BST 210 Project: Check-In 2

Daniel Herrera, Willow Duffell, Lauren Mock

11/6/2021

Group 4 Members:

Daniel Herrera
Willow Duffell
Lauren Mock

Logistic regression to predict survival

We want to predict the probability of survival after transplantation, given known covariates that can be measured before transplantation.

question: what determines dosage amount? so should we include this as a predictor or not?

```
# remove variables that are measured after transplantation
predictors <- bone[,c(1:24,37)] # selects only predictors and survival (the outcome)

# logistic model with all covariates
mod_all_vars <- glm(survival_status ~ ., family = binomial(), data = predictors)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(mod_all_vars)
```

```
##
## Call:
## glm(formula = survival_status ~ ., family = binomial(), data = predictors)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
##  -8.49     0.00     0.00     0.00     8.49
##
## Coefficients: (8 not defined because of singularities)
##                                  Estimate Std. Error    z value Pr(>|z|)
## (Intercept)                     5.225e+16  9.288e+08   56248972   <2e-16 ***
## donor_age                      -1.101e+13  1.259e+06   -8746490   <2e-16 ***
## donor_age_below_35yes          -5.032e+14  2.056e+07  -24476219   <2e-16 ***
## donor_ABOA                      8.323e+13  1.326e+07    6274183   <2e-16 ***
## donor_ABOAB                     4.244e+13  2.241e+07    1893738   <2e-16 ***
## donor_ABOB                     -1.613e+14  1.684e+07   -9577145   <2e-16 ***
```

```
## donor_CMVabsent                  -2.471e+15  6.966e+07  -35473036   <2e-16 ***
## donor_CMVpresent                 -2.322e+15  6.747e+07  -34408475   <2e-16 ***
## recipient_age                    -2.082e+14  3.746e+06  -55584889   <2e-16 ***
## recipient_age_below_10yes        -4.587e+15  7.250e+07  -63264479   <2e-16 ***
## recipient_age_int10_20           -3.893e+15  7.521e+07  -51763853   <2e-16 ***
## recipient_age_int5_10             7.178e+14  2.117e+07   33911823   <2e-16 ***
## recipient_gendermale             -4.457e+14  1.283e+07  -34736324   <2e-16 ***
## recipient_body_mass               3.943e+13  6.992e+05   56394815   <2e-16 ***
## recipient_ABO0                    5.992e+15  1.053e+08   56929168   <2e-16 ***
## recipient_ABOA                    6.374e+15  1.044e+08   61066102   <2e-16 ***
## recipient_ABOAB                   8.149e+15  1.069e+08   76217000   <2e-16 ***
## recipient_ABOB                    6.529e+15  1.056e+08   61844593   <2e-16 ***
## recipient_rhminus                -7.362e+15  7.767e+07  -94792174   <2e-16 ***
## recipient_rhplus                 -6.799e+15  7.607e+07  -89378726   <2e-16 ***
## recipient_CMVabsent              -3.751e+15  8.196e+07  -45766000   <2e-16 ***
## recipient_CMVpresent             -6.393e+15  6.178e+07 -103473019   <2e-16 ***
## diseaseAML                        2.363e+14  1.840e+07   12842621   <2e-16 ***
## diseasechronic                    5.964e+14  1.606e+07   37141089   <2e-16 ***
## diseaselymphoma                   3.793e+15  2.872e+07  132048777   <2e-16 ***
## diseasenonmalignant               3.783e+14  1.709e+07   22136973   <2e-16 ***
## disease_groupnonmalignant                NA         NA         NA       NA
## gender_matchother                -3.012e+14  1.658e+07  -18166742   <2e-16 ***
## ABO_matchmatched                  4.224e+14  1.357e+07   31117596   <2e-16 ***
## ABO_matchmismatched                      NA         NA         NA       NA
## CMV_status0                       3.104e+15  8.549e+07   36308361   <2e-16 ***
## CMV_status1                       2.987e+15  8.144e+07   36680277   <2e-16 ***
## CMV_status2                       6.102e+15  6.789e+07   89891007   <2e-16 ***
## CMV_status3                       5.696e+15  6.186e+07   92069476   <2e-16 ***
## HLA_match..out.of.10.            -4.401e+15  9.116e+07  -48277177   <2e-16 ***
## HLA_mismatchmismatched           -2.659e+15  8.186e+07  -32478517   <2e-16 ***
## antigen0                          1.592e+15  7.991e+07   19917980   <2e-16 ***
## antigen1                         -4.390e+15  2.072e+08  -21191309   <2e-16 ***
## antigen2                         -7.893e+15  2.735e+08  -28859725   <2e-16 ***
## antigen3                         -9.568e+15  3.512e+08  -27247229   <2e-16 ***
## allele0                                  NA         NA         NA       NA
## allele1                           4.897e+15  1.805e+08   27128269   <2e-16 ***
## allele2                           2.226e+15  1.025e+08   21723064   <2e-16 ***
## allele3                                  NA         NA         NA       NA
## allele4                                  NA         NA         NA       NA
## HLA_group_1matched                       NA         NA         NA       NA
## HLA_group_1mismatched                    NA         NA         NA       NA
## HLA_group_1one_allel             -1.152e+15  4.117e+07  -27983055   <2e-16 ***
## HLA_group_1one_antigen            3.266e+14  3.059e+07   10675726   <2e-16 ***
## HLA_group_1three_diffs            1.063e+15  4.363e+07   24364093   <2e-16 ***
## HLA_group_1two_diffs                     NA         NA         NA       NA
## risk_grouplow                    -3.750e+14  1.327e+07  -28262664   <2e-16 ***
## stem_cell_sourceperipheral_blood -1.213e+15  1.299e+07  -93362641   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance:  254.51  on 184  degrees of freedom
## Residual deviance: 4397.33  on 140  degrees of freedom
```

```
##   (2 observations deleted due to missingness)
## AIC: 4487.3
##
## Number of Fisher Scoring iterations: 24
```

```
p_hats <- mod_all_vars$fitted.values
head(p_hats)
```

```
##            1            2            3            4            5            6
## 2.220446e-16 1.000000e+00 1.000000e+00 1.000000e+00 2.220446e-16 2.220446e-16
```

```
# only predicts 2 values, not sure why...

# regardless, we will definitely need to select some variables that make sense
```

