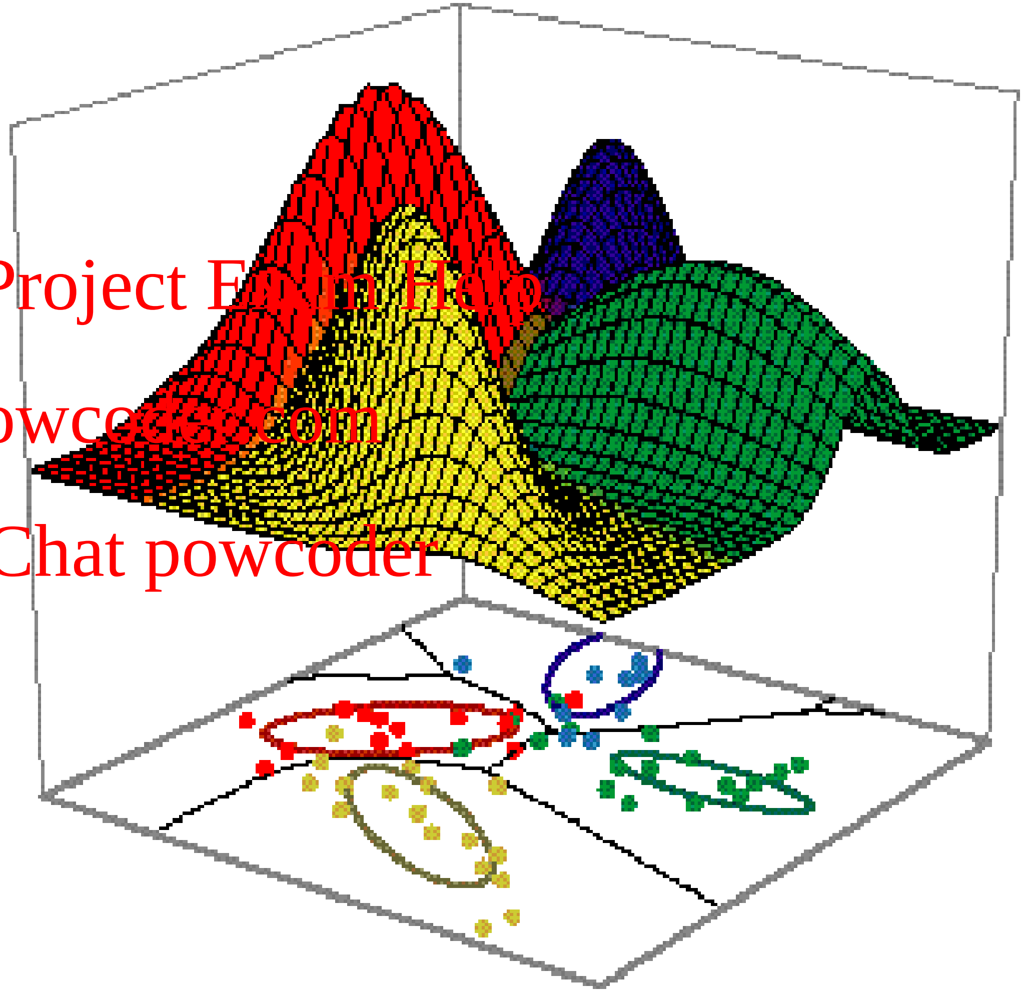
Posterior probability $p(C_1|x)$ for two classes C_1, C_2



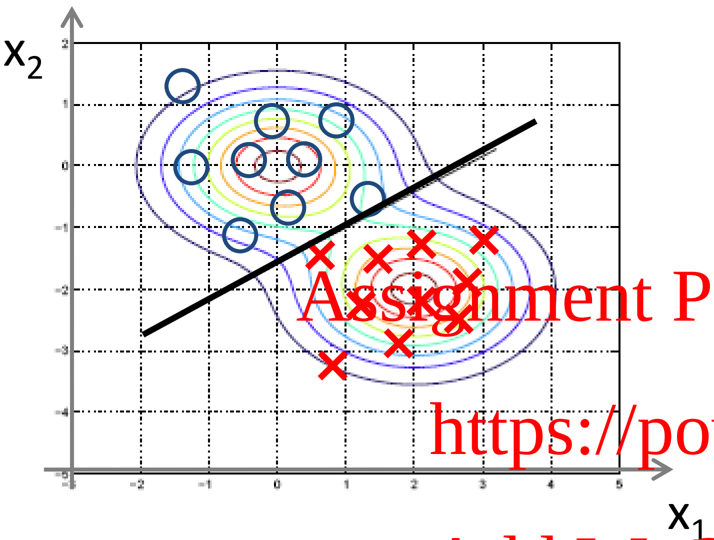
Bishop Figure 4.10 The left-hand plot shows the class-conditional densities for two classes, denoted red and blue. On the right is the corresponding posterior probability $p(C1|\mathbf{x})$, which is given by a logistic sigmoid of a linear function of $\mathbf{x}$. The surface in the right-hand plot is coloured using a proportion of red ink given by $p(C1|\mathbf{x})$ and a proportion of blue ink given by $p(C2|\mathbf{x}) = 1 - p(C1|\mathbf{x})$.

