

Machine Learning in R

Lecture 2 -- Classification, linear algebra and loss functions

John Kornak – Epidemiology and Biostatistics

john.kornak@ucsf.edu

Last lecture

- What is Machine Learning?
- Supervised vs. Unsupervised learning
- Introduction to prediction techniques for supervised learning with continuous outcomes:
 - Local estimation and nearest neighbors
 - Loss functions and mean square error
 - The training-test paradigm
 - The curse of dimensionality
 - Linear regression modeling (including polynomials)
 - Some trade-offs

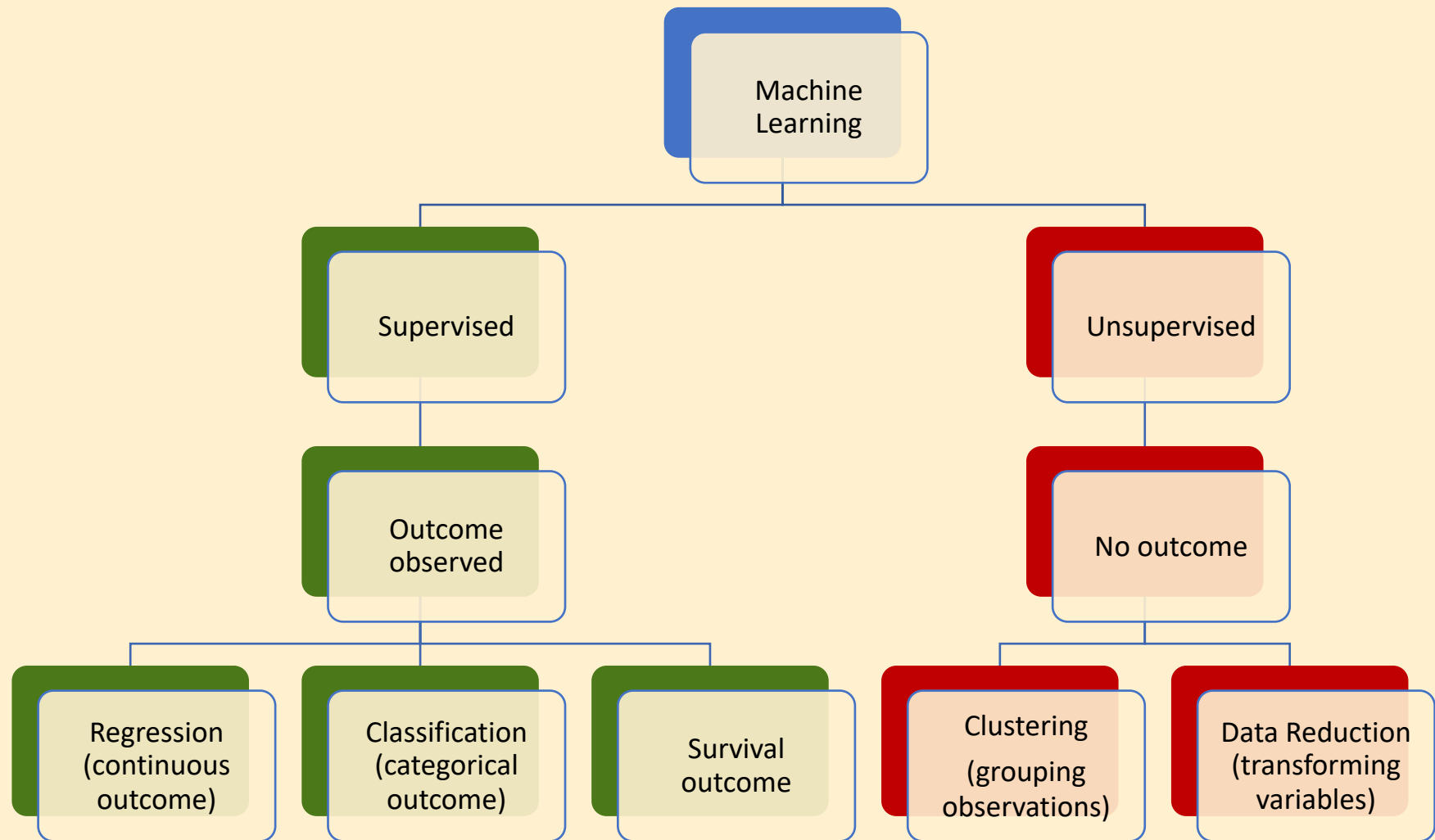
In this lecture:

- Supervised learning methods for classification
 - Logistic and multinomial regression
 - Linear discriminant analysis

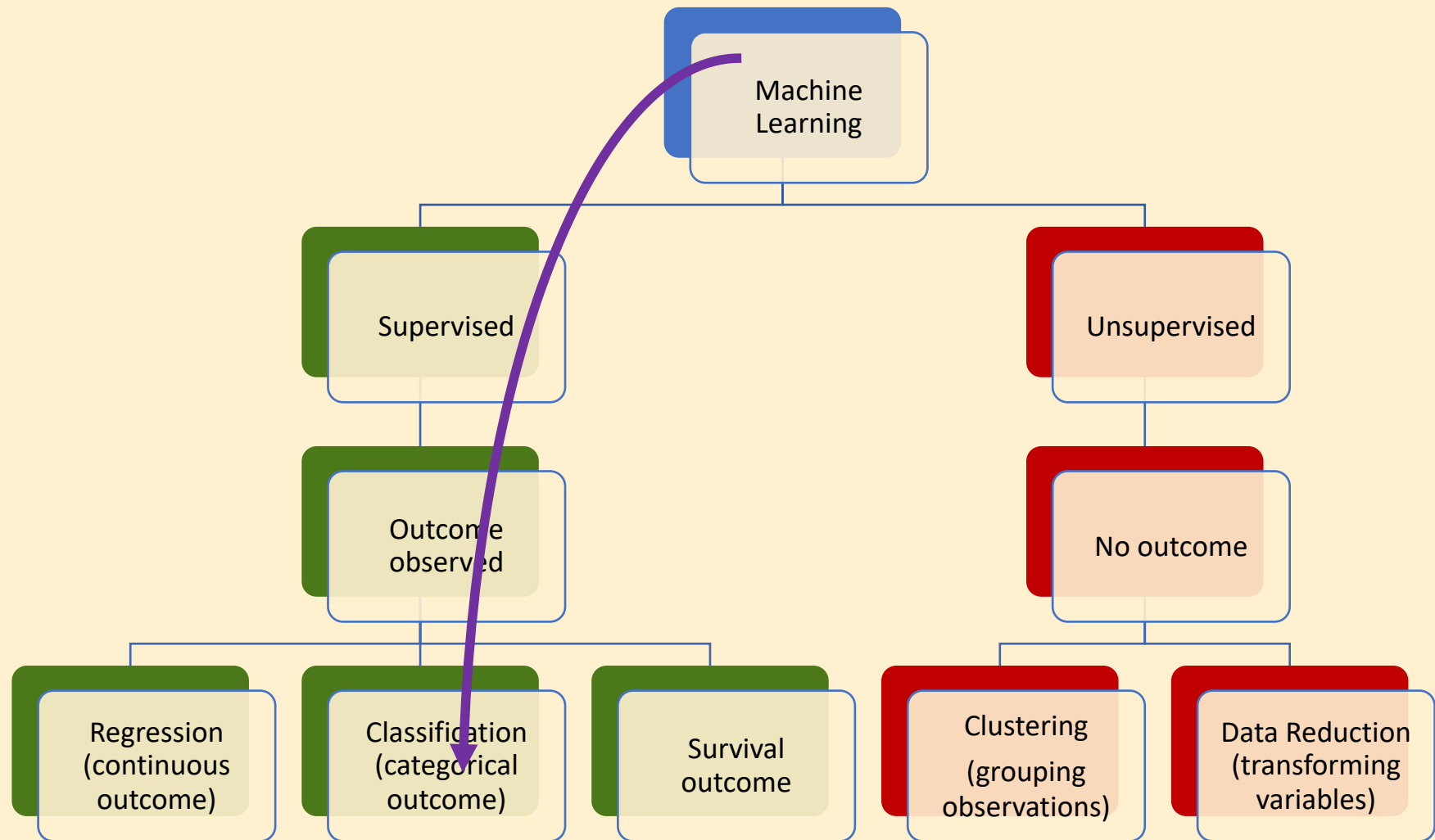
Classification

- Supervised learning where the outcome is a class variable
- Unordered class variable such as:
 $dx \in \{\text{cancer}, \text{benign}\}$
 $\text{blood_type} \in \{\text{typeA}, \text{typeB}, \text{typeAB}, \text{typeO}\}$
- Given a set of predictors X and a set of possible classification outcomes $\mathcal{C}$, the objective is to find a function/algorithm $C(X)$ that takes X as input and predicts its value for Y (which is a class value)
- Often interested in estimating the *probabilities* of each class – for example, if diagnosing cancer it is important to know just how certain that diagnosis is – we may act very differently if $\text{Pr}(\text{cancer}) = 0.52$ vs. $\text{Pr}(\text{cancer}) = 0.98$

Supervised vs. unsupervised



Supervised vs. unsupervised



Classification with logistic regression

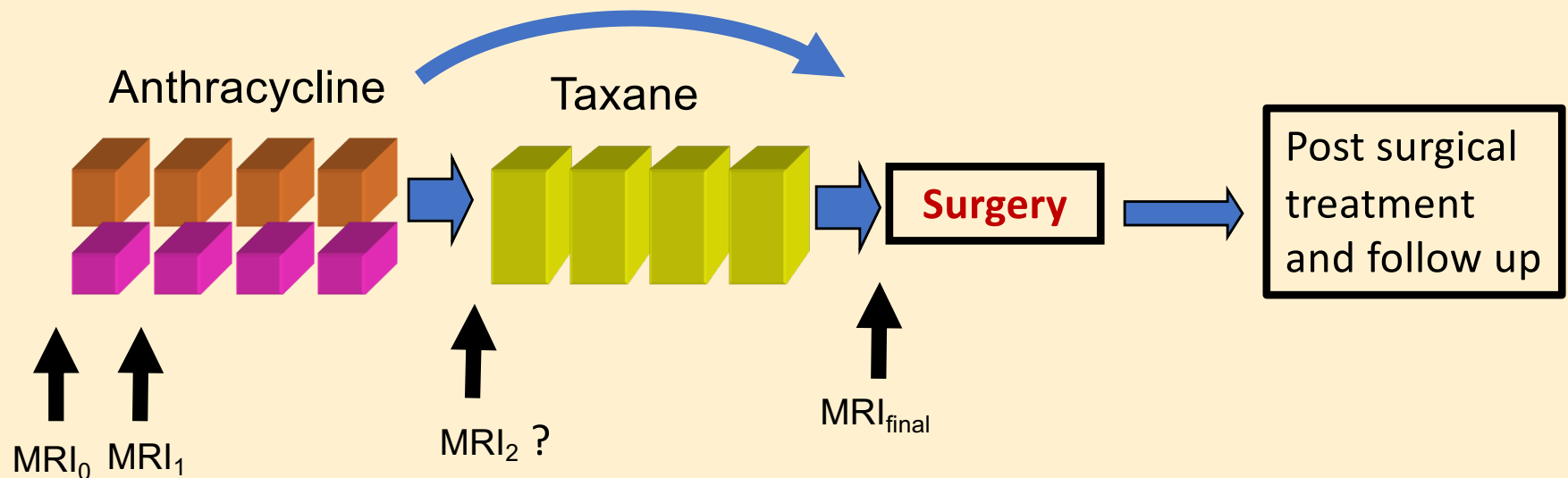
- Logistic regression can provide us with a classification approach
- Consider $p(X) = \Pr(Y = 1|X)$
- $p(X) = \frac{e^{\beta_0 + \beta_1 X_1}}{1 + e^{\beta_0 + \beta_1 X_1}}$, for a single predictor (X_1) model
- Note that $0 \leq p(X) \leq 1$
- Rearranging we get $\log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X_1$, the *log odds* or *logit* transformation of $p(X)$
- Straightforward to generalize to p predictors

Logistic regression example – ISPY1

- Multicenter **I**nvestigation of **S**erial Studies to **P**redict **Y**our **T**herapeutic **R**esponse with **I**maging and mo**L**ecular **A**nalysis (**I-SPY TRIAL**) breast cancer trial
- A study of imaging and tissue-based biomarkers for predicting *pathological complete response (pCR)* and *recurrence free survival (RFS)*
- Inclusion criteria: T3 tumors > 3cm diameter, receiving neoadjuvant chemotherapy with an anthracycline-cyclophosphamide regimen alone or followed by a taxane
- Study was open to enrollment from May 2002 to March 2006. 237 patients were enrolled, of which 230 met eligibility criteria.