Variables related to the donor

```
# variables related to the donor
mod_donor <- glm(survival_status ~ donor_age + donor_age_below_35 + donor_ABO + donor_CMV,
            family = binomial(), data = predictors)

head(mod_donor$fitted.values)
```

```
##         1         2         3         4         5         6
## 0.4402879 0.4244372 0.4065115 0.5376755 0.4498189 0.4356239
```

We can then decide to remove the variable for donor_age_below_35 because of multicolineary. Including this variable will not allow us to properly "hold other variables constant" since the variable age modifies this variable of age_below_35.

We see here that our predictors are pretty bad, even on a testing only dataset.

```
mod_donor <- glm(survival_status ~ donor_age + donor_ABO + donor_CMV,
            family = binomial(), data = predictors)
summary(mod_donor)
```

```
##
## Call:
## glm(formula = survival_status ~ donor_age + donor_ABO + donor_CMV,
##     family = binomial(), data = predictors)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3162  -1.1098  -0.7714   1.1937   1.7385
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      0.44441    1.69898   0.262    0.794
## donor_age        0.01637    0.01819   0.900    0.368
## donor_ABOA       0.26370    0.33777   0.781    0.435
```

```
## donor_ABOAB      -0.98789    0.70064  -1.410    0.159
## donor_ABOB       -0.09199    0.45044  -0.204    0.838
## donor_CMVabsent  -1.09646    1.56593  -0.700    0.484
## donor_CMVpresent -1.37539    1.57384  -0.874    0.382
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 257.69  on 186  degrees of freedom
## Residual deviance: 252.42  on 180  degrees of freedom
## AIC: 266.42
##
## Number of Fisher Scoring iterations: 4
```

```
prediction <- ifelse(mod_donor$fitted.values > 0.5, 1, 0)
confusionMatrix(data = as.factor(prediction),
                reference = as.factor(predictors$survival_status),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 73 58
##          1 29 27
##
##                Accuracy : 0.5348
##                  95% CI : (0.4605, 0.6079)
##     No Information Rate : 0.5455
##     P-Value [Acc > NIR] : 0.643915
##
##                   Kappa : 0.0343
##
##  Mcnemar's Test P-Value : 0.002683
##
##             Sensitivity : 0.3176
##             Specificity : 0.7157
##          Pos Pred Value : 0.4821
##          Neg Pred Value : 0.5573
##              Prevalence : 0.4545
##          Detection Rate : 0.1444
##    Detection Prevalence : 0.2995
##       Balanced Accuracy : 0.5167
##
##        'Positive' Class : 1
##
```

Variables related to the recipient

Here we have to deal with the issue of having missing values for the variable recipient_body_mass. We elect to remove the variables here since only two values are missing and a sample of 185 should be sufficient.

We can later choose to explore these cases individually to assess if there was any particular reason that these cases should be explored further.

