S632 HW2

Erik Parker

January 27th, 2018

1. ELM 2.1: The dataset *wbca* comes from a study of breast cancer in Wisconsin. There are 681 cases of potentially cancerous tumors of which 238 are actually malignant. The purpose of the study was the determin whether a new procedure could be effective in determining tumor status.

```
rm(list = ls())

library(ggplot2)
library(faraway)
library(dplyr)
library(pscl)

cancer <- wbca
```

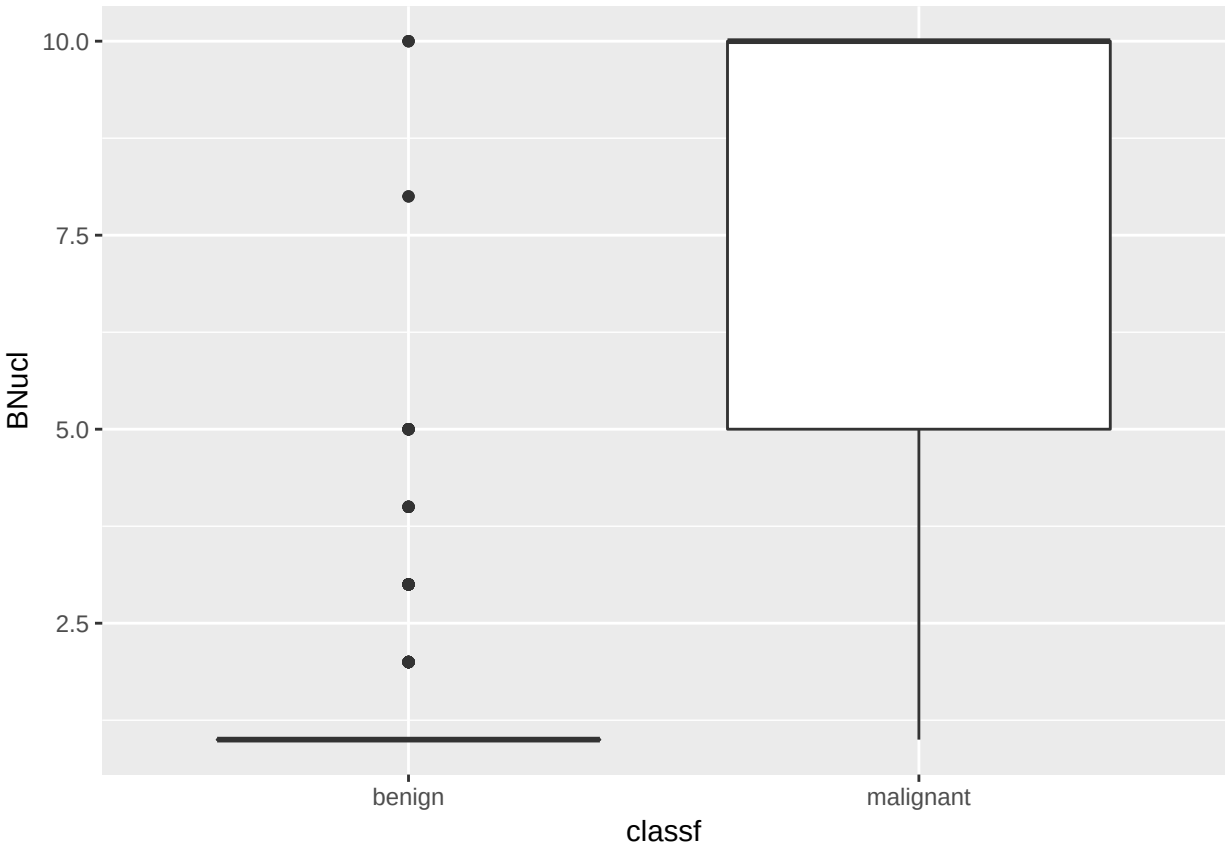
a) Plot the relationship between the classification and *BNucl*

i. Explain why $plot(Class \sim BNucl, wbca)$ does not work well

> The above plotting style does not work well because *Class* is a binomial response variable, and so has only two possibilities - 0 and 1. When plotted against *BNucl*, or really any other regressor, the result is just a series of overlapping points at the two y-axis values, arranged along the range of x-axis values. This isn't useful because it is impossible to tell what the frequency of each of the represented x-axis values is - they all just show up as single points.

ii. Create a factor version of *Class* and replicate the first panel of Fig. 2.1. Comment on the shape of the boxplots.