Experimental design of ISPY1 imaging

- 3 or 4 MRI scans
 - Before treatment (T0)
 - After 1 cycle AC (T1)
 - Before surgery (Tfinal)



ISPY1 -- Dynamic contrast-enhanced (DCE) MRI

- DCE-MRI is standard for breast cancer evaluation in high-risk patients
- T1-weighted imaging performed before and after injection of gadolinium contrast agent
- Used for cancer detection (screening) and evaluation of response to treatment
- For evaluation of response to treatment, additional requirements include:
 - Sequential MRIs performed on same (or equivalent) scanner
 - All scan parameters kept constant
 - Gadolinium administration kept constant (dose, rate of injection, flush volume)

***contrast
injection***

← 80-100 second temporal resolution; 8 minute duration →

Pre-contrast



Post-contrast 1

Post-contrast 2

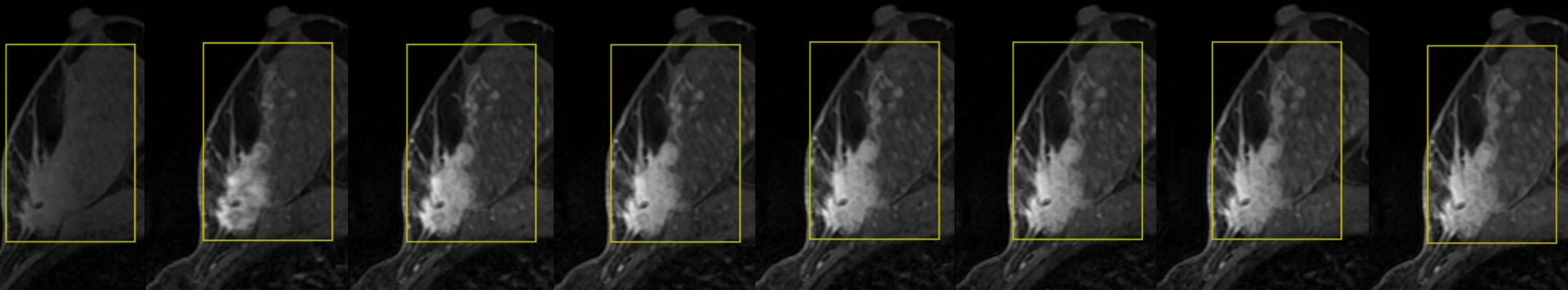
Post-contrast 3

Post-contrast 4

Post-contrast 5

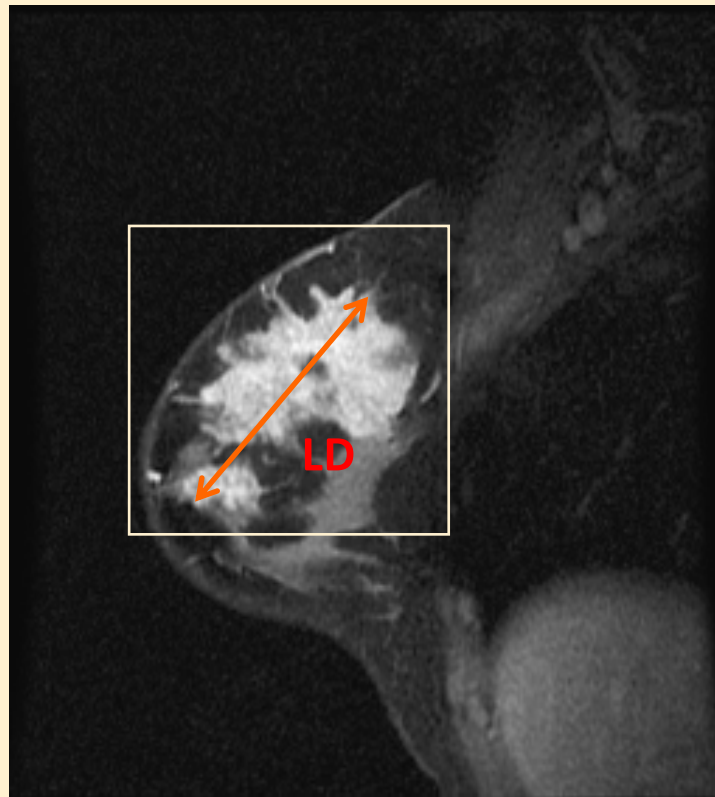
Post-contrast 6

Post-contrast 7



Longest diameter (LD) predictor

Measured by radiologist (in I-SPY 1 clinical trial, diameter measurements at treatment time-points were required to be along same axis as baseline)



Longest diameter (LD)

Can longest diameter @T0 predict pCR?

- Fit logistic regression with pCR as outcome (1=pCR, 0=not pCR) and longest diameter on MRI at T0 as predictor

```
> glm0 <- glm(pCR ~ MRI_LD_T0, family = "binomial", data=dat)  
>
```


Can longest diameter @T0 predict pCR?

- Fit logistic regression with pCR as outcome (1=pCR, 0=not pCR) and longest diameter on MRI at T0 as predictor

```
> glm0 <- glm(pCR ~ MRI_LD_T0, family = "binomial", data=dat)
>
> summary(glm0)
```

Call:
glm(formula = pCR ~ MRI_LD_T0, family = "binomial", data = dat)

Deviance Residuals:

Min	1Q	Median	3Q	Max
-0.8944	-0.8287	-0.7822	1.5164	1.8393

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-0.614675	0.385799	-1.593	0.111
MRI_LD_T0	-0.004745	0.005252	-0.904	0.366

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 236.52 on 198 degrees of freedom
Residual deviance: 235.69 on 197 degrees of freedom
(7 observations deleted due to missingness)
AIC: 239.69

Number of Fisher Scoring iterations: 4

```
>
```

Can longest diameter @T0 predict pCR?

- We can get confidence intervals too

```
> confint(glm0)
Waiting for profiling to be done...
                2.5 %      97.5 %
(Intercept) -1.37709620 0.141067045
MRI_LD_T0   -0.01536686 0.005324228
>
```

or

```
> library(jtools)
> summ(glm0, confint=TRUE, digits=3)
MODEL INFO:
Observations: 199 (7 missing obs. deleted)
Dependent Variable: pCR
Type: Generalized linear model
Family: binomial
Link function: logit

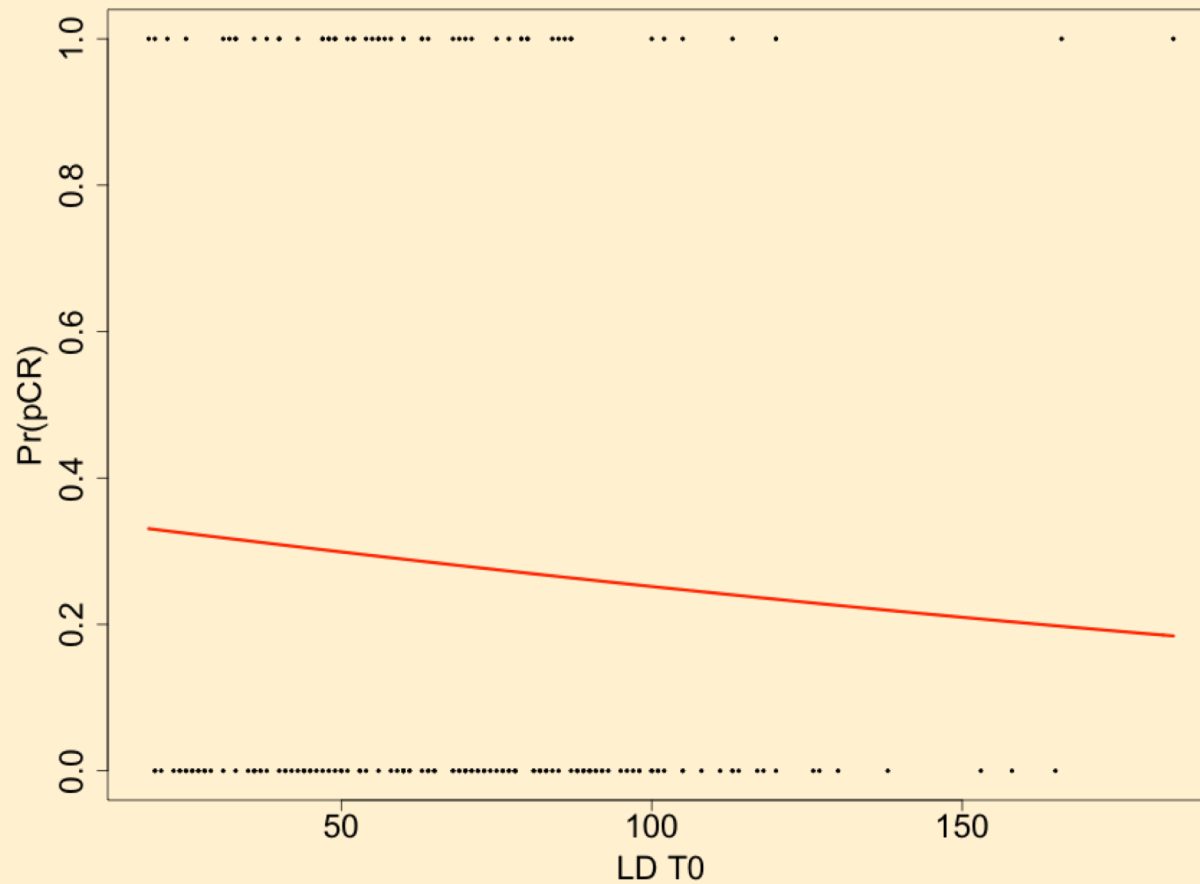
MODEL FIT:
 $\chi^2(1) = 0.836$ ,  $p = 0.361$ 
Pseudo- $R^2$  (Cragg-Uhler) = 0.006
Pseudo- $R^2$  (McFadden) = 0.004
AIC = 239.687, BIC = 246.273

Standard errors: MLE
```

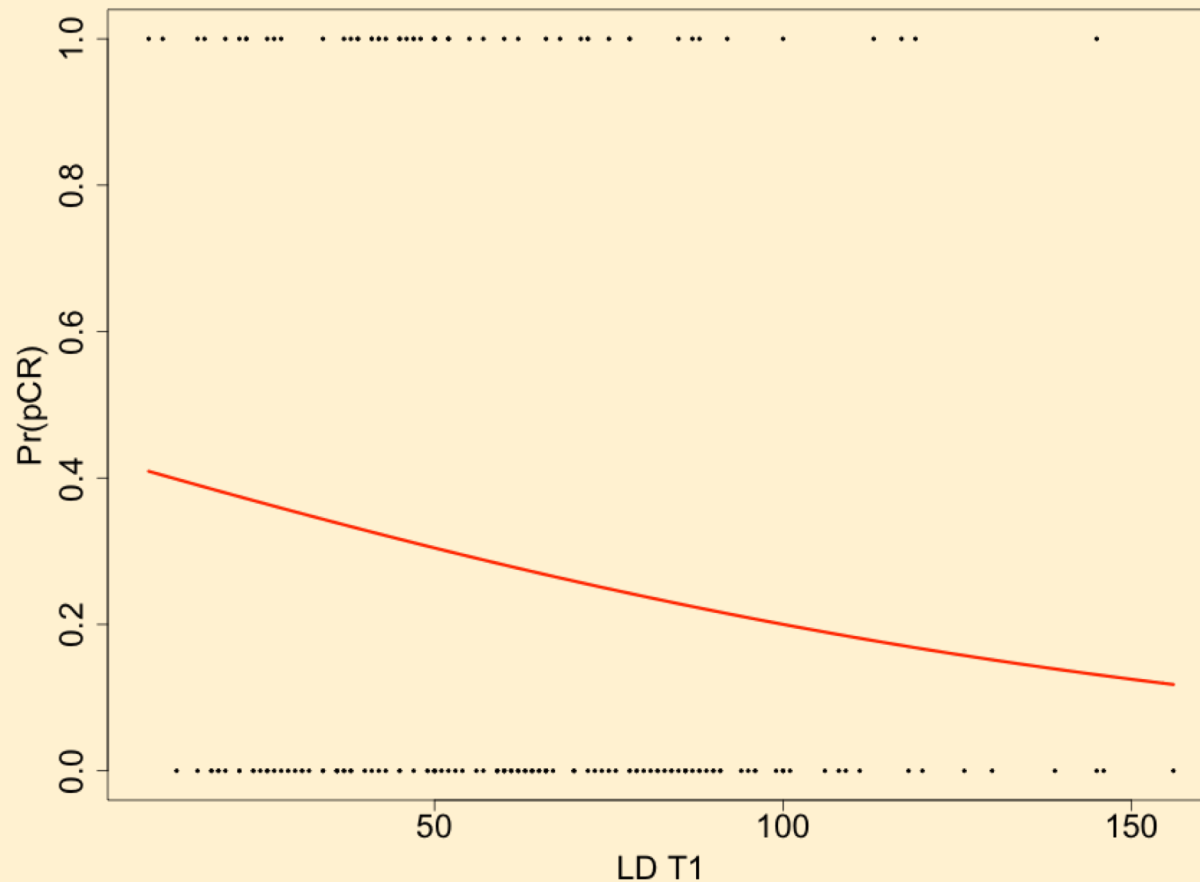
	Est.	2.5%	97.5%	z val.	p
(Intercept)	-0.615	-1.371	0.141	-1.593	0.111
MRI_LD_T0	-0.005	-0.015	0.006	-0.904	0.366

```
> |
```

Can longest diameter @T0 predict pCR?