Our model here is not great, an accuracy of 64.32%. It does not have a balanced sensitivity versus specificity and is actually better at predicting those among the population with outcome of 1, death. This may be to our benefit though, as we would really like to know, of those who died, how many are we predicting death.

```
#create subset of data with no missing values, ie remove 2 missing bmi
recip_complete <- predictors %>%
  select(c(survival_status, recipient_age, recipient_gender, recipient_body_mass,
           recipient_ABO, recipient_rh, recipient_CMV)) %>%
  drop_na()

mod_recip <- glm(survival_status ~ recipient_age + recipient_gender + recipient_body_mass +
                   recipient_ABO + as.factor(recipient_rh) + as.factor(recipient_CMV),
            family = binomial(), data = recip_complete)
summary(mod_recip)
```

```
##
## Call:
## glm(formula = survival_status ~ recipient_age + recipient_gender +
##     recipient_body_mass + recipient_ABO + as.factor(recipient_rh) +
##     as.factor(recipient_CMV), family = binomial(), data = recip_complete)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7688  -0.9855  -0.7422   1.1409   1.8842
##
## Coefficients:
##                                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)                      14.00011 1455.39766   0.010   0.9923
## recipient_age                    -0.04094    0.07014  -0.584   0.5594
## recipient_gendermale              0.19051    0.32644   0.584   0.5595
## recipient_body_mass               0.03560    0.01949   1.826   0.0678 .
## recipient_ABOO                    1.48006 2058.24309   0.001   0.9994
## recipient_ABOA                    1.81733 2058.24305   0.001   0.9993
## recipient_ABOAB                   1.83790 2058.24318   0.001   0.9993
## recipient_ABOB                    1.72790 2058.24309   0.001   0.9993
## as.factor(recipient_rh)minus    -16.39755 1455.39764  -0.011   0.9910
## as.factor(recipient_rh)plus     -15.68622 1455.39758  -0.011   0.9914
## as.factor(recipient_CMV)absent   -1.34419    0.68649  -1.958   0.0502 .
## as.factor(recipient_CMV)present  -1.14159    0.66511  -1.716   0.0861 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 254.51  on 184  degrees of freedom
## Residual deviance: 233.51  on 173  degrees of freedom
## AIC: 257.51
##
## Number of Fisher Scoring iterations: 14
```

```
prediction <- ifelse(mod_recip$fitted.values > 0.5, 1, 0)
confusionMatrix(data = as.factor(prediction),
                reference = as.factor(recip_complete$survival_status),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 78 42
##          1 24 41
##
##                Accuracy : 0.6432
##                  95% CI : (0.5696, 0.7122)
##     No Information Rate : 0.5514
##     P-Value [Acc > NIR] : 0.006994
##
##                   Kappa : 0.264
##
##  Mcnemar's Test P-Value : 0.036389
##
##             Sensitivity : 0.4940
##             Specificity : 0.7647
##          Pos Pred Value : 0.6308
##          Neg Pred Value : 0.6500
##              Prevalence : 0.4486
##          Detection Rate : 0.2216
##    Detection Prevalence : 0.3514
##       Balanced Accuracy : 0.6293
##
##        'Positive' Class : 1
##
```

```
mod_disease <- glm(survival_status ~ disease + disease_group,
            family = binomial(), data = predictors)
summary(mod_disease)
```

```
##
## Call:
## glm(formula = survival_status ~ disease + disease_group, family = binomial(),
##     data = predictors)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1010  -1.0788  -0.9695   1.2793   1.4006
##
## Coefficients: (1 not defined because of singularities)
##                             Estimate Std. Error z value Pr(>|z|)
## (Intercept)                -0.23639    0.24423  -0.968    0.333
## diseaseAML                  0.05407    0.42646   0.127    0.899
## diseasechronic             -0.07727    0.38826  -0.199    0.842
## diseaselymphoma            16.80246  799.84828   0.021    0.983
## diseasenonmalignant        -0.27444    0.43930  -0.625    0.532
## disease_groupnonmalignant        NA         NA      NA       NA
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 257.69  on 186  degrees of freedom
## Residual deviance: 242.43  on 182  degrees of freedom
```

```
## AIC: 252.43
##
## Number of Fisher Scoring iterations: 15
```