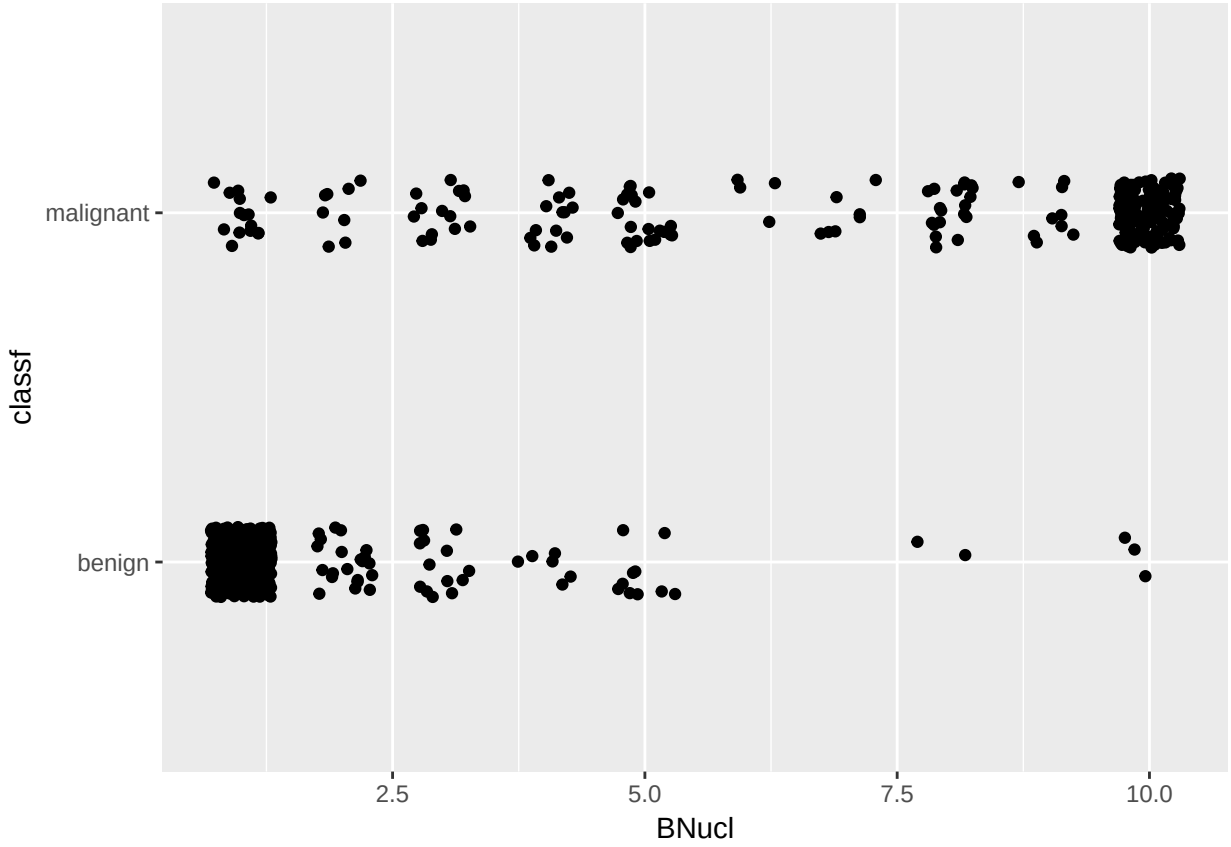
```
cancer$classf <- ifelse(cancer$Class == "0", "malignant", "benign")

ggplot(cancer, aes(x = classf, y = BNucl)) + geom_boxplot()
```



From these boxplots, it is clear that the vast majority of malignant tumors have very high bare nuclei scores (BNucl), though there is a little variation on the lower end of these scores. On the other hand, virtually all benign tumors have BNucl scores of 0, with very few cases above that. So for these data, a BNucl score above 0 is largely indicative of a malignant tumor.

iii. Produce a version of the second panel of Fig. 2.1. What does this plot say about the distribution?

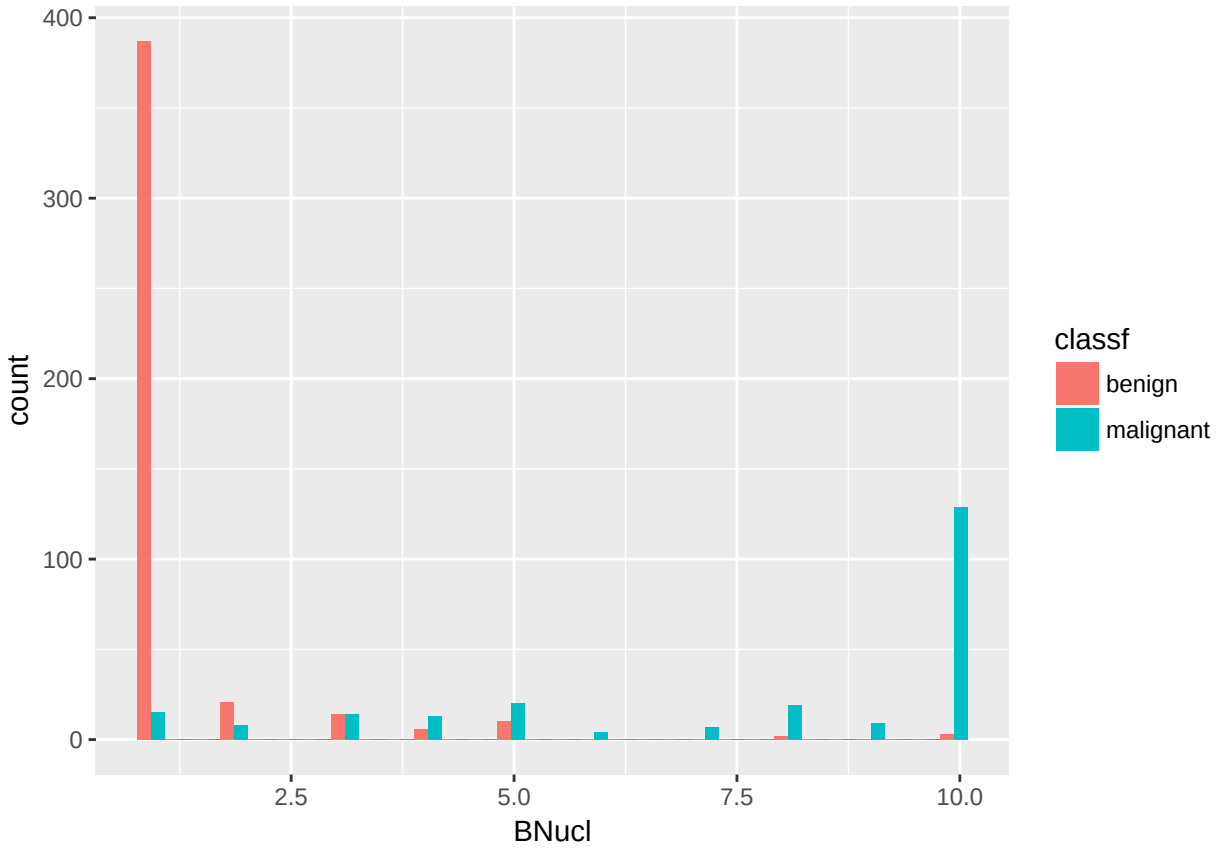
```
ggplot(cancer, aes(BNucl, classf)) + geom_point(position = position_jitter(width = 0.3,
    height = 0.1))
```