What about longest diameter @T1 predicting pCR?



What about longest diameter @T1 predicting pCR?

```
> glm1 <- glm(pCR ~ MRI_LD_T1, family = "binomial", data=dat)
> summ(glm1, confint=TRUE, digits=3)
```

MODEL INFO:

Observations: 191 (15 missing obs. deleted)

Dependent Variable: pCR

Type: Generalized linear model

Family: binomial

Link function: logit

MODEL FIT:

$\chi^2(1) = 3.950$, $p = 0.047$

Pseudo-R² (Cragg-Uhler) = 0.029

Pseudo-R² (McFadden) = 0.017

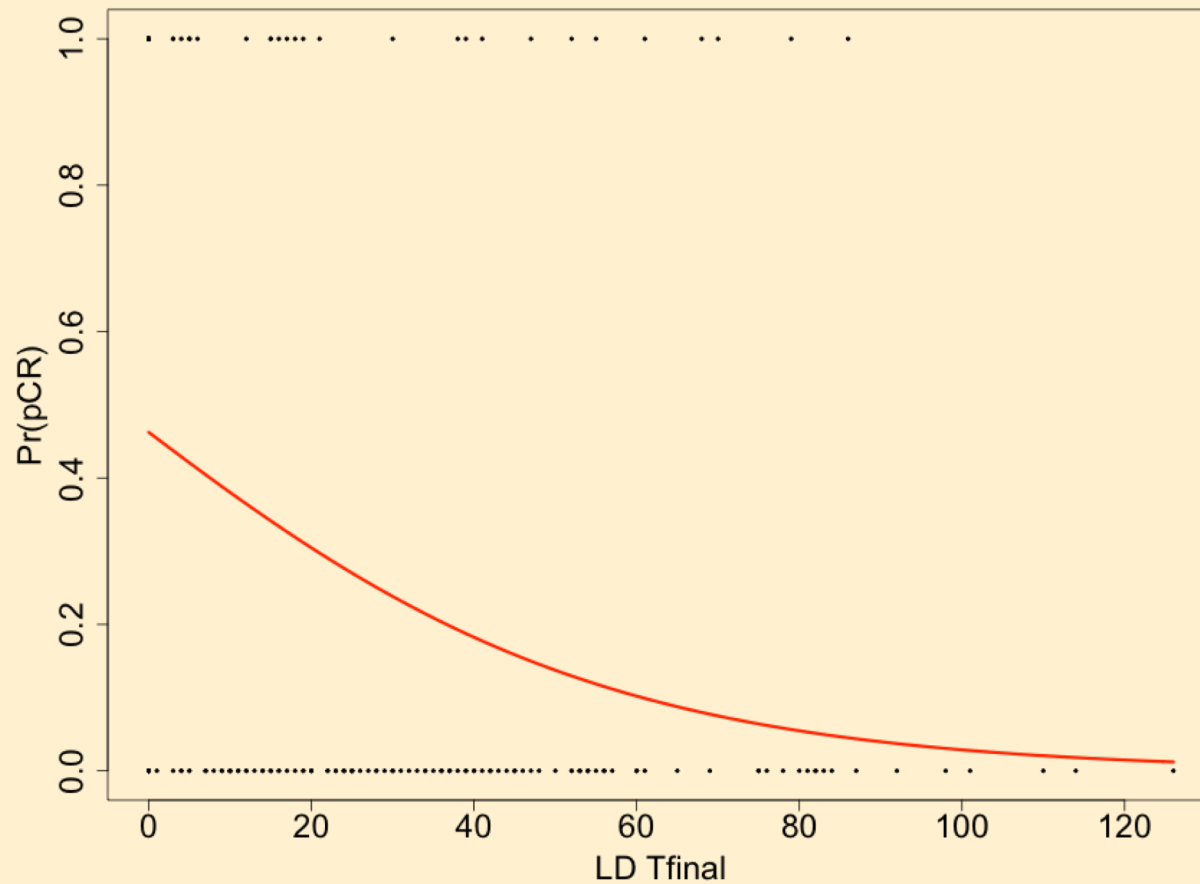
AIC = 227.533, *BIC* = 234.038

Standard errors: MLE

	Est.	2.5%	97.5%	z val.	p
(Intercept)	-0.267	-0.991	0.458	-0.721	0.471
MRI_LD_T1	-0.011	-0.023	0.000	-1.926	0.054

```
> |
```

Longest diameter @Tfinal predicting pCR?



Longest diameter @Tfinal predicting pCR?

```
> glmfinal <- glm(pCR ~ MRI_LD_Tfinal, family = "binomial", data=dat)
> summ(glmfinal, confint=TRUE, digits=3)
```

MODEL INFO:

Observations: 192 (14 missing obs. deleted)

Dependent Variable: pCR

Type: Generalized linear model

Family: binomial

Link function: logit

MODEL FIT:

$\chi^2(1) = 22.394$, $p = 0.000$

Pseudo-R² (Cragg-Uhler) = 0.159

Pseudo-R² (McFadden) = 0.099

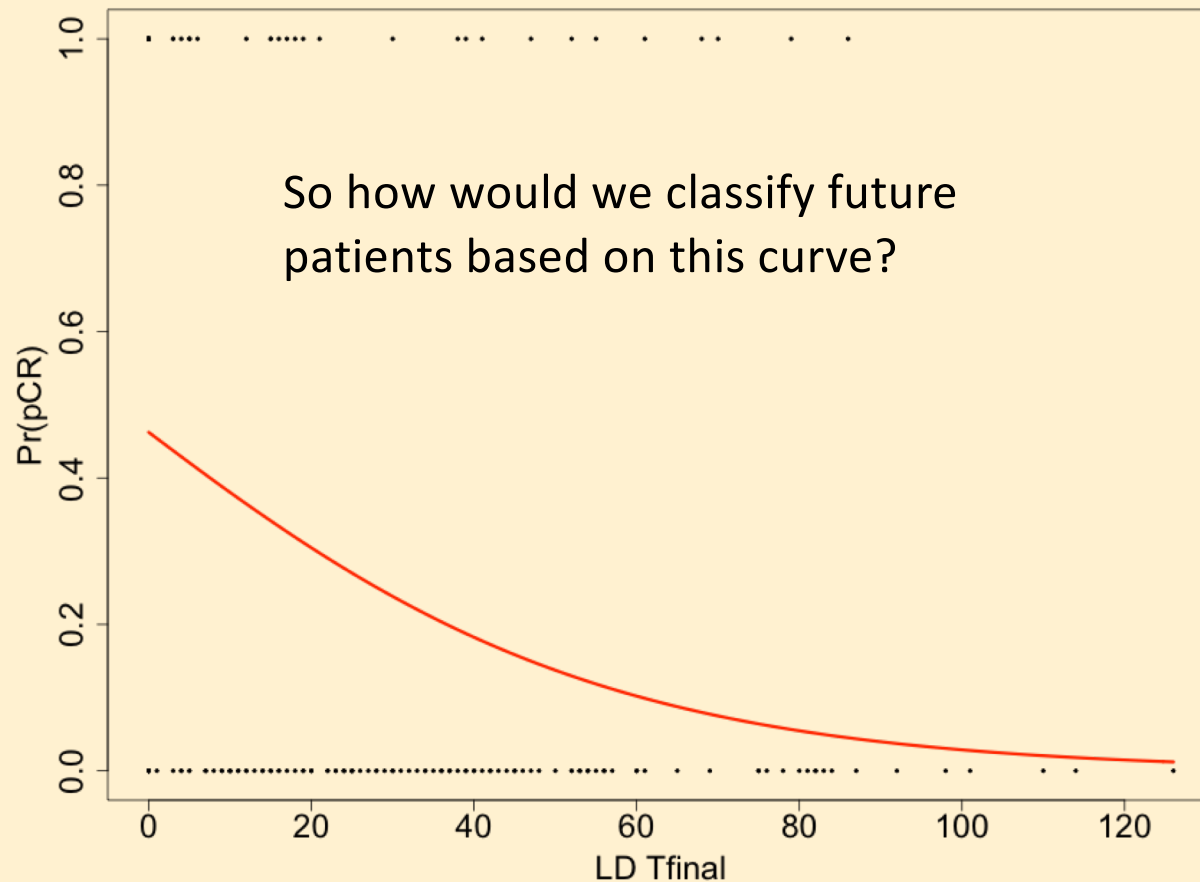
AIC = 207.850, BIC = 214.365

Standard errors: MLE

	Est.	2.5%	97.5%	z val.	p
(Intercept)	-0.151	-0.606	0.305	-0.648	0.517
MRI_LD_Tfinal	-0.034	-0.050	-0.017	-4.072	0.000

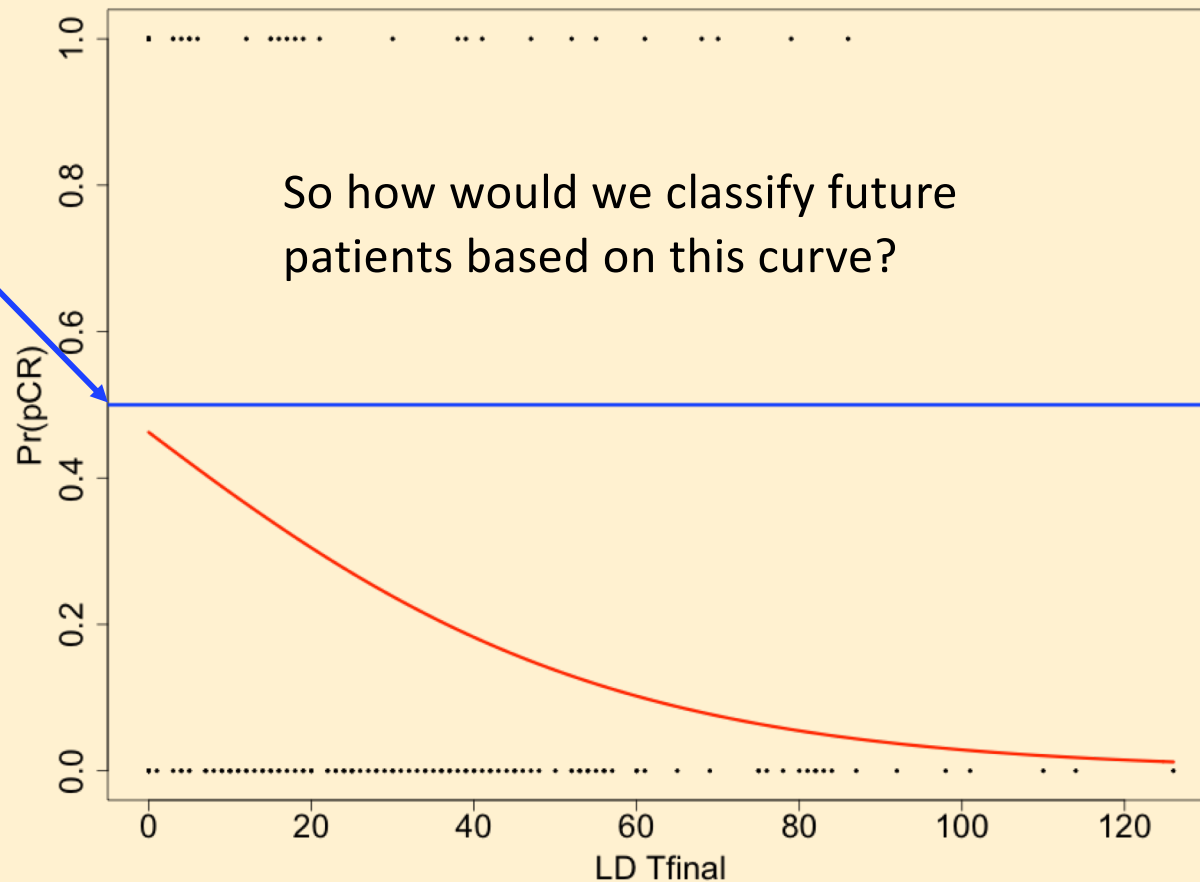
```
> |
```


Longest diameter @Tfinal predicting pCR?