Since disease_group and disease have overlap in the shared malignant level, we will select only one. Here we find an accuracy of 59.36% and no statistically significant predictor variables.

```
mod_disease <- glm(survival_status ~ disease ,
            family = binomial(), data = predictors)
summary(mod_disease)
```

```
##
## Call:
## glm(formula = survival_status ~ disease, family = binomial(),
##     data = predictors)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1010  -1.0788  -0.9695   1.2793   1.4006
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -0.23639    0.24423  -0.968    0.333
## diseaseAML            0.05407    0.42646   0.127    0.899
## diseasechronic       -0.07727    0.38826  -0.199    0.842
## diseaselymphoma      16.80246  799.84828   0.021    0.983
## diseasenonmalignant  -0.27444    0.43930  -0.625    0.532
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 257.69  on 186  degrees of freedom
## Residual deviance: 242.43  on 182  degrees of freedom
## AIC: 252.43
##
## Number of Fisher Scoring iterations: 15
```

```
prediction <- ifelse(mod_disease$fitted.values > 0.5, 1, 0)
confusionMatrix(data = as.factor(prediction),
                reference = as.factor(predictors$survival_status),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 102  76
##          1   0   9
##
##                Accuracy : 0.5936
##                  95% CI : (0.5195, 0.6647)
##     No Information Rate : 0.5455
##     P-Value [Acc > NIR] : 0.1057
```

```
##
##                   Kappa : 0.1144
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.10588
##             Specificity : 1.00000
##          Pos Pred Value : 1.00000
##          Neg Pred Value : 0.57303
##              Prevalence : 0.45455
##          Detection Rate : 0.04813
##    Detection Prevalence : 0.04813
##       Balanced Accuracy : 0.55294
##
##        'Positive' Class : 1
##
```

Variables related to the closeness of the match

Per our medical expert, not specifically domain expert, the suggestion is to not include HLA Match AND antigen/allele. In this case we will choose to only choose the variable HLA Match out of 10.

We see again here that our model is pretty ineffective as detecting survival status with an accuracy of 58.82%.

```
mod_match <- glm(survival_status ~ gender_match + ABO_match + CMV_status + HLA_match..out.of.10. +
            antigen + allele,
            family = binomial(), data = predictors)
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(mod_match)
```

```
##
## Call:
## glm(formula = survival_status ~ gender_match + ABO_match + CMV_status +
##     HLA_match..out.of.10. + antigen + allele, family = binomial(),
##     data = predictors)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
##  -8.49   -8.49    0.00    0.00    8.49
##
## Coefficients: (3 not defined because of singularities)
##                          Estimate Std. Error   z value Pr(>|z|)
## (Intercept)           -1.169e+16  7.547e+08 -15485999   <2e-16 ***
## gender_matchother     -4.629e+14  1.375e+07 -33657953   <2e-16 ***
## ABO_matchmatched      -3.273e+15  7.148e+07 -45782797   <2e-16 ***
## ABO_matchmismatched   -4.546e+15  7.047e+07 -64513924   <2e-16 ***
## CMV_status0           -2.993e+12  2.047e+07   -146206   <2e-16 ***
## CMV_status1           -4.574e+14  2.190e+07 -20883191   <2e-16 ***
```

```
## CMV_status2            1.074e+14  1.990e+07   5395713   <2e-16 ***
## CMV_status3            4.129e+14  2.070e+07  19950018   <2e-16 ***
## HLA_match..out.of.10.  1.265e+15  7.492e+07  16885679   <2e-16 ***
## antigen0               3.541e+15  6.871e+07  51529432   <2e-16 ***
## antigen1               7.844e+15  1.783e+08  43994258   <2e-16 ***
## antigen2               7.998e+15  2.441e+08  32764239   <2e-16 ***
## antigen3               1.020e+16  3.173e+08  32144169   <2e-16 ***
## allele0                       NA         NA        NA       NA
## allele1               -2.548e+15  1.612e+08 -15802759   <2e-16 ***
## allele2               -1.831e+15  9.090e+07 -20147579   <2e-16 ***
## allele3                       NA         NA        NA       NA
## allele4                       NA         NA        NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance:  257.69  on 186  degrees of freedom
## Residual deviance: 6199.51  on 172  degrees of freedom
## AIC: 6229.5
##
## Number of Fisher Scoring iterations: 25
```

```
p_hats <- mod_match$fitted.values
head(p_hats, 30)
```

```
##            1            2            3            4            5            6
## 1.000000e+00 1.000000e+00 1.000000e+00 2.220446e-16 1.000000e+00 2.220446e-16
##            7            8            9           10           11           12
## 2.220446e-16 2.220446e-16 2.220446e-16 1.000000e+00 1.000000e+00 1.000000e+00
##           13           14           15           16           17           18
## 2.220446e-16 2.220446e-16 1.000000e+00 1.000000e+00 1.000000e+00 2.220446e-16
##           19           20           21           22           23           24
## 2.220446e-16 2.220446e-16 1.000000e+00 1.000000e+00 1.000000e+00 1.000000e+00
##           25           26           27           28           29           30
## 1.000000e+00 2.220446e-16 1.000000e+00 1.000000e+00 2.220446e-16 1.000000e+00
```

```
mean(p_hats)
```

```
## [1] 0.6256684
```

```
# looks like something weird is happening with antigen (very bad predictor of survival)

# when we remove antigen, it looks pretty normal
# we can explore it without antigen or allele
mod_match <- glm(survival_status ~ gender_match + ABO_match + CMV_status + HLA_match..out.of.10.,
            family = binomial(), data = predictors)
summary(mod_match)
```

```
##
## Call:
## glm(formula = survival_status ~ gender_match + ABO_match + CMV_status +
```

```
##     HLA_match..out.of.10., family = binomial(), data = predictors)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5061  -1.0567  -0.8726   1.2093   1.5847
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)            16.0742   882.7454   0.018   0.9855
## gender_matchother      -0.0930     0.4135  -0.225   0.8221
## ABO_matchmatched      -13.7270   882.7437  -0.016   0.9876
## ABO_matchmismatched   -14.1991   882.7436  -0.016   0.9872
## CMV_status0            -0.7433     0.6183  -1.202   0.2293
## CMV_status1            -1.1942     0.6732  -1.774   0.0761 .
## CMV_status2            -0.7154     0.5999  -1.192   0.2331
## CMV_status3            -0.6612     0.6244  -1.059   0.2896
## HLA_match..out.of.10.  -0.1508     0.1910  -0.789   0.4298
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 257.69  on 186  degrees of freedom
## Residual deviance: 250.92  on 178  degrees of freedom
## AIC: 268.92
##
## Number of Fisher Scoring iterations: 13
```

```
prediction <- ifelse(mod_match$fitted.values > 0.5, 1, 0)
confusionMatrix(data = as.factor(prediction),
                reference = as.factor(predictors$survival_status),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 74 49
##          1 28 36
##
##                Accuracy : 0.5882
##                  95% CI : (0.5141, 0.6595)
##     No Information Rate : 0.5455
##     P-Value [Acc > NIR] : 0.13522
##
##                   Kappa : 0.1522
##
##  Mcnemar's Test P-Value : 0.02265
##
##             Sensitivity : 0.4235
##             Specificity : 0.7255
##          Pos Pred Value : 0.5625
##          Neg Pred Value : 0.6016
##              Prevalence : 0.4545
```

```
##          Detection Rate : 0.1925
##    Detection Prevalence : 0.3422
##       Balanced Accuracy : 0.5745
##
##        'Positive' Class : 1
##
```