This plot, like the previous boxplot, shows us that the *BNucl* distributions of malignant and benign tumors are concentrated at the extreme high and low ends respectively. But, malignant tumors show more variation along the range of *BNucl* values in their distribution than do benign tumors.

iv. Produce the histogram from Fig. 2.2.

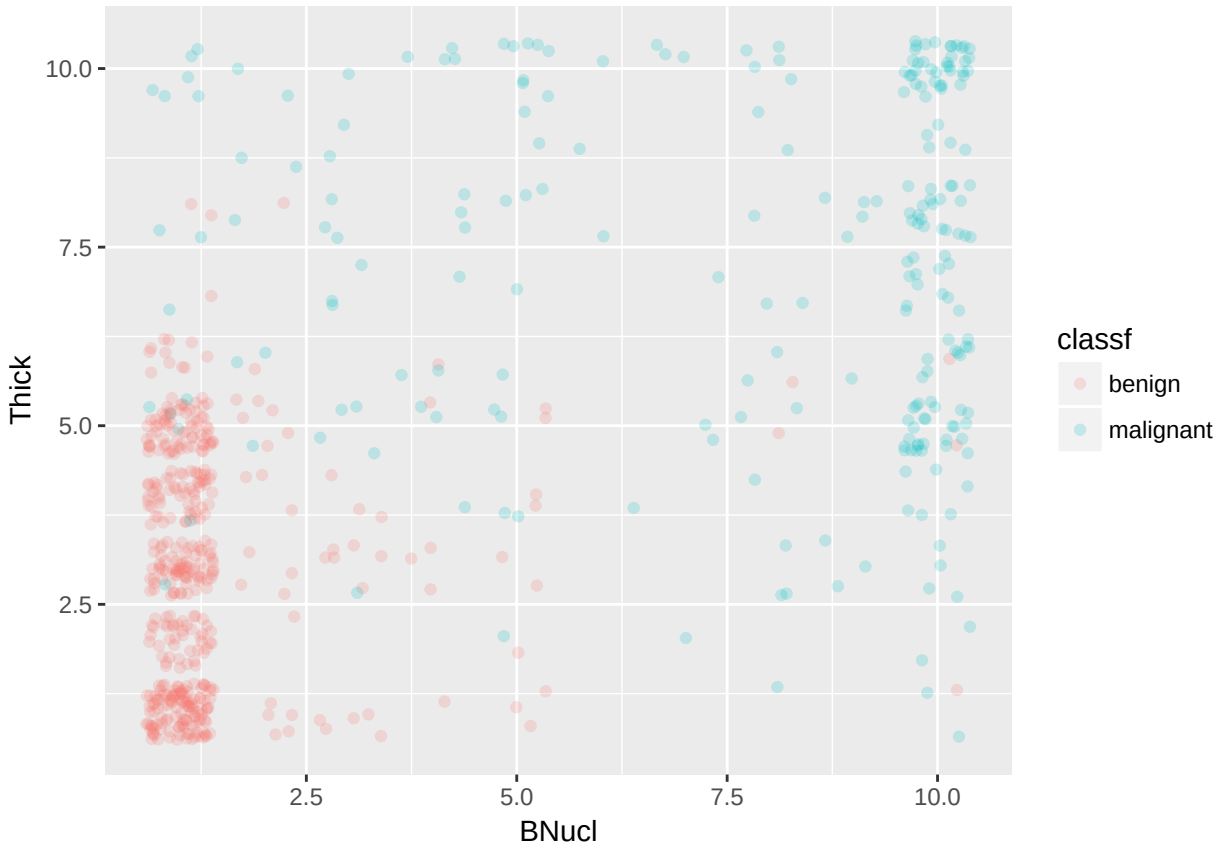
```
ggplot(cancer, aes(x = BNucl, fill = classf)) + geom_histogram(position = position_dodge())
```



Like all of the previous plots, this one also shows that benign tumors largely have scores of 0 for *BNucl*, though they do show some other values which are mostly concentrated on the low end. Malignant tumors also largely have *BNucl* scores pinned to the top end of the distribution (10), and then show more variation in their distribution, with scores across the high, middle, and low end of *BNucl* values.

b) Produce a version of Fig. 2.3 for the predictors *BNucl* and *Thick*. Produce an alternative version with only one panel but where the two types are plotted differently. Compare the two plots.

```
ggplot(cancer, aes(x = BNucl, y = Thick, color = classf)) + geom_point(alpha = 0.2,
    position = position_jitter())
```



This cool plot shows us that there is a large amount of segregation between benign and malignant tumors when the predictors *BNucl* and *Thick* are considered together. Benign tumors generally have lower scores for both predictors - so their points clump in the bottom left, while malignant tumors show higher scores for both generally - and so clump in the top right. Though again, we see much more variation in the values of not just *BNucl*, but also *Thick* for malignant tumors, so their points are more widely distributed around the space of the figure.