Longest diameter @Tfinal predicting pCR?

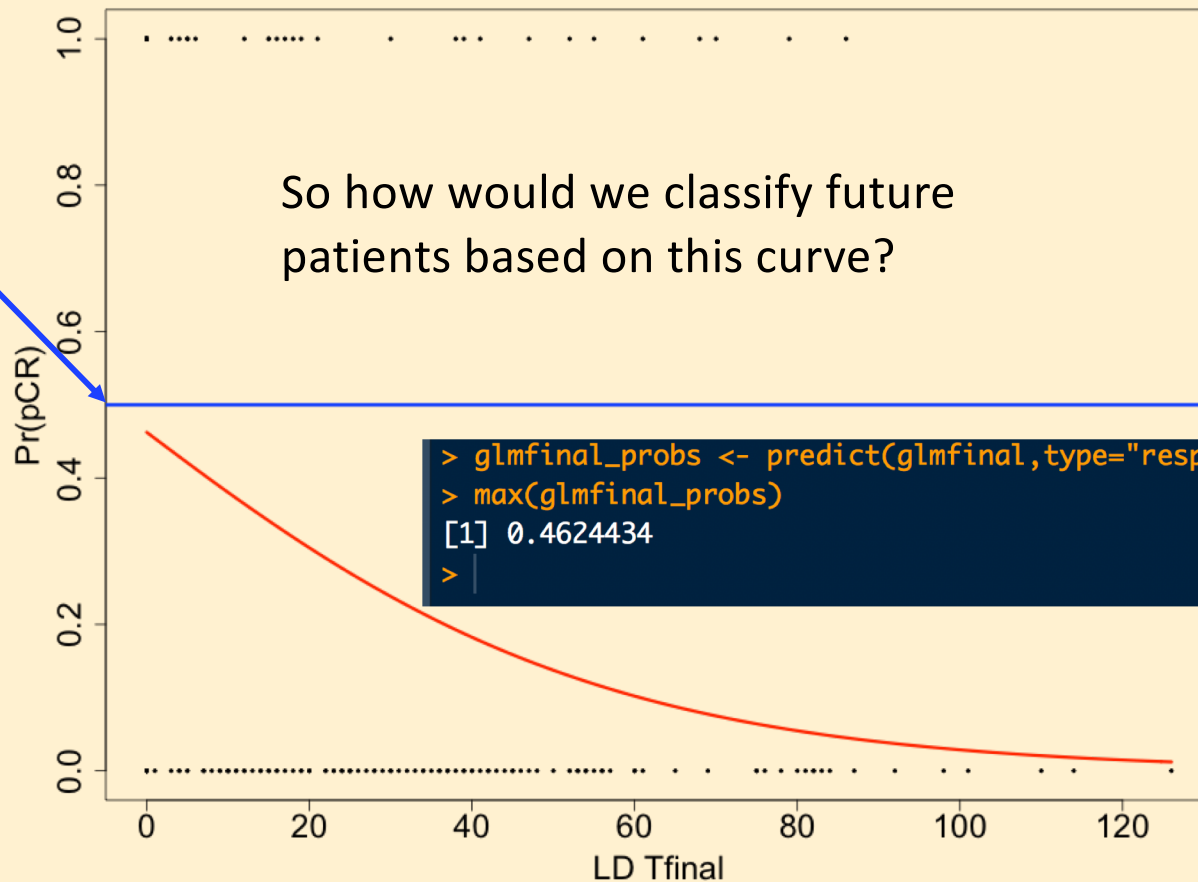
Threshold at
 $p(X) = 0.5$?



Longest diameter @Tfinal predicting pCR?

Threshold at
 $p(X) = 0.5$?

classifies
everyone as
not pCR

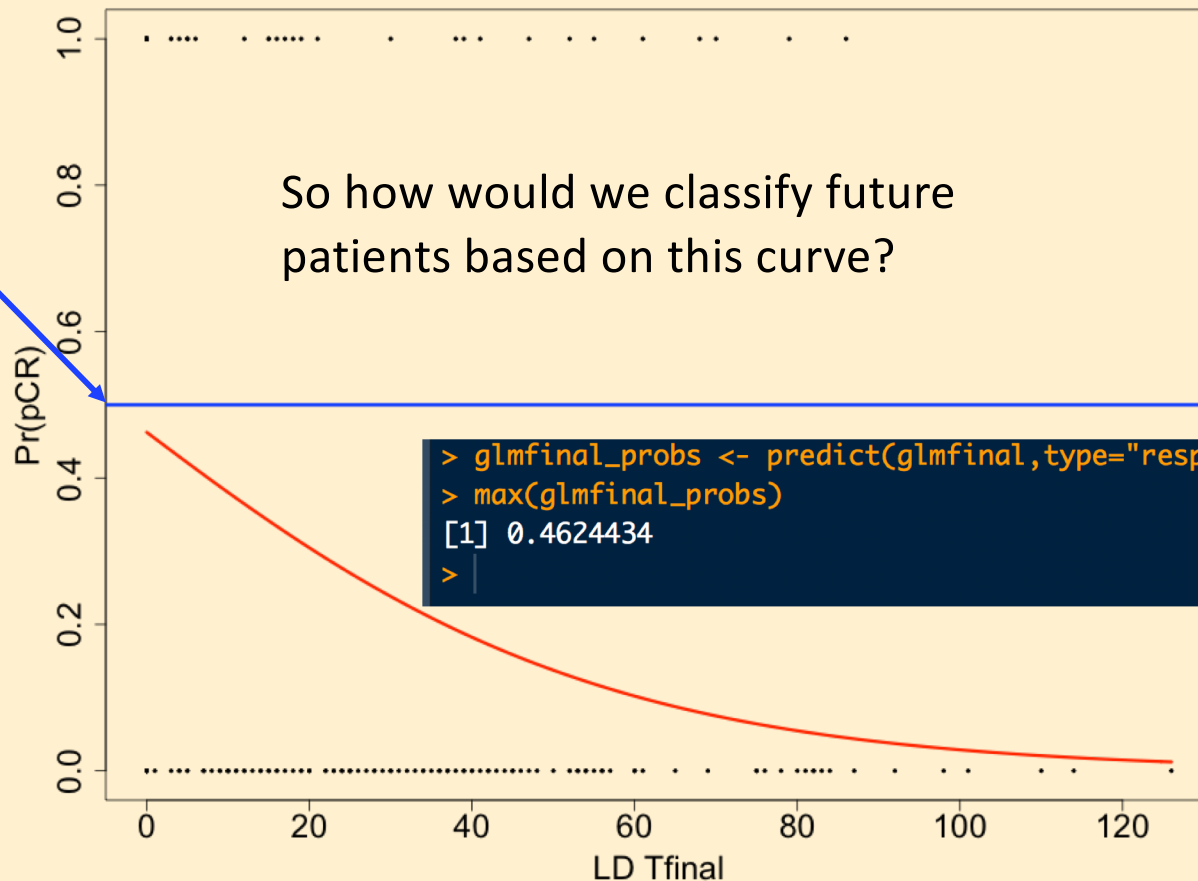


```
> glmfinal_probs <- predict(glmfinal,type="response")  
> max(glmfinal_probs)  
[1] 0.4624434  
> |
```

Maybe we need other or more predictors?

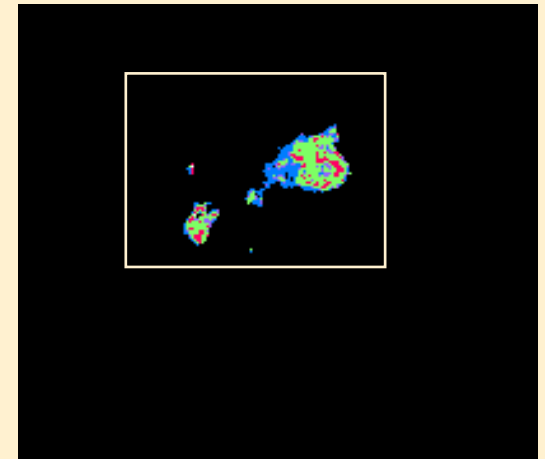
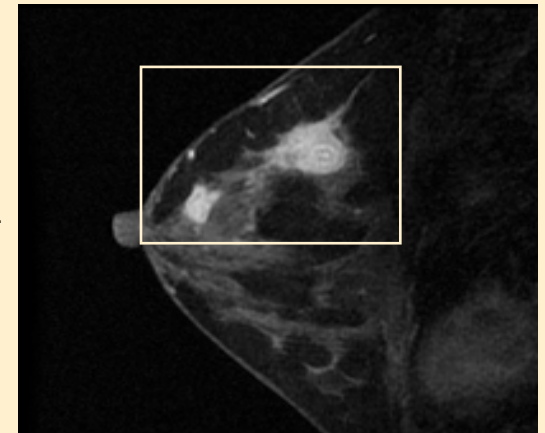
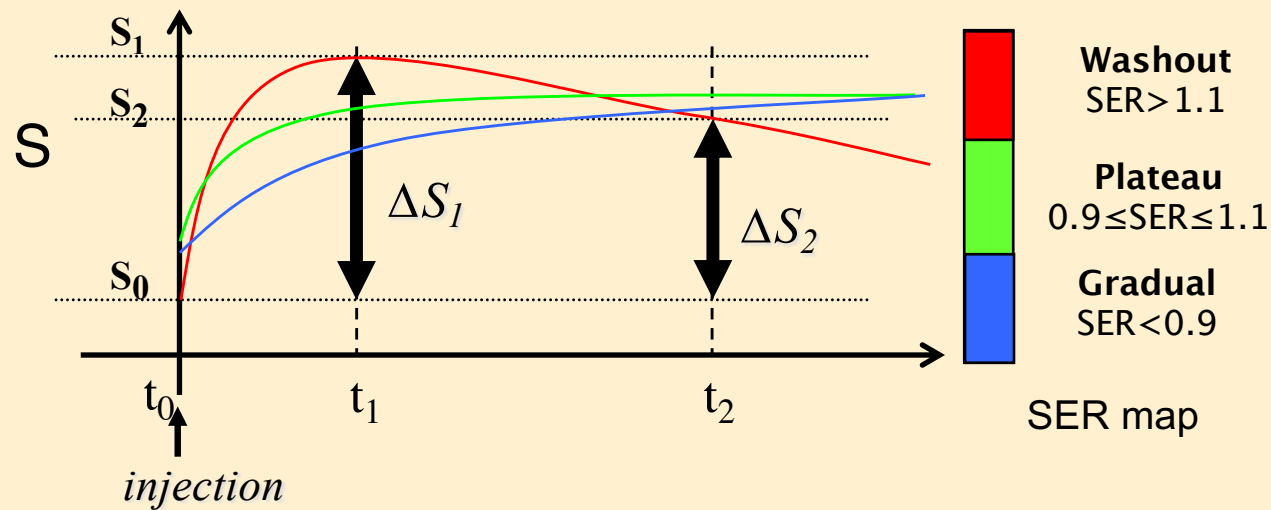
Threshold at
 $p(X) = 0.5$?

classifies
everyone as
not pCR



We need to make it over 0.5 in order to predict anyone as pCR = 1

Percent Enhancement (PE), Signal Enhancement Ratio (SER), and Functional Tumor Volume (FTV)



- Early percent enhancement (PE) = $\frac{\Delta S_1}{S_0} \times 100\%$
- Signal enhancement ratio (SER) = $\frac{\Delta S_1}{\Delta S_2}$

Higher SER (i.e. faster washout) indicative of cancer tissue

Functional tumor volume (FTV) is the volume of tissue for which both PE and SER are both above their respective pre-specified thresholds

Try LD Tfinal * FTV change T0 to T2

```
> glmMult <- glm(pCR ~ MRI_LD_Tfinal * ftvPePctChgT0_T2, family = "binomial", data=dat)
> summ(glmMult, confint=TRUE, digits=3)
MODEL INFO:
Observations: 142 (64 missing obs. deleted)
Dependent Variable: pCR
Type: Generalized linear model
Family: binomial
Link function: logit

MODEL FIT:
 $\chi^2(3) = 24.399$ ,  $p = 0.000$ 
Pseudo- $R^2$  (Cragg-Uhler) = 0.227
Pseudo- $R^2$  (McFadden) = 0.145
AIC = 152.451, BIC = 164.274

Standard errors: MLE
```

	Est.	2.5%	97.5%	z val.
(Intercept)	-5.275	-9.228	-1.321	-2.615
MRI_LD_Tfinal	0.027	-0.021	0.075	1.120
ftvPePctChgT0_T2	-0.054	-0.096	-0.013	-2.569
MRI_LD_Tfinal:ftvPePctChgT0_T2	0.001	0.000	0.001	2.209

```
-----
p
-----
(Intercept) 0.009
MRI_LD_Tfinal 0.263
ftvPePctChgT0_T2 0.010
MRI_LD_Tfinal:ftvPePctChgT0_T2 0.027
-----
> glmMult_probs <- predict(glmMult, type="response")
> max(glmMult_probs, na.rm=TRUE)
[1] 0.5393695
>
```

But we would really need to perform independent validation to see if we are really getting anything extra out of using this more complex model.