Variables related to stem cell source

Again, we are seeing that are model is not doing to well in terms of accuracy of prediction. However, the model for stem cell source is doing well in terms of sensitivity, unfortunately this is offset by its specificity.

```
mod_source <- glm(survival_status ~ stem_cell_source,
            family = binomial(), data = predictors)
summary(mod_source)
```

```
##
## Call:
## glm(formula = survival_status ~ stem_cell_source, family = binomial(),
##     data = predictors)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.302  -1.045  -1.045   1.316   1.316
##
## Coefficients:
##                                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       0.2877     0.3118   0.923   0.3562
## stem_cell_sourceperipheral_blood  -0.6076     0.3543  -1.715   0.0863 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 257.69  on 186  degrees of freedom
## Residual deviance: 254.71  on 185  degrees of freedom
## AIC: 258.71
##
## Number of Fisher Scoring iterations: 4
```

```
prediction <- ifelse(mod_source$fitted.values > 0.5, 1, 0)

confusionMatrix(data = as.factor(prediction),
                reference = as.factor(predictors$survival_status),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 84 61
##          1 18 24
```

```
##
##                Accuracy : 0.5775
##                  95% CI : (0.5033, 0.6493)
##     No Information Rate : 0.5455
##     P-Value [Acc > NIR] : 0.2099
##
##                   Kappa : 0.1105
##
##  Mcnemar's Test P-Value : 2.297e-06
##
##             Sensitivity : 0.2824
##             Specificity : 0.8235
##          Pos Pred Value : 0.5714
##          Neg Pred Value : 0.5793
##              Prevalence : 0.4545
##          Detection Rate : 0.1283
##    Detection Prevalence : 0.2246
##       Balanced Accuracy : 0.5529
##
##        'Positive' Class : 1
##
```

This all lead me to wonder why our predictions are so subpar. Perhaps we should look at the prevalence of death?

We see that there are a reasonable amount who survived and died indicating there is not a substantial imbalance which we would need to consider.

```
bone %>%
  count(survival_status)
```

Below we will try again with a more "full" model now that we have parsed out some variables due to concerns of collinearity due to variables measuring the same or similar metrics. We remove variables included in recipient and donor that can be summarized by '. . . _match" variables.