c) Fit a binary regression with *Class* as the response and the other nine variables as predictors. Report the residual deviance and associated degrees of freedom. Can those measures be used to determine if this model fits the data?

```
m.cancer <- glm(Class ~ Adhes + BNucl + Chrom + Epith + Mitos + NNucl + Thick +
    UShap + USize, family = binomial, cancer)

summary(m.cancer)
```

```
##
## Call:
## glm(formula = Class ~ Adhes + BNucl + Chrom + Epith + Mitos +
##     NNucl + Thick + UShap + USize, family = binomial, data = cancer)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.48282  -0.01179   0.04739   0.09678   3.06425
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept) 11.16678    1.41491   7.892 2.97e-15 ***
## Adhes       -0.39681    0.13384  -2.965  0.00303 **
## BNucl       -0.41478    0.10230  -4.055 5.02e-05 ***
## Chrom       -0.56456    0.18728  -3.014  0.00257 **
## Epith       -0.06440    0.16595  -0.388  0.69795
## Mitos       -0.65713    0.36764  -1.787  0.07387 .
## NNucl       -0.28659    0.12620  -2.271  0.02315 *
## Thick       -0.62675    0.15890  -3.944 8.01e-05 ***
## UShap       -0.28011    0.25235  -1.110  0.26699
## USize        0.05718    0.23271   0.246  0.80589
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 881.388  on 680  degrees of freedom
## Residual deviance:  89.464  on 671  degrees of freedom
## AIC: 109.46
##
## Number of Fisher Scoring iterations: 8
```

```
1 - pchisq(791.9, 9)
```

```
## [1] 0
```

The residual deviance and associated degrees of freedom for this model are 89.464 and 671. This information can be used to determine how well the model fits the data, at least when compared to a null model with no regressors. To do this we can use the difference between the null and residual deviance, on a chi-squared distribution with p degrees of freedom. The result of this test, as shown above, is a very low p-value, letting us know that we can be confident that there is some relationship between the predictors and response.