In fact, is 0.5 really the right threshold value for us here?

What if it is twice as bad to misclassify pCR as not pCR than it is to misclassify not pCR as pCR?

Let's think about this as a loss function

- We could define the loss function as follows:
 - Loss = 0 when we correctly classify either **pCR** or **non-pCR**
 - Loss = 1 when we misclassify **non-pCR** as **pCR**
 - Loss = 2 when we misclassify **pCR** as **non-pCR**
- We would then find the probability threshold that minimizes the expected loss over all observations
- This could then be evaluated with reserved data
- We will not pursue this idea further with logistic regression because there are classification methods that deal with this problem more naturally – including *discriminant analysis...*

Extending logistic regression for more than 2 classes

- We have discussed logistic regression with 2 classes (e.g. pCR vs. non-pCR) but we can readily extend to more than 2 classes (e.g. heart problems vs. lung problems vs liver problems).
- One common version has the following form:

$$\Pr(Y = k|X) = \frac{e^{\beta_{0k} + \beta_{1k}X_1 + \dots + \beta_{pk}X_p}}{\sum_{l=1}^K e^{\beta_{0l} + \beta_{1l}X_1 + \dots + \beta_{pl}X_p}}$$

- Often referred to as *multinomial regression* – e.g. you can use the `multinom` function in the `nnet` package or the `mlogit` package
- Note that Σ is the summation sign so that

$$\sum_{i=0}^p x_i = x_0 + x_1 + \dots + x_p \text{ or}$$

$$\sum_{n=1}^N f_n(x) = f_1(x) + f_2(x) + \dots + f_N(x),$$

$$\text{e.g. if } f_n(x) = x^n, \text{ then } \sum_{n=1}^N f_n(x) = x + x^2 + x^3 + \dots + x^N$$

Discriminant analysis

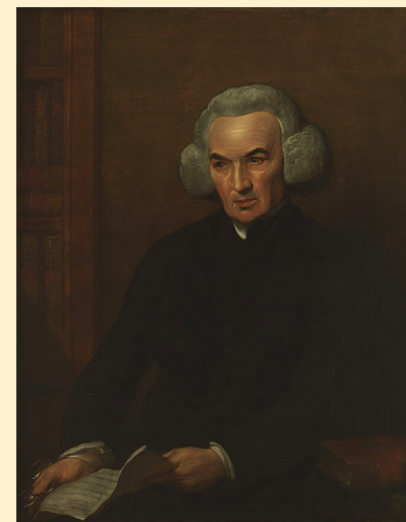
The idea behind discriminant analysis is to think about a model for the X s within each class (i.e. $\Pr(X = x|Y = k)$) and then flip things around to get $\Pr(Y = k|X = x)$ using Bayes' Theorem:

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



The Reverend
Thomas Bayes
1701-1761

Work was published
by Richard Price
1723-1791



Discriminant analysis

For discriminant analysis we write Bayes Theorem slightly differently as in the following form:

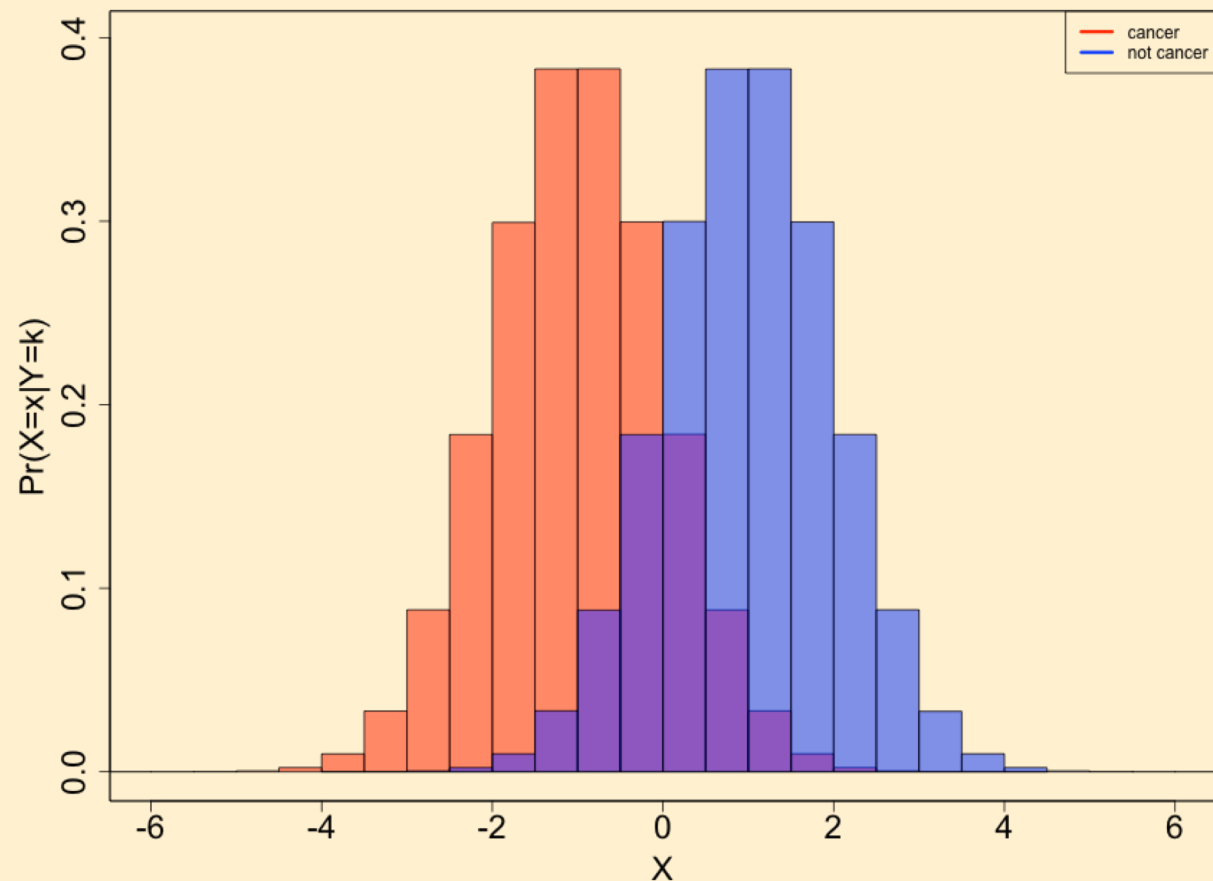
$$\Pr(Y = k|X = x) = p_k(x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^K \pi_l f_l(x)}$$

where,

- 1) $f_x(x) = \Pr(X = x|Y = k)$ is the *probability density* for X in class k (assumed normal)
- 2) $\pi_k = \Pr(Y = k)$ is the *marginal or prior probability* for class k

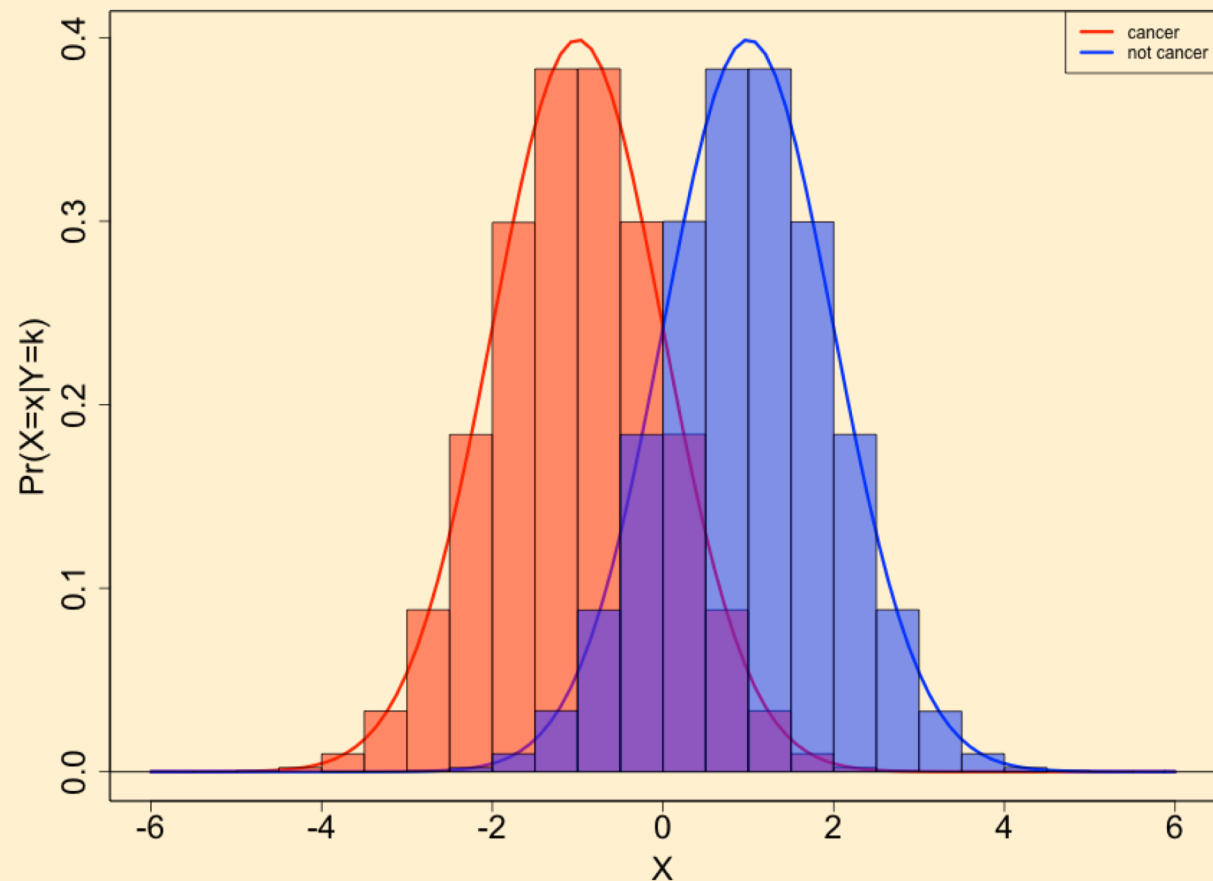
Equal prior probabilities

- Simulated example
- Cancer patients predictor X has mean = -1, sd = 1
- Non-cancer patients predictor X has mean = 1, sd = 1
- Simulated 10^7 samples from each distribution and plotted histogram