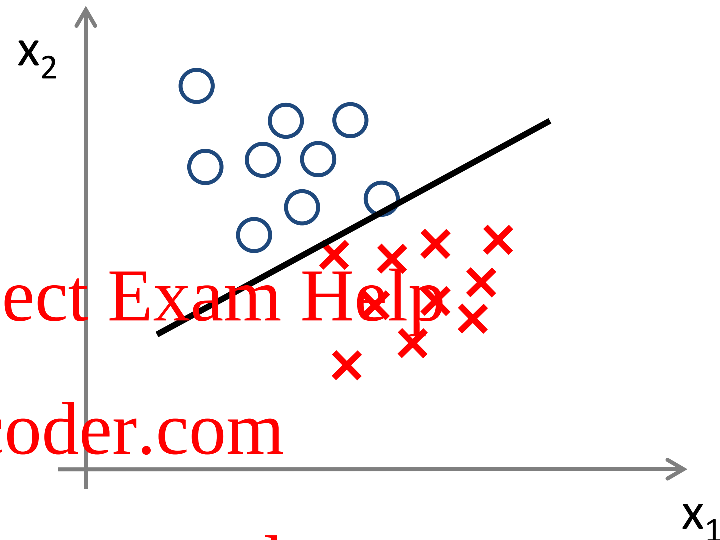
More than two classes, unequal covariances

- more general case of unequal covariances (here shown for four classes)
- QDA
- the decision hypersurface is no longer a hyperplane, i.e. it is nonlinear.



Generative vs Discriminative





- Generative: model the class-conditional distribution of features
- Pros: Can use it to generate new features
- Cons: more parameters, e.g. LDA has O(n^2)

- Discriminative: model the decision boundary directly, e.g. Logistic Regression
- Pros: fewer parameters, e.g. LogReg has O(n)
- Cons: Cannot generate new features

Do they produce the same classifier?

- Generative LDA approach will estimate μ1,μ2, and Σ to maximize joint likelihood p(x,y) and then compute the linear decision boundary, i.e., θ_j and θ_0 are functions of those parameters. In particular, θ_j and θ_0 are not completely independent.

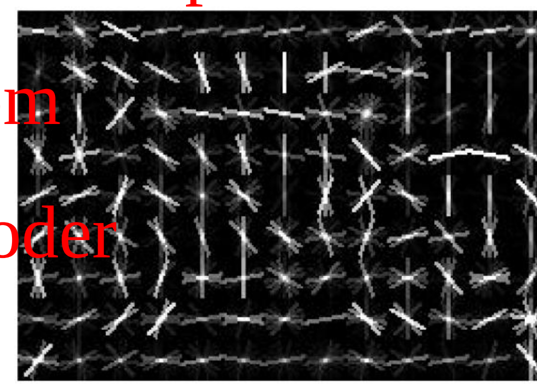
- Discriminative approach (logistic regression) will directly estimate θ_j and θ_0, without assuming any constraints between them, by maximizing conditional likelihood p(y|x)

- The two methods will give different decision boundaries, even both are linear.

LDA for image classification

- Discriminative Decorrelation for Clustering and Classification
  Hariharan, Malik and Ramanan, 2012
- Showed that LDA requires a lot less training than discriminative models for this task

(a) Image (left) and HOG (right)



Learned LDA model for class “bicycle”

http://home.bharathh.info/pubs/pdfs/BharathECCV2012.pdf

Next Class

Midterm! (no lecture)

Next Tuesday-

Probabilistic Models II: Bayesian Methods

priors over parameters; Bayesian linear regression

Reading: Bishop Ch 2.3