d) Use AIC to determine the best subset of variables

```
step(m.cancer, scope = list(lower = ~1, upper = m.cancer), direction = "both")
```

```
## Start:  AIC=109.46
## Class ~ Adhes + BNucl + Chrom + Epith + Mitos + NNucl + Thick +
##     UShap + USize
##
##         Df Deviance    AIC
## - USize  1   89.523 107.52
## - Epith  1   89.613 107.61
## - UShap  1   90.627 108.63
## <none>       89.464 109.46
## - Mitos  1   93.551 111.55
## - NNucl  1   95.204 113.20
## - Adhes  1   98.844 116.84
## - Chrom  1   99.841 117.84
## - BNucl  1  109.000 127.00
## - Thick  1  110.239 128.24
##
## Step:  AIC=107.52
## Class ~ Adhes + BNucl + Chrom + Epith + Mitos + NNucl + Thick +
##     UShap
```

```
##
##         Df Deviance    AIC
## - Epith  1   89.662 105.66
## - UShap  1   91.355 107.36
## <none>       89.523 107.52
## + USize  1   89.464 109.46
## - Mitos  1   93.552 109.55
## - NNucl  1   95.231 111.23
## - Adhes  1   99.042 115.04
## - Chrom  1  100.153 116.15
## - BNucl  1  109.064 125.06
## - Thick  1  110.465 126.47
##
## Step:  AIC=105.66
## Class ~ Adhes + BNucl + Chrom + Mitos + NNucl + Thick + UShap
##
##         Df Deviance    AIC
## <none>       89.662 105.66
## - UShap  1   91.884 105.88
## + Epith  1   89.523 107.52
## + USize  1   89.613 107.61
## - Mitos  1   93.714 107.71
## - NNucl  1   95.853 109.85
## - Adhes  1  100.126 114.13
## - Chrom  1  100.844 114.84
## - BNucl  1  109.762 123.76
## - Thick  1  110.632 124.63
```

```
##
## Call:  glm(formula = Class ~ Adhes + BNucl + Chrom + Mitos + NNucl +
##     Thick + UShap, family = binomial, data = cancer)
##
## Coefficients:
## (Intercept)        Adhes        BNucl        Chrom        Mitos
##     11.0333      -0.3984      -0.4192      -0.5679      -0.6456
##       NNucl        Thick        UShap
##     -0.2915      -0.6216      -0.2541
##
## Degrees of Freedom: 680 Total (i.e. Null);  673 Residual
## Null Deviance:       881.4
## Residual Deviance: 89.66     AIC: 105.7
```

```
m.0 <- glm(Class ~ 1, family = binomial, cancer)

step(m.0, scope = list(lower = m.0, upper = m.cancer), direction = "both")
```

```
## Start:  AIC=883.39
## Class ~ 1
##
##         Df Deviance    AIC
## + USize  1   251.77 255.77
## + UShap  1   265.32 269.32
## + BNucl  1   331.46 335.46
## + Chrom  1   379.41 383.41
## + Epith  1   450.30 454.30
```

```
## + Thick  1   451.69 455.69
## + NNucl  1   454.76 458.76
## + Adhes  1   459.10 463.10
## + Mitos  1   713.09 717.09
## <none>       881.39 883.39
##
## Step:  AIC=255.77
## Class ~ USize
##
##         Df Deviance    AIC
## + BNucl  1   161.41 167.41
## + Thick  1   190.83 196.83
## + Chrom  1   200.26 206.26
## + NNucl  1   216.29 222.29
## + UShap  1   218.28 224.28
## + Adhes  1   224.85 230.85
## + Epith  1   236.32 242.32
## + Mitos  1   237.97 243.97
## <none>       251.77 255.77
## - USize  1   881.39 883.39
##
## Step:  AIC=167.41
## Class ~ USize + BNucl
##
##         Df Deviance    AIC
## + Thick  1   127.76 135.76
## + NNucl  1   143.41 151.41
## + Chrom  1   144.09 152.09
## + UShap  1   148.43 156.43
## + Mitos  1   153.26 161.26
## + Adhes  1   155.48 163.48
## + Epith  1   156.64 164.64
## <none>       161.41 167.41
## - BNucl  1   251.77 255.77
## - USize  1   331.46 335.46
##
## Step:  AIC=135.76
## Class ~ USize + BNucl + Thick
##
##         Df Deviance    AIC
## + Chrom  1   112.91 122.91
## + NNucl  1   113.34 123.34
## + Adhes  1   117.23 127.23
## + UShap  1   122.43 132.43
## + Epith  1   123.94 133.94
## + Mitos  1   124.98 134.98
## <none>       127.76 135.76
## - Thick  1   161.41 167.41
## - USize  1   184.19 190.19
## - BNucl  1   190.82 196.82
##
## Step:  AIC=122.91
## Class ~ USize + BNucl + Thick + Chrom
##
```

```
##         Df Deviance    AIC
## + Adhes  1   102.61 114.61
## + NNucl  1   104.50 116.50
## + Mitos  1   109.64 121.64
## + UShap  1   109.70 121.70
## + Epith  1   110.61 122.61
## <none>       112.91 122.91
## - Chrom  1   127.76 135.76
## - USize  1   129.19 137.19
## - Thick  1   144.09 152.09
## - BNucl  1   148.57 156.57
##
## Step:  AIC=114.61
## Class ~ USize + BNucl + Thick + Chrom + Adhes
##
##         Df Deviance    AIC
## + NNucl  1   95.042 109.04
## + Mitos  1   98.340 112.34
## + UShap  1  100.240 114.24
## <none>      102.608 114.61
## + Epith  1  101.447 115.45
## - USize  1  111.384 121.38
## - Adhes  1  112.913 122.91
## - Chrom  1  117.230 127.23
## - BNucl  1  129.494 139.49
## - Thick  1  139.226 149.23
##
## Step:  AIC=109.04
## Class ~ USize + BNucl + Thick + Chrom + Adhes + NNucl
##
##         Df Deviance    AIC
## + Mitos  1   90.923 106.92
## - USize  1   96.494 108.49
## <none>       95.042 109.04
## + UShap  1   93.713 109.71
## + Epith  1   94.777 110.78
## - NNucl  1  102.608 114.61
## - Adhes  1  104.496 116.50
## - Chrom  1  105.792 117.79
## - BNucl  1  120.039 132.04
## - Thick  1  129.419 141.42
##
## Step:  AIC=106.92
## Class ~ USize + BNucl + Thick + Chrom + Adhes + NNucl + Mitos
##
##         Df Deviance    AIC
## - USize  1   91.884 105.88
## <none>       90.923 106.92
## + UShap  1   89.613 107.61
## + Epith  1   90.627 108.63
## - Mitos  1   95.042 109.04
## - NNucl  1   98.340 112.34
## - Adhes  1  100.870 114.87
## - Chrom  1  102.551 116.55
```

```
## - Thick  1  115.780 129.78
## - BNucl  1  116.676 130.68
##
## Step:  AIC=105.88
## Class ~ BNucl + Thick + Chrom + Adhes + NNucl + Mitos
##
##         Df Deviance    AIC
## + UShap  1   89.662 105.66
## <none>       91.884 105.88
## + USize  1   90.923 106.92
## + Epith  1   91.355 107.36
## - Mitos  1   96.494 108.49
## - NNucl  1  103.711 115.71
## - Adhes  1  105.473 117.47
## - Chrom  1  109.699 121.70
## - BNucl  1  124.813 136.81
## - Thick  1  130.842 142.84
##
## Step:  AIC=105.66
## Class ~ BNucl + Thick + Chrom + Adhes + NNucl + Mitos + UShap
##
##         Df Deviance    AIC
## <none>       89.662 105.66
## - UShap  1   91.884 105.88
## + Epith  1   89.523 107.52
## + USize  1   89.613 107.61
## - Mitos  1   93.714 107.71
## - NNucl  1   95.853 109.85
## - Adhes  1  100.126 114.13
## - Chrom  1  100.844 114.84
## - BNucl  1  109.762 123.76
## - Thick  1  110.632 124.63
##
## Call:  glm(formula = Class ~ BNucl + Thick + Chrom + Adhes + NNucl +
##     Mitos + UShap, family = binomial, data = cancer)
##
## Coefficients:
## (Intercept)        BNucl        Thick        Chrom        Adhes
##     11.0333      -0.4192      -0.6216      -0.5679      -0.3984
##       NNucl        Mitos        UShap
##     -0.2915      -0.6456      -0.2541
##
## Degrees of Freedom: 680 Total (i.e. Null);  673 Residual
## Null Deviance:       881.4
## Residual Deviance: 89.66     AIC: 105.7
```

Using AIC, we see that the best subset of regressors tested is contained in the model *Class ~ BNucl + Thick + Chrom + Adhes + NNucl + Mitos + UShap*.

e) Suppose that cancer is classified as benign if *p>0.5* and malignant if *p<0.5*. Compute the number of errors of both types that will be made if this method is applied to the current data with the reduced model.

```
m.cancer.red <- glm(Class ~ BNucl + Thick + Chrom + Adhes + NNucl + Mitos +
    UShap, family = binomial, cancer)

cancer.m <- na.omit(cancer)
cancer.m <- mutate(cancer.m, predprob = predict(m.cancer.red, type = "response"))
cancer.m <- mutate(cancer.m, predout = ifelse(predprob < 0.5, "malignant", "benign"))
xtabs(~classf + predout, cancer.m)
```

```
##            predout
## classf      benign malignant
##   benign       434         9
##   malignant     11       227
```

So, using the reduced model and the current data: there will be a total of 20 misclassifications. 9 patients with benign tumors will be misdiagnosed with malignant ones, and 11 patients with malignant tumors will be misdiagnosed with benign ones.

f) Suppose we change the cutoff to 0.9, so that $p<0.9$ is malignant and $p>0.9$ is benign. Compute the number of errors in this case.