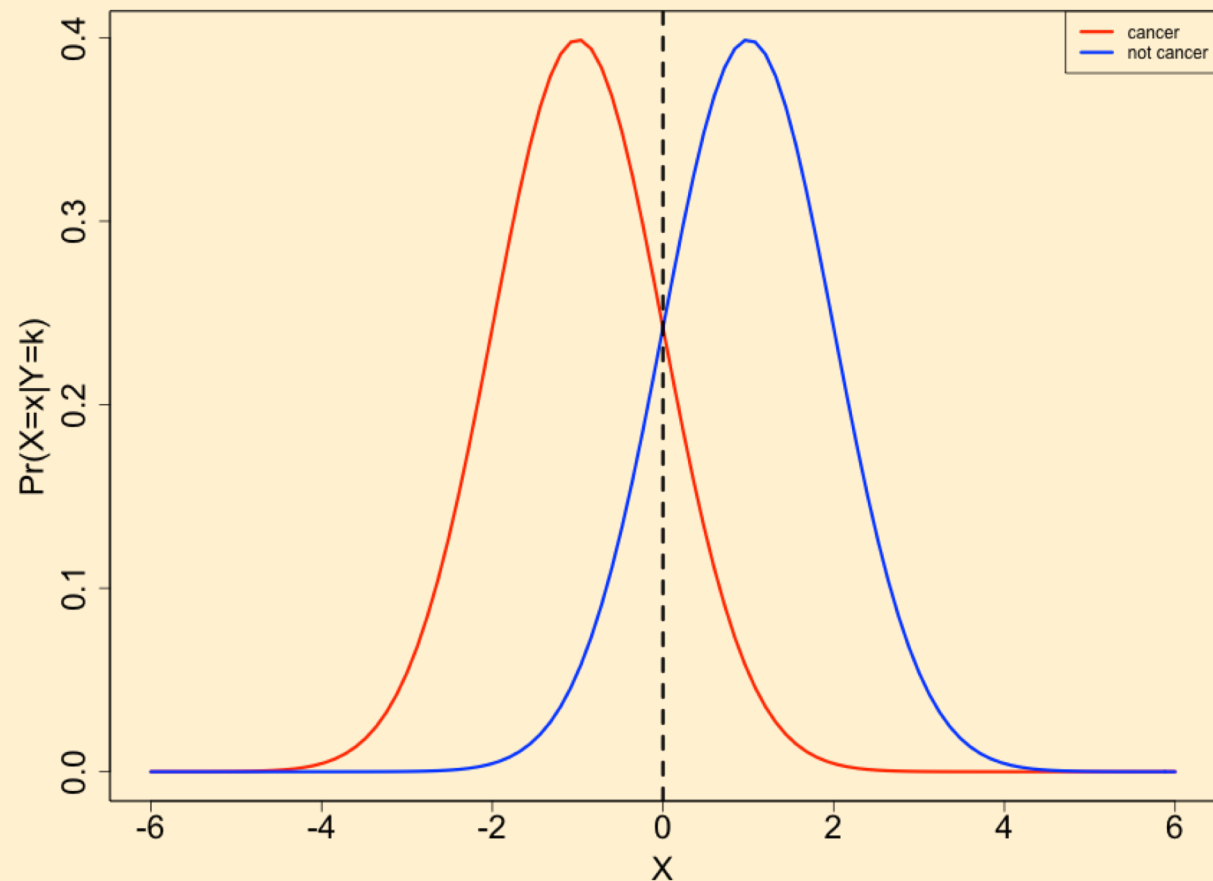
Equal prior probabilities

- Simulated example
- Cancer patients predictor X has mean = -1, sd = 1
- Non-cancer patients predictor X has mean = 1, sd = 1
- Simulated 10^7 samples from each distribution and plotted histogram
- Adding true probability density curves



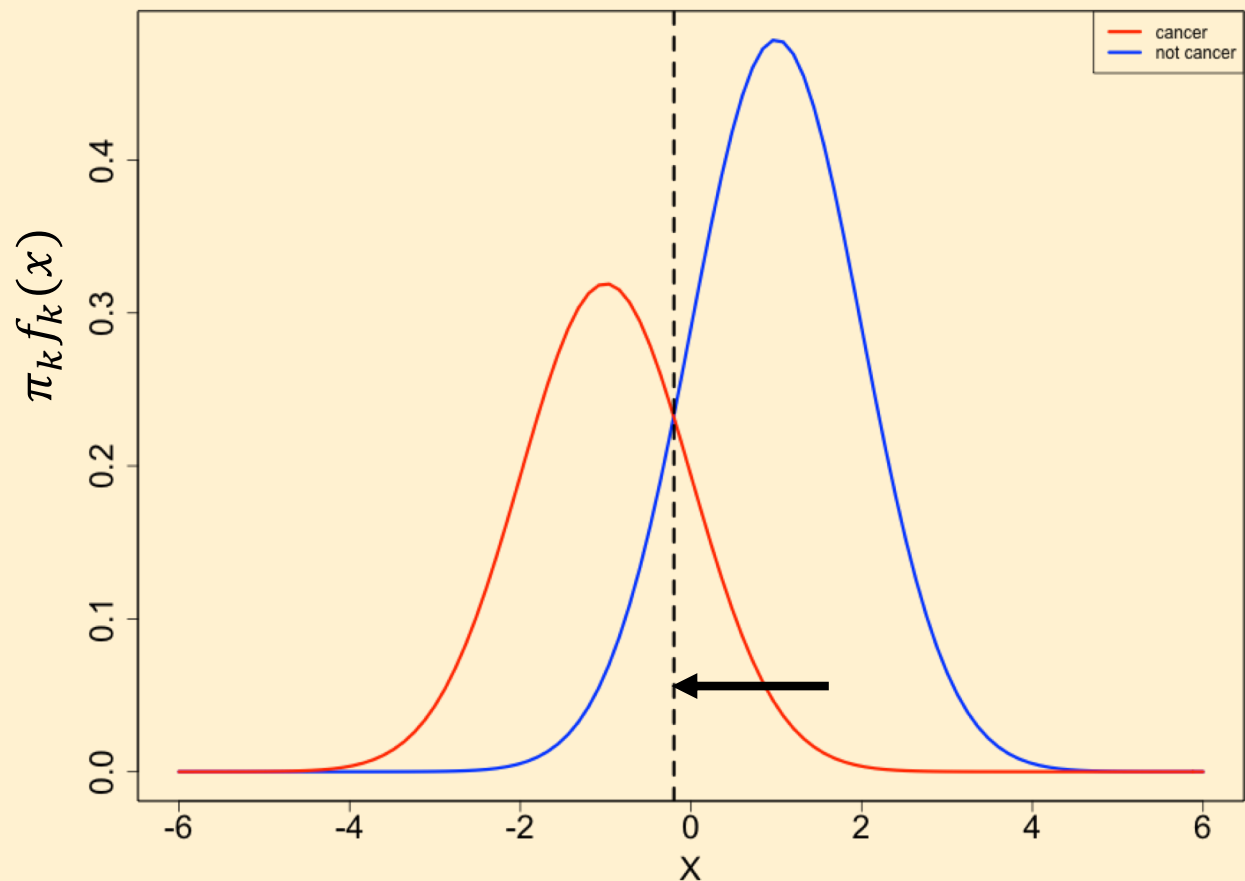
Equal prior probabilities

- For a new patient with observed predictor value X , choose class with highest corresponding density
- Decision boundary is at $X = 0$ when prior probabilities are equal



Unequal prior probabilities

- Now need to consider the prior probabilities are different and compare $\pi_k f_k(x)$ for the classes
- Decision boundary is shifted



Why discriminant analysis vs. logistic regression?

- When the classes are well-separated, the parameter estimates for the logistic regression model are surprisingly unstable. Linear discriminant analysis does not suffer from this problem
- If n is small and the distribution of the predictors X is *approximately* normal in each of the classes, the linear discriminant model is again more stable than the logistic regression model
- Linear discriminant analysis is easier to interpret when we have more than two response classes

Linear discriminant analysis (LDA) with $p = 1$

The normal/Gaussian probability density function has the form:

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma_k}\right)^2}$$

Here μ_k is and σ_k is the standard deviation for class k .
For LDA assume $\sigma_k = \sigma$ for all classes k .

Linear discriminant analysis (LDA) with $p = 1$

We can plug this normal probability density into the discriminant analysis Bayes formula

$$p_k(x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^K \pi_l f_l(x)}$$

to get

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma}\right)^2}}{\sum_{l=1}^K \pi_l \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma}\right)^2}}$$

Fortunately, this can be simplified with many cancellations

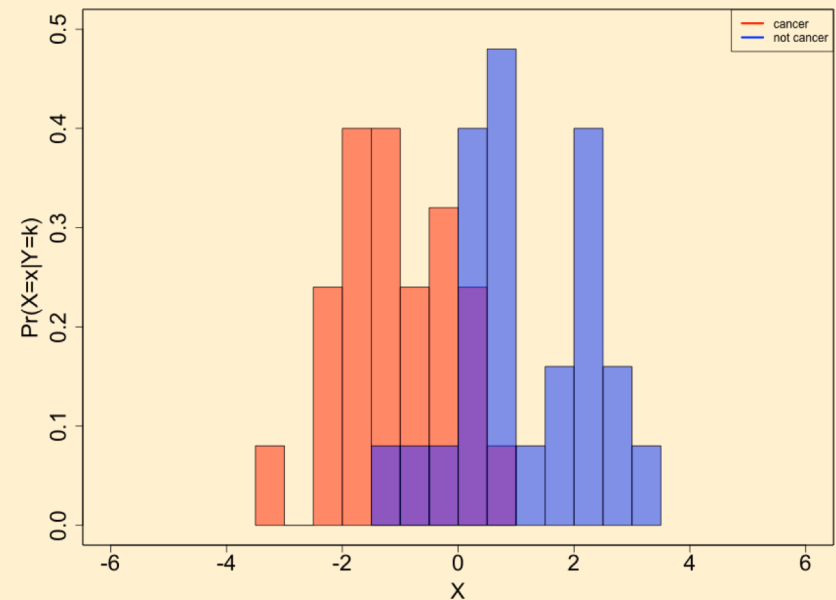
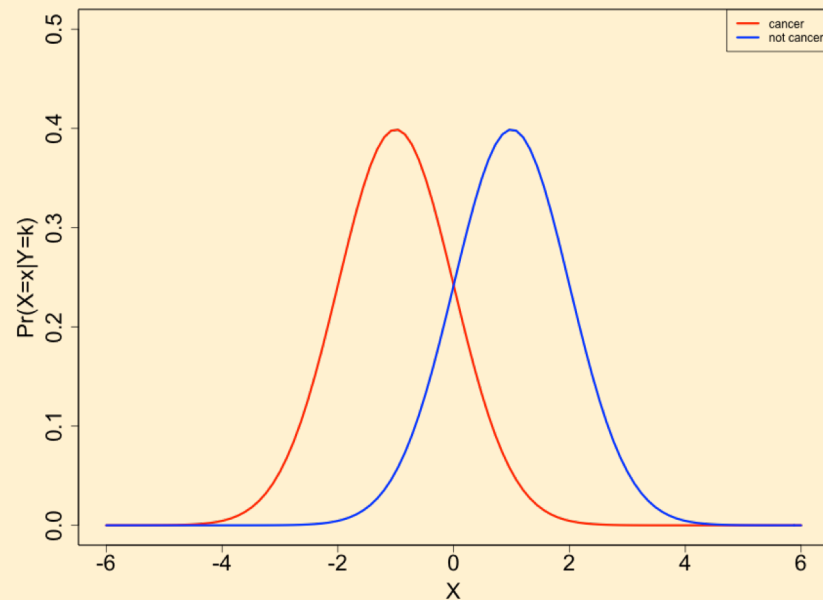
The discriminant function

- To classify at a value $X = x$, we determine which k gives the largest $p_k(x)$
- Taking logs of the equation for $p_k(x)$ on the previous slide, and discarding terms that do not depend on k , we see that this is equivalent to assigning x to the class with the largest *discriminant function score*:

$$\delta_k(x) = x \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

noting that $\delta_k(x)$ is a **linear** function of x

We don't know μ_k 's or σ



With a sample like on the right (25 in each group) we need to get an estimate of the parameters which of course will be estimated with more error than if we have 10^7 in each group like before

Estimating μ_k 's, σ , and π_k 's

$\hat{\pi}_k = \frac{n_k}{n}$, the proportion of observations in class k

$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i$, the mean of the x 's in class k

$\hat{\sigma}^2 = \frac{1}{n-K} \sum_{k=1}^K \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2$, pooled variance