Our model here has an accuracy of 65.41%

```
# attempt at a similar "full model"

#find complete cases only
cases_comp <- predictors %>%
  select(c(survival_status, recipient_age, recipient_gender, recipient_body_mass,
           recipient_ABO, recipient_rh, recipient_CMV, donor_age, stem_cell_source, disease, gender_mat
  drop_na()


full_mod <- glm(survival_status ~ donor_age  + stem_cell_source + recipient_age  + recipient_body_mass
                family = binomial(),
                data = cases_comp)

summary(full_mod)
```

```
##
## Call:
```

```
## glm(formula = survival_status ~ donor_age + stem_cell_source +
##     recipient_age + recipient_body_mass + as.factor(recipient_rh) +
##     disease + gender_match + ABO_match + CMV_status + stem_cell_source,
##     family = binomial(), data = cases_comp)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6611  -0.9506  -0.6656   1.1449   1.9932
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       1.680e+01  3.956e+03   0.004   0.9966
## donor_age                         8.174e-03  1.982e-02   0.412   0.6800
## stem_cell_sourceperipheral_blood -6.510e-01  3.907e-01  -1.666   0.0956 .
## recipient_age                    -7.952e-02  7.672e-02  -1.037   0.3000
## recipient_body_mass               4.236e-02  2.089e-02   2.028   0.0426 *
## as.factor(recipient_rh)minus     -1.879e+01  3.956e+03  -0.005   0.9962
## as.factor(recipient_rh)plus      -1.802e+01  3.956e+03  -0.005   0.9964
## diseaseAML                        2.823e-01  4.919e-01   0.574   0.5661
## diseasechronic                    1.848e-01  4.331e-01   0.427   0.6697
## diseaselymphoma                   1.795e+01  1.190e+03   0.015   0.9880
## diseasenonmalignant               1.089e-01  4.824e-01   0.226   0.8214
## gender_matchother                -1.567e-01  4.550e-01  -0.344   0.7306
## ABO_matchmatched                  1.288e+00  5.595e+03   0.000   0.9998
## ABO_matchmismatched               1.078e+00  5.595e+03   0.000   0.9998
## CMV_status0                      -6.847e-01  6.913e-01  -0.990   0.3219
## CMV_status1                      -1.131e+00  7.491e-01  -1.510   0.1310
## CMV_status2                      -6.518e-01  6.648e-01  -0.980   0.3268
## CMV_status3                      -5.189e-01  6.990e-01  -0.742   0.4579
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 254.51  on 184  degrees of freedom
## Residual deviance: 217.62  on 167  degrees of freedom
## AIC: 253.62
##
## Number of Fisher Scoring iterations: 16
```

```
prediction <- ifelse(full_mod$fitted.values > 0.5, 1, 0)
confusionMatrix(data = as.factor(prediction),
                reference = as.factor(cases_comp$survival_status),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 83 45
##          1 19 38
##
##                Accuracy : 0.6541
##                  95% CI : (0.5808, 0.7223)
```

```
##     No Information Rate : 0.5514
##     P-Value [Acc > NIR] : 0.002891
##
##                   Kappa : 0.2797
##
##  Mcnemar's Test P-Value : 0.001778
##
##             Sensitivity : 0.4578
##             Specificity : 0.8137
##          Pos Pred Value : 0.6667
##          Neg Pred Value : 0.6484
##              Prevalence : 0.4486
##          Detection Rate : 0.2054
##    Detection Prevalence : 0.3081
##       Balanced Accuracy : 0.6358
##
##        'Positive' Class : 1
##
```