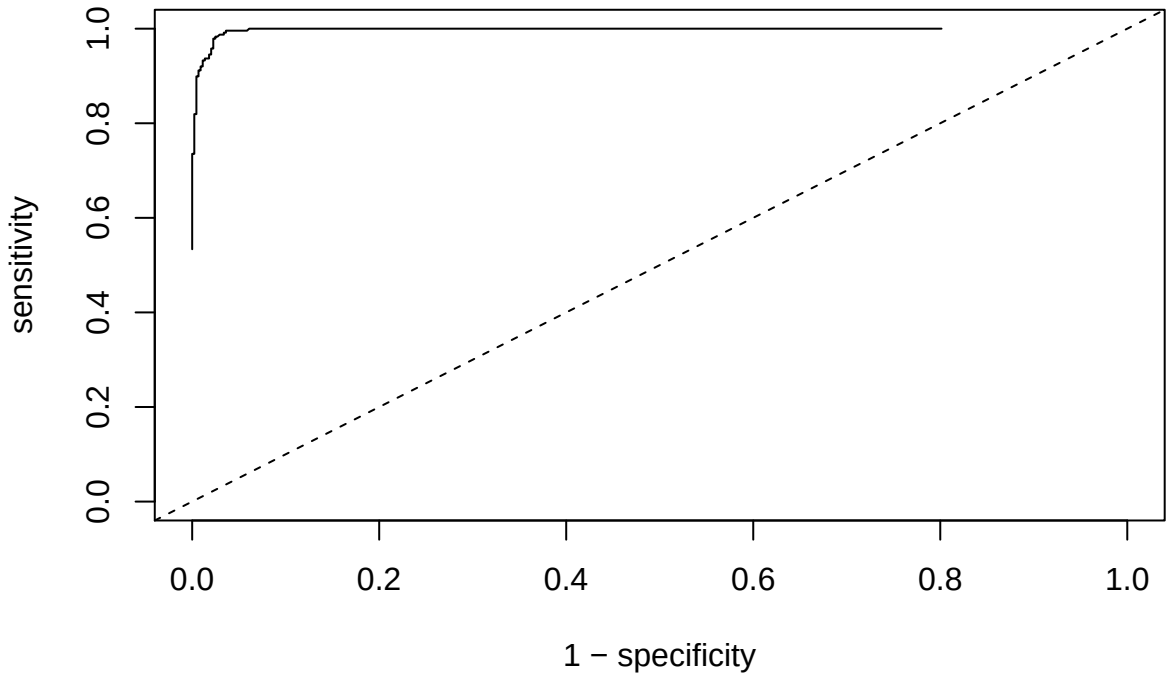
```
cancer.m <- mutate(cancer.m, predout = ifelse(predprob < 0.9, "malignant", "benign"))
xtabs(~classf + predout, cancer.m)
```

```
##            predout
## classf      benign malignant
##   benign       427        16
##   malignant      1       237
```

In this case, only one patient will be misdagnosed as having a benign tumor when it is actually malignant. But now 16 will be labeled as having malignant tumors, when they are actually benign. Overall, the total number of errors is lower, and most importantly the number of missed diagnoses is much lower.

g) Produce an ROC plot and comment on the effectiveness of the new diagnostic test.

```
thresh <- seq(0.001, 0.9999, 0.001)
sensitivity <- numeric(length(thresh))
specificity <- numeric(length(thresh))
for (j in seq(along = thresh)) {
    pp <- ifelse(cancer.m$predprob < thresh[j], "malignant", "benign")
    xx <- xtabs(~classf + pp, cancer.m)
    specificity[j] <- xx[1, 1]/(xx[1, 1] + xx[1, 2])
    sensitivity[j] <- xx[2, 2]/(xx[2, 1] + xx[2, 2])
}
plot(1 - specificity, sensitivity, type = "l", ylim = c(0, 1), xlim = c(0, 1))
abline(0, 1, lty = 2)
```



From the above ROC plot, we can see that the effectiveness of the new diagnostic test is very high - as the curve is pulled very far into the top left corner. This means that the test has a very high true positive rate (the sensitivity) and a low false positive rate (1 - specificity).