$= \sum_{k=1}^K \frac{n_k-1}{n-K} \hat{\sigma}_k^2$, a weighted average of the individual class variances

where $\hat{\sigma}_k^2 = \frac{1}{n_k-1} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2$ is the usual formula for the estimated variance of the k^{th} class

Running LDA in R

Training data

```
> head(datLDAtrain)
  cancer      x
1 cancer -1.8408555
2 cancer  0.3843593
3 cancer -2.2554919
4 cancer -0.9298572
5 cancer  0.7114409
6 cancer -1.6029080
>
> tail(datLDAtrain)
  cancer      x
45 benign 0.5434311
46 benign 1.5622234
47 benign 0.1129915
48 benign 0.5397554
49 benign 0.2756715
50 benign 0.9307888
>
```

Running in R

```
> library(MASS)
> ldafit <- lda(cancer ~ x, data=datLDAtrain)
> ldafit
Call:
lda(cancer ~ x, data = datLDAtrain)

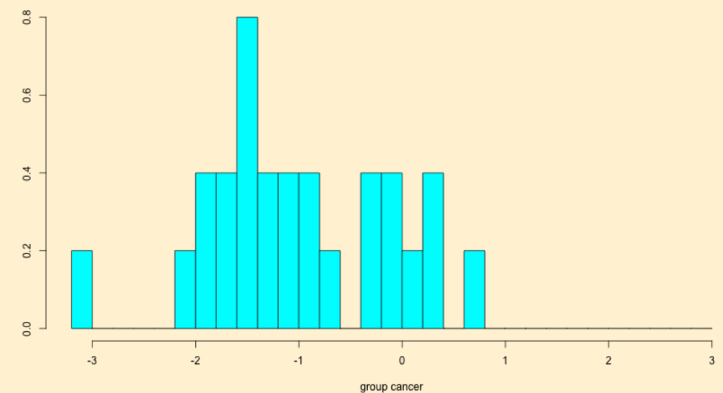
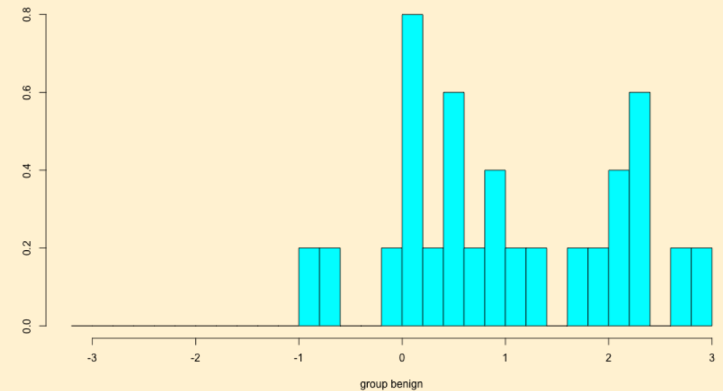
Prior probabilities of groups:
benign cancer
      0.5    0.5

Group means:
              x
benign  1.160870
cancer -1.031002

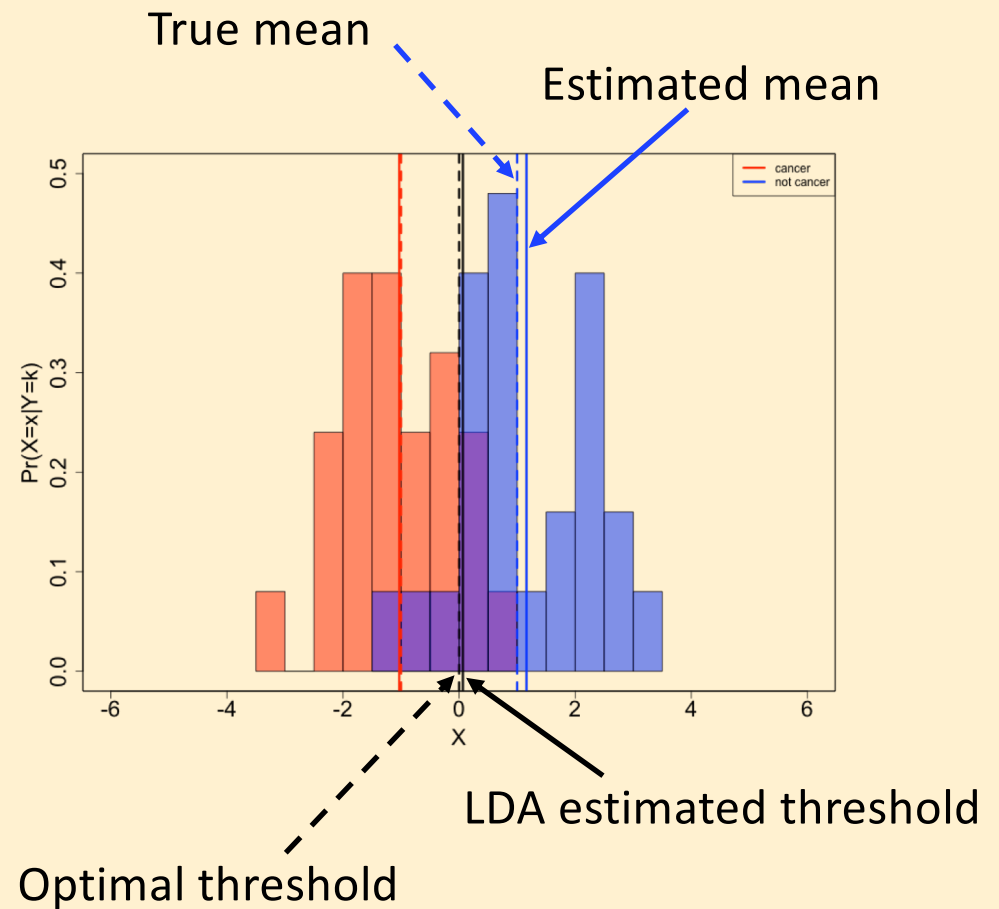
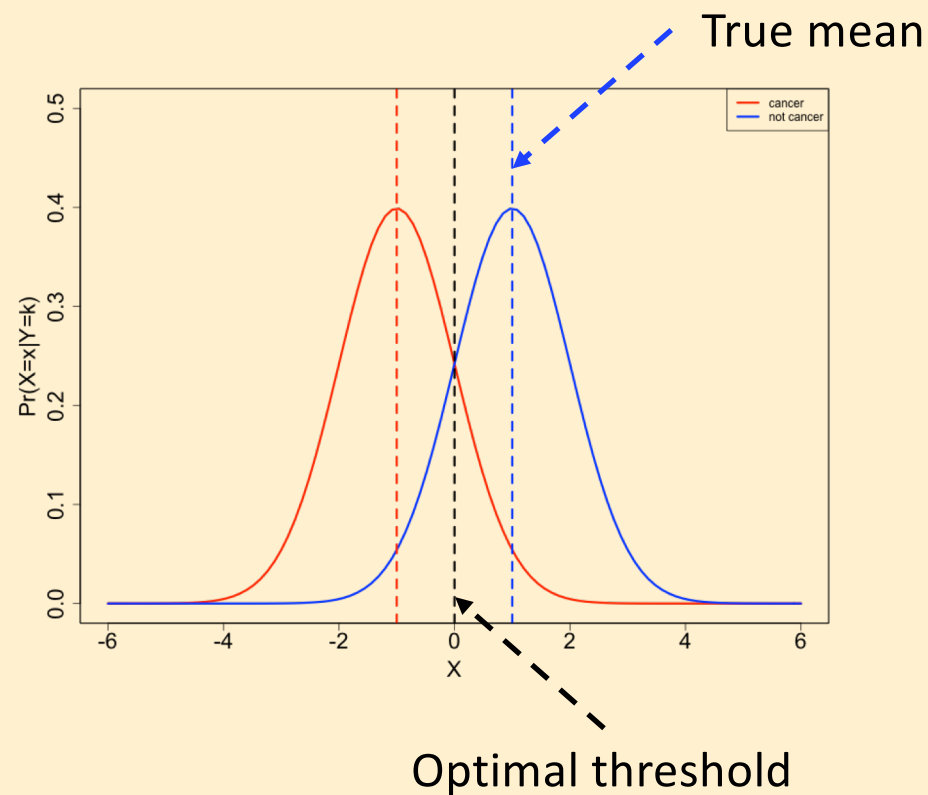
Coefficients of linear discriminants:
      LD1
x 0.9299661
>
```

Plotting in R

```
> plot(ldafit)
>
```



We don't know μ_k 's or σ



Assessing the prediction quality (training)

```
> ldapred <- predict(ldafit)
> names(ldapred)
[1] "class"      "posterior" "x"
> head(ldapred$class)
[1] cancer benign cancer cancer benign cancer
Levels: benign cancer
> head(ldapred$posterior)
      benign      cancer
1 0.02627175 0.9737282
2 0.64691493 0.3530851
3 0.01214489 0.9878551
4 0.13173109 0.8682689
5 0.77303446 0.2269655
6 0.04063734 0.9593627
> head(ldapred$x)
      LD1
1 -1.7723199
2  0.2970544
3 -2.1579177
4 -0.9251224
5  0.6012291
6 -1.5510368
> ldaclass <- ldapred$class
> table(ldaclass, datLDAttrain$cancer)

ldaclass benign cancer
benign    22      4
cancer     3     21
> mean(ldaclass==datLDAttrain$cancer)
[1] 0.86
```

```
> sum(ldapred$posterior[,1]>=.5)
[1] 26
> sum(ldapred$posterior[,1]<.5)
[1] 24
> ldapred$posterior[1:20,1]
      1      2      3      4      5
0.02627175 0.64691493 0.01214489 0.13173109 0.77303446
      6      7      8      9     10
0.04063734 0.05147819 0.03830490 0.07172969 0.14717868
      11     12     13     14     15
0.57649673 0.02823463 0.01684509 0.08969813 0.01711825
      16     17     18     19     20
0.09261077 0.04105283 0.00211063 0.17333030 0.07513651
> ldaclass[1:20]
[1] cancer benign cancer cancer benign cancer cancer
[8] cancer cancer cancer benign cancer cancer cancer
[15] cancer cancer cancer cancer cancer cancer
Levels: benign cancer
> sum(ldapred$posterior[,1]>.9)
[1] 11
```

But this is assessing the fit on the training data. What about looking at independent test data?

Assessing the prediction quality (test)

```
> ldapred <- predict(ldafit)
> names(ldapred)
[1] "class"      "posterior" "x"
> head(ldapred$class)
[1] cancer benign cancer cancer benign cancer
Levels: benign cancer
> head(ldapred$posterior)
      benign      cancer
1 0.02627175 0.9737282
2 0.64691493 0.3530851
3 0.01214489 0.9878551
4 0.13173109 0.8682689
5 0.77303446 0.2269655
6 0.04063734 0.9593627
> head(ldapred$x)
      LD1
1 -1.7723199
2  0.2970544
3 -2.1579177
4 -0.9251224
5  0.6012291
6 -1.5510368
> ldaclass <- ldapred$class
> table(ldaclass, datLDAttrain$cancer)

ldaclass benign cancer
benign    22      4
cancer     3     21
> mean(ldaclass==datLDAttrain$cancer)
[1] 0.86
```

```
> sum(ldapredTest$posterior[,1]>=.5)
[1] 26
> sum(ldapredTest$posterior[,1]<.5)
[1] 24
> ldapredTest$posterior[1:20,1]
      1      2      3      4      5
0.18128970 0.03868277 0.40804326 0.77823152 0.12208289
      6      7      8      9     10
0.21064043 0.01098210 0.35010863 0.12634818 0.01787955
     11     12     13     14     15
0.77844592 0.01402389 0.31421734 0.06195664 0.04088588
     16     17     18     19     20
0.12839867 0.77178034 0.01641177 0.07128815 0.89712643
> ldaclassTest[1:20]
[1] cancer cancer cancer cancer benign cancer cancer cancer cancer
[9] cancer cancer benign cancer cancer cancer cancer cancer
[17] benign cancer cancer benign
Levels: benign cancer
> sum(ldapredTest$posterior[,1]>.9)
[1] 8
> |
```

What if we want to change the prior probabilities

What if our data was from a matched case-control study (1:1) but we know that the prior probabilities at clinic will be in the proportion 80% non-cancer to 20% cancer. We may want to adjust our decision process to optimize with respect to these proportions.

```
> ldafit2 <- lda(cancer ~ x, prior=c(0.8, 0.2), data=datLDAtrain)
> ldafit2
Call:
lda(cancer ~ x, data = datLDAtrain, prior = c(0.8, 0.2))

Prior probabilities of groups:
benign cancer
  0.8    0.2

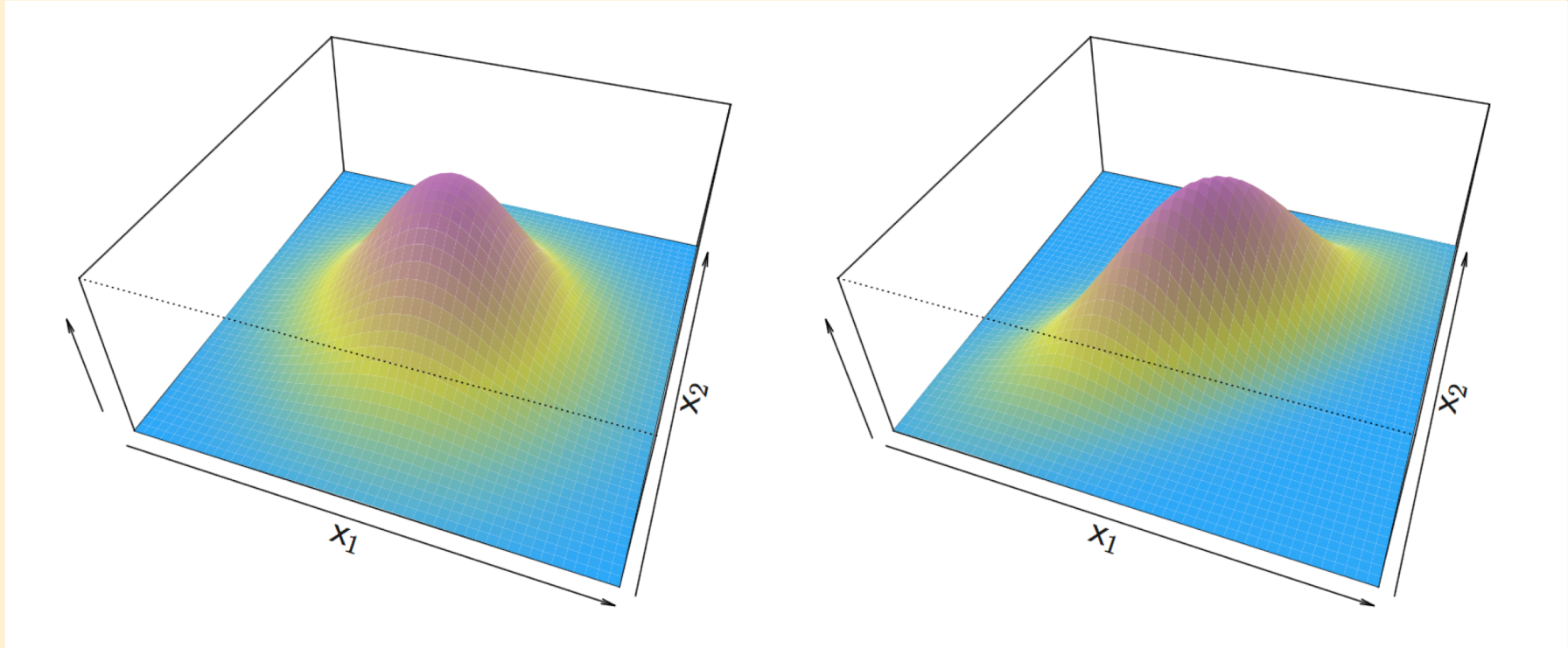
Group means:
             x
benign  1.160870
cancer -1.031002

Coefficients of linear discriminants:
      LD1
x  0.9299661
>
```

```
> ldapred2 <- predict(ldafit2)
> names(ldapred2)
[1] "class"    "posterior" "x"
> head(ldapred2$class)
[1] cancer benign cancer cancer benign cancer
Levels: benign cancer
> head(ldapred2$posterior)
      benign      cancer
1 0.09740964 0.90259036
2 0.87993345 0.12006655
3 0.04687180 0.95312820
4 0.37767123 0.62232877
5 0.93161842 0.06838158
6 0.14488601 0.85511399
> head(ldapred2$x)
      LD1
1 -2.38382993
2 -0.31445563
3 -2.76942770
4 -1.53663247
5 -0.01028091
6 -2.16254683
> ldaclass2 <- ldapred2$class
> table(ldaclass2, datLDAtrain$cancer)

ldaclass2 benign cancer
benign      23      8
cancer       2     17
> mean(ldaclass2==datLDAtrain$cancer)
[1] 0.8
>
```

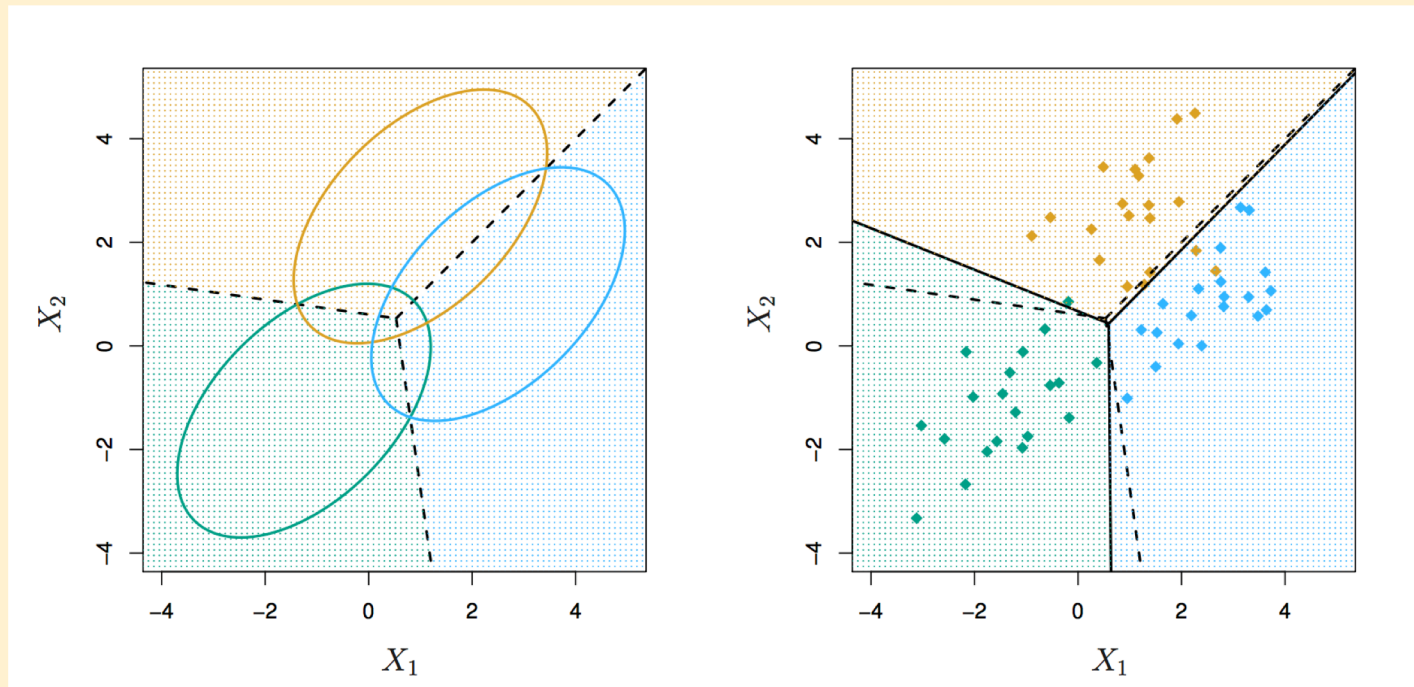

LDA when $p > 1$



Interested reader can look at the math in ISLR Ch. 4

However, an important note here is that SD's for each variable and correlations between them are assumed to be the same for each group – this blows up to a lot of parameter assumptions when number of predictors gets large

Illustration: $p = 2$ and $K = 3$

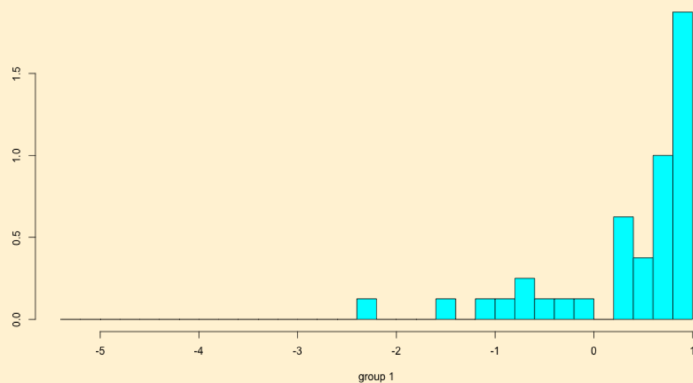
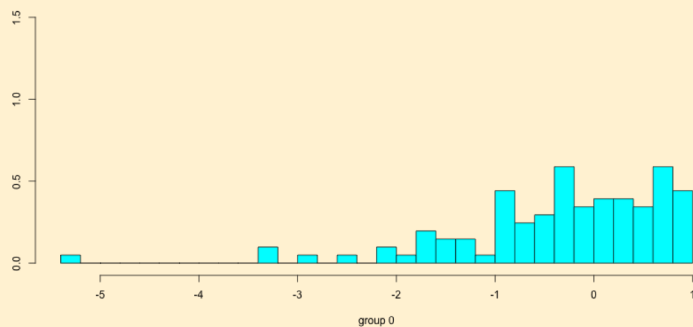


Here $\pi_1 = \pi_2 = \pi_3 = 1/3$

Dashed lines are the *Bayes decision boundaries* – if they were known then they would give fewest misclassification errors on average among all possible classifiers

LDA with 2 predictors in R

Go back to the breast cancer data



```
> ##### Example on breast data with 2 predictors
> ## First normalize the predictors
> dat$LDfz <- (dat$MRI_LD_Tfinal-mean(dat$MRI_LD_Tfinal,na.rm=TRUE))/sd(dat$MRI_LD_Tfinal,na.rm=TRUE)
> dat$ftvChgT0_T2 <- (dat$ftvPePctChgT0_T2-mean(dat$ftvPePctChgT0_T2,na.rm=TRUE))/sd(dat$ftvPePctChgT0_T2,na.rm=TRUE)
> ldaMult <- lda(pCR ~ MRI_LD_Tfinal + ftvPePctChgT0_T2, data=dat)
> ldaMult
Call:
lda(pCR ~ MRI_LD_Tfinal + ftvPePctChgT0_T2, data = dat)

Prior probabilities of groups:
      0      1
0.7183099 0.2816901

Group means:
      MRI_LD_Tfinal ftvPePctChgT0_T2
0      34.90196      -67.13666
1      16.77500      -93.01038

Coefficients of linear discriminants:
              LD1
MRI_LD_Tfinal  -0.030306250
ftvPePctChgT0_T2 -0.005432031
> plot(ldaMult)
>
```

Getting probabilities from $\delta_k(x)$

Once we have estimates $\hat{\delta}_k(x)$ we can extract class probabilities

$$\widehat{\Pr}(Y = k|X = x) = \frac{e^{\hat{\delta}_k(x)}}{\sum_{l=1}^K e^{\hat{\delta}_l(x)}}$$

So classifying to largest $\hat{\delta}_k(x)$ gives the same answers as classifying to the largest $\widehat{\Pr}(Y = k|X = x)$

E.g., if $K = 2$, then classify to class 2 if

$\widehat{\Pr}(Y = k|X = x) \geq 0.5$, otherwise classify as class 1

Summary

- Supervised learning methods for classification
 - Logistic and multinomial regression
 - Linear discriminant analysis

Announcement

On Friday, April 26th, Dr. Geoff Tison will present at our Friday Science Meeting.

Title: Using Machine Learning in Medical Research

The talk will provide an overview of machine learning methods, their relationship to more traditional statistical techniques and examples of applications to medical research.

Dr. Geoff Tison, MD MPH is a cardiologist and researcher in the division of Cardiology at UCSF. His research program focuses broadly on cardiovascular disease prevention, with particular expertise in the applications of machine learning to medical data.

Pizza will be served. We hope to see you there!

When & Where

Friday, April 26th

1-2pm, Room 2100

Mission Hall

550-16th Street, 2nd Floor

San Francisco, CA 94158

Next Lecture

- Matrix algebra
- Loss functions and optimization
 - Gradient descent
- Cross-validation methods
- Neural networks?