We can try to use forward selection here:

```
library(broom)

cases_comp <- cases_comp %>%
  select(c(survival_status, recipient_age, recipient_body_mass,
           recipient_rh, disease, donor_age, stem_cell_source, disease, gender_match, ABO_match, CMV_sta

mod_basic <- glm(survival_status ~ 1, data=cases_comp, family = "binomial")
stepModel <- step(mod_basic, direction="forward",
                  scope=(~ donor_age  + stem_cell_source + recipient_age  + recipient_body_mass + as.fa
                  data=cases_comp)
```

```
## Start:  AIC=256.51
## survival_status ~ 1
##
##                            Df Deviance    AIC
## + recipient_body_mass       1   243.83 247.83
## + disease                   4   239.08 249.08
## + recipient_age             1   246.82 250.82
## + as.factor(recipient_rh)   2   248.44 254.44
## + stem_cell_source          1   251.21 255.21
## <none>                          254.51 256.51
## + donor_age                 1   253.89 257.89
## + ABO_match                 2   252.08 258.08
## + gender_match              1   254.45 258.45
## + CMV_status                4   250.99 260.99
##
## Step:  AIC=247.83
## survival_status ~ recipient_body_mass
##
##                            Df Deviance    AIC
## + disease                   4   230.45 242.45
## + as.factor(recipient_rh)   2   238.88 246.88
```

```
## + stem_cell_source         1   241.27 247.27
## <none>                         243.83 247.83
## + donor_age                1   243.62 249.62
## + recipient_age            1   243.68 249.68
## + gender_match             1   243.78 249.78
## + ABO_match                2   242.28 250.28
## + CMV_status               4   239.54 251.54
##
## Step:  AIC=242.45
## survival_status ~ recipient_body_mass + disease
##
##                           Df Deviance    AIC
## + as.factor(recipient_rh)  2   224.23 240.23
## + stem_cell_source         1   227.67 241.67
## <none>                         230.45 242.45
## + recipient_age            1   229.92 243.92
## + gender_match             1   230.14 244.14
## + donor_age                1   230.23 244.23
## + ABO_match                2   229.11 245.11
## + CMV_status               4   227.32 247.32
##
## Step:  AIC=240.23
## survival_status ~ recipient_body_mass + disease + as.factor(recipient_rh)
##
##                     Df Deviance    AIC
## + stem_cell_source   1   221.57 239.57
## <none>                   224.23 240.23
## + recipient_age      1   223.51 241.51
## + donor_age          1   224.13 242.13
## + gender_match       1   224.15 242.15
## + ABO_match          2   224.08 244.08
## + CMV_status         4   221.73 245.73
##
## Step:  AIC=239.57
## survival_status ~ recipient_body_mass + disease + as.factor(recipient_rh) +
##     stem_cell_source
##
##                   Df Deviance    AIC
## <none>                 221.57 239.57
## + recipient_age    1   220.54 240.54
## + donor_age        1   221.46 241.46
## + gender_match     1   221.54 241.54
## + ABO_match        2   221.25 243.25
## + CMV_status       4   219.37 245.37
```

```
tidy(stepModel)
```

Using forward selection, we are given the following model:

survival_status ~ recipient_body_mass + disease + as.factor(recipient_rh) + stem_cell_source With an AIC of 239.57

Next, we can try backwards selection:

```r
mod_back <- glm(survival_status ~ donor_age  + stem_cell_source + recipient_age  + recipient_body_mass +
                data = cases_comp,
                family = "binomial")

backStepModel <- step(mod_back,
                      direction = "backward",
                      data = cases_comp)
```

```
## Start:  AIC=253.62
## survival_status ~ donor_age + stem_cell_source + recipient_age +
##     recipient_body_mass + as.factor(recipient_rh) + disease +
##     gender_match + ABO_match + CMV_status + stem_cell_source
##
##                            Df Deviance    AIC
## - CMV_status                4   220.14 248.14
## - ABO_match                 2   217.92 249.92
## - gender_match              1   217.74 251.74
## - donor_age                 1   217.79 251.79
## - recipient_age             1   218.72 252.72
## <none>                          217.62 253.62
## - as.factor(recipient_rh)   2   221.84 253.84
## - stem_cell_source          1   220.41 254.41
## - recipient_body_mass       1   221.99 255.99
## - disease                   4   231.58 259.58
##
## Step:  AIC=248.14
## survival_status ~ donor_age + stem_cell_source + recipient_age +
##     recipient_body_mass + as.factor(recipient_rh) + disease +
##     gender_match + ABO_match
##
##                            Df Deviance    AIC
## - ABO_match                 2   220.39 244.39
## - gender_match              1   220.21 246.21
## - donor_age                 1   220.21 246.21
## - recipient_age             1   221.11 247.11
## <none>                          220.14 248.14
## - as.factor(recipient_rh)   2   224.74 248.74
## - stem_cell_source          1   223.20 249.20
## - recipient_body_mass       1   223.97 249.97
## - disease                   4   235.49 255.49
##
## Step:  AIC=244.39
## survival_status ~ donor_age + stem_cell_source + recipient_age +
##     recipient_body_mass + as.factor(recipient_rh) + disease +
##     gender_match
##
##                            Df Deviance    AIC
## - gender_match              1   220.44 242.44
## - donor_age                 1   220.49 242.49
## - recipient_age             1   221.44 243.44
## <none>                          220.39 244.39
## - stem_cell_source          1   223.32 245.32
## - as.factor(recipient_rh)   2   226.38 246.38
```

```
## - recipient_body_mass      1   224.43 246.43
## - disease                  4   236.16 252.16
##
## Step:  AIC=242.44
## survival_status ~ donor_age + stem_cell_source + recipient_age +
##     recipient_body_mass + as.factor(recipient_rh) + disease
##
##                            Df Deviance    AIC
## - donor_age                 1   220.54 240.54
## - recipient_age             1   221.46 241.46
## <none>                          220.44 242.44
## - stem_cell_source          1   223.43 243.43
## - recipient_body_mass       1   224.43 244.43
## - as.factor(recipient_rh)   2   226.64 244.64
## - disease                   4   236.20 250.20
##
## Step:  AIC=240.55
## survival_status ~ stem_cell_source + recipient_age + recipient_body_mass +
##     as.factor(recipient_rh) + disease
##
##                            Df Deviance    AIC
## - recipient_age             1   221.57 239.57
## <none>                          220.54 240.54
## - stem_cell_source          1   223.51 241.51
## - recipient_body_mass       1   224.62 242.62
## - as.factor(recipient_rh)   2   226.94 242.94
## - disease                   4   236.28 248.28
##
## Step:  AIC=239.57
## survival_status ~ stem_cell_source + recipient_body_mass + as.factor(recipient_rh) +
##     disease
##
##                            Df Deviance    AIC
## <none>                          221.57 239.57
## - stem_cell_source          1   224.23 240.23
## - as.factor(recipient_rh)   2   227.67 241.67
## - recipient_body_mass       1   228.21 244.21
## - disease                   4   236.52 246.52
```

```
tidy(backStepModel)
```

With this model selection method, we are given the following model:

survival_status ~ stem_cell_source + recipient_body_mass + as.factor(recipient_rh) + disease With an AIC value of 239.57

So, whether we use the forward or backwards selection process, we are given the same model.

Summary statistics of this model are given below:

Here we see that this model gives an overall accuracy of 66% (getting a little better) with sensitivity of 45.7% and specificity of 82.4%. Still, we are not classifying deaths with a very high success rate (yet!)

```
mod.result <- glm(survival_status ~ stem_cell_source + recipient_body_mass + as.factor(recipient_rh) + d
                  data = cases_comp,
```

```
                 family = binomial())
summary(mod.result)
```

```
##
## Call:
## glm(formula = survival_status ~ stem_cell_source + recipient_body_mass +
##     as.factor(recipient_rh) + disease, family = binomial(), data = cases_comp)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5498  -0.9842  -0.6701   1.1515   1.9073
##
## Coefficients:
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       1.719e+01  2.759e+03   0.006   0.9950
## stem_cell_sourceperipheral_blood -6.179e-01  3.800e-01  -1.626   0.1039
## recipient_body_mass               2.182e-02  8.635e-03   2.527   0.0115 *
## as.factor(recipient_rh)minus     -1.863e+01  2.759e+03  -0.007   0.9946
## as.factor(recipient_rh)plus      -1.783e+01  2.759e+03  -0.006   0.9948
## diseaseAML                        1.701e-01  4.594e-01   0.370   0.7112
## diseasechronic                    1.325e-01  4.138e-01   0.320   0.7487
## diseaselymphoma                   1.795e+01  1.229e+03   0.015   0.9884
## diseasenonmalignant               1.102e-01  4.662e-01   0.236   0.8131
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 254.51  on 184  degrees of freedom
## Residual deviance: 221.57  on 176  degrees of freedom
## AIC: 239.57
##
## Number of Fisher Scoring iterations: 16
```

```
prediction <- ifelse(mod.result$fitted.values > 0.45, 1, 0) # Changed the cutoff to get better results
```

```
confusionMatrix(data = as.factor(prediction),
                reference = as.factor(cases_comp$survival_status),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 69 37
##          1 33 46
##
##                Accuracy : 0.6216
##                  95% CI : (0.5475, 0.6917)
##     No Information Rate : 0.5514
##     P-Value [Acc > NIR] : 0.03172
```

```
##
##                   Kappa : 0.2317
##
##  Mcnemar's Test P-Value : 0.71992
##
##             Sensitivity : 0.5542
##             Specificity : 0.6765
##          Pos Pred Value : 0.5823
##          Neg Pred Value : 0.6509
##              Prevalence : 0.4486
##          Detection Rate : 0.2486
##    Detection Prevalence : 0.4270
##       Balanced Accuracy : 0.6153
##
##        'Positive' Class : 1
##
```

Are biggest issue seems to be our sensitivity, or having our model correctly predict a case of death. Let's see where the max possible sensitivity could be based on our current predictive model based on an ROC curve and the area under the curve.