h) Assign every third observation to a test set, and the remaining two thirds to a training set. Use the training set to determine the model and the test set to assess its performance.

```
test <- cancer[seq(3, nrow(cancer), 3), ]
train <- cancer[-c(seq(3, nrow(cancer), 3)), ]

m.train <- glm(Class ~ Adhes + BNucl + Chrom + Epith + Mitos + NNucl + Thick +
    UShap + USize, family = binomial, train)

step(m.train, scope = list(lower = ~1, upper = m.train), direction = "both")
```

```
## Start:  AIC=77.65
## Class ~ Adhes + BNucl + Chrom + Epith + Mitos + NNucl + Thick +
##     UShap + USize
##
##         Df Deviance    AIC
## - Epith  1   58.340 76.340
## - USize  1   58.880 76.880
## <none>       57.651 77.651
## - Mitos  1   60.712 78.712
## - UShap  1   61.450 79.450
## - Chrom  1   65.983 83.983
## - BNucl  1   67.373 85.373
## - NNucl  1   67.538 85.538
## - Adhes  1   68.073 86.073
## - Thick  1   71.162 89.162
##
## Step:  AIC=76.34
```

```
## Class ~ Adhes + BNucl + Chrom + Mitos + NNucl + Thick + UShap +
##     USize
##
##         Df Deviance    AIC
## - USize  1   59.536 75.536
## <none>       58.340 76.340
## - Mitos  1   61.264 77.264
## + Epith  1   57.651 77.651
## - UShap  1   61.702 77.702
## - Chrom  1   66.515 82.515
## - BNucl  1   67.402 83.402
## - NNucl  1   67.556 83.556
## - Adhes  1   68.310 84.310
## - Thick  1   72.311 88.311
##
## Step:  AIC=75.54
## Class ~ Adhes + BNucl + Chrom + Mitos + NNucl + Thick + UShap
##
##         Df Deviance    AIC
## <none>       59.536 75.536
## - UShap  1   61.894 75.894
## - Mitos  1   62.329 76.329
## + USize  1   58.340 76.340
## + Epith  1   58.880 76.880
## - Chrom  1   66.762 80.762
## - NNucl  1   67.576 81.576
## - BNucl  1   68.332 82.332
## - Adhes  1   68.359 82.359
## - Thick  1   72.363 86.363
##
## Call:  glm(formula = Class ~ Adhes + BNucl + Chrom + Mitos + NNucl +
##     Thick + UShap, family = binomial, data = train)
##
## Coefficients:
## (Intercept)        Adhes        BNucl        Chrom        Mitos
##     11.5571      -0.4249      -0.3341      -0.5963      -0.5822
##       NNucl        Thick        UShap
##     -0.4192      -0.6037      -0.2943
##
## Degrees of Freedom: 453 Total (i.e. Null);  446 Residual
## Null Deviance:       592.8
## Residual Deviance: 59.54     AIC: 75.54
```

```r
m.0.train <- glm(Class ~ 1, family = binomial, train)

step(m.0.train, scope = list(lower = m.0.train, upper = m.train), direction = "both")
```

```
## Start:  AIC=594.8
## Class ~ 1
##
##         Df Deviance    AIC
## + UShap  1   176.35 180.35
## + USize  1   176.41 180.41
## + BNucl  1   237.90 241.90
```

```
## + Chrom  1   269.46 273.46
## + NNucl  1   291.30 295.30
## + Thick  1   305.54 309.54
## + Adhes  1   307.49 311.49
## + Epith  1   323.13 327.13
## + Mitos  1   445.21 449.21
## <none>       592.80 594.80
##
## Step:  AIC=180.35
## Class ~ UShap
##
##         Df Deviance    AIC
## + BNucl  1   120.87 126.87
## + Thick  1   127.43 133.43
## + Chrom  1   130.36 136.36
## + Mitos  1   138.88 144.88
## + Adhes  1   139.03 145.03
## + NNucl  1   145.38 151.38
## + USize  1   149.40 155.40
## + Epith  1   163.04 169.04
## <none>       176.35 180.35
## - UShap  1   592.80 594.80
##
## Step:  AIC=126.87
## Class ~ UShap + BNucl
##
##         Df Deviance    AIC
## + Thick  1   95.677 103.68
## + Mitos  1   98.831 106.83
## + Chrom  1  102.630 110.63
## + NNucl  1  103.709 111.71
## + USize  1  109.443 117.44
## + Adhes  1  110.245 118.25
## + Epith  1  115.023 123.02
## <none>      120.874 126.87
## - BNucl  1  176.354 180.35
## - UShap  1  237.904 241.90
##
## Step:  AIC=103.68
## Class ~ UShap + BNucl + Thick
##
##         Df Deviance     AIC
## + Adhes  1   79.047  89.047
## + Chrom  1   81.318  91.318
## + NNucl  1   83.394  93.394
## + Mitos  1   86.791  96.791
## + USize  1   90.684 100.684
## + Epith  1   92.311 102.311
## <none>       95.677 103.677
## - Thick  1  120.874 126.874
## - BNucl  1  127.434 133.434
## - UShap  1  137.947 143.947
##
## Step:  AIC=89.05
```

```
## Class ~ UShap + BNucl + Thick + Adhes
##
##         Df Deviance     AIC
## + Chrom  1   69.824  81.824
## + NNucl  1   70.451  82.451
## + Mitos  1   75.017  87.017
## <none>       79.047  89.047
## + Epith  1   78.652  90.652
## + USize  1   78.733  90.733
## - BNucl  1   92.158 100.158
## - Adhes  1   95.677 103.677
## - UShap  1  102.764 110.764
## - Thick  1  110.245 118.245
##
## Step:  AIC=81.82
## Class ~ UShap + BNucl + Thick + Adhes + Chrom
##
##         Df Deviance     AIC
## + NNucl  1   62.329  76.329
## + Mitos  1   67.576  81.576
## <none>       69.824  81.824
## + Epith  1   69.761  83.761
## + USize  1   69.801  83.801
## - BNucl  1   77.634  87.634
## - Chrom  1   79.047  89.047
## - UShap  1   80.377  90.377
## - Adhes  1   81.318  91.318
## - Thick  1   96.172 106.172
##
## Step:  AIC=76.33
## Class ~ UShap + BNucl + Thick + Adhes + Chrom + NNucl
##
##         Df Deviance    AIC
## + Mitos  1   59.536 75.536
## <none>       62.329 76.329
## - UShap  1   65.087 77.087
## + USize  1   61.264 77.264
## + Epith  1   61.731 77.731
## - NNucl  1   69.824 81.824
## - Chrom  1   70.451 82.451
## - BNucl  1   70.875 82.875
## - Adhes  1   72.399 84.399
## - Thick  1   87.178 99.178
##
## Step:  AIC=75.54
## Class ~ UShap + BNucl + Thick + Adhes + Chrom + NNucl + Mitos
##
##         Df Deviance    AIC
## <none>       59.536 75.536
## - UShap  1   61.894 75.894
## - Mitos  1   62.329 76.329
## + USize  1   58.340 76.340
## + Epith  1   58.880 76.880
## - Chrom  1   66.762 80.762
```

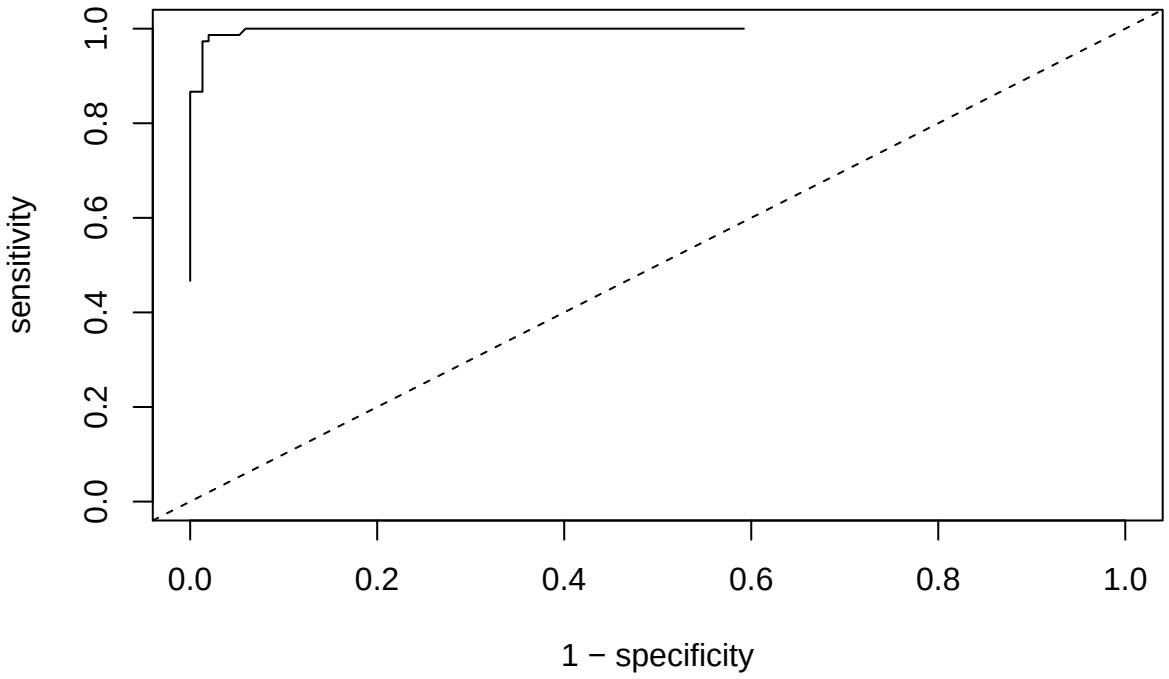
```
## - NNucl  1   67.576 81.576
## - BNucl  1   68.332 82.332
## - Adhes  1   68.359 82.359
## - Thick  1   72.363 86.363
```

```
##
## Call:  glm(formula = Class ~ UShap + BNucl + Thick + Adhes + Chrom +
##     NNucl + Mitos, family = binomial, data = train)
##
## Coefficients:
## (Intercept)        UShap        BNucl        Thick        Adhes
##     11.5571      -0.2943      -0.3341      -0.6037      -0.4249
##       Chrom        NNucl        Mitos
##     -0.5963      -0.4192      -0.5822
##
## Degrees of Freedom: 453 Total (i.e. Null);  446 Residual
## Null Deviance:       592.8
## Residual Deviance: 59.54     AIC: 75.54
```

```
# Best model determined based on training set
m.train.best <- glm(Class ~ Adhes + BNucl + Chrom + Mitos + NNucl + Thick +
    UShap, family = binomial, train)

# Now evaluating model on test set
test.m <- na.omit(test)
test.m <- mutate(test.m, predprob = predict(m.train.best, newdata = test, type = "response"))
test.m <- mutate(test.m, predout = ifelse(predprob < 0.5, "malignant", "benign"))
xtabs(~classf + predout, test.m)
```

```
##            predout
## classf      benign malignant
##   benign       150         2
##   malignant      5        70
```

```
thresh <- seq(0.001, 0.9999, 0.001)
sensitivity <- numeric(length(thresh))
specificity <- numeric(length(thresh))
for (j in seq(along = thresh)) {
    pp <- ifelse(test.m$predprob < thresh[j], "malignant", "benign")
    xx <- xtabs(~classf + pp, test.m)
    specificity[j] <- xx[1, 1]/(xx[1, 1] + xx[1, 2])
    sensitivity[j] <- xx[2, 2]/(xx[2, 1] + xx[2, 2])
}

plot(1 - specificity, sensitivity, type = "l", ylim = c(0, 1), xlim = c(0, 1))
abline(0, 1, lty = 2)
```



Here we see from the ROC plot, that while this model is also very good, it is a little worse than the previous one. This makes sense though as the previous model uses the same data it was constructed based on to test its predictive ability, while here we used two seperate datasets. So, when we construct a model in this more "correct" way, we see that it is still quite good for these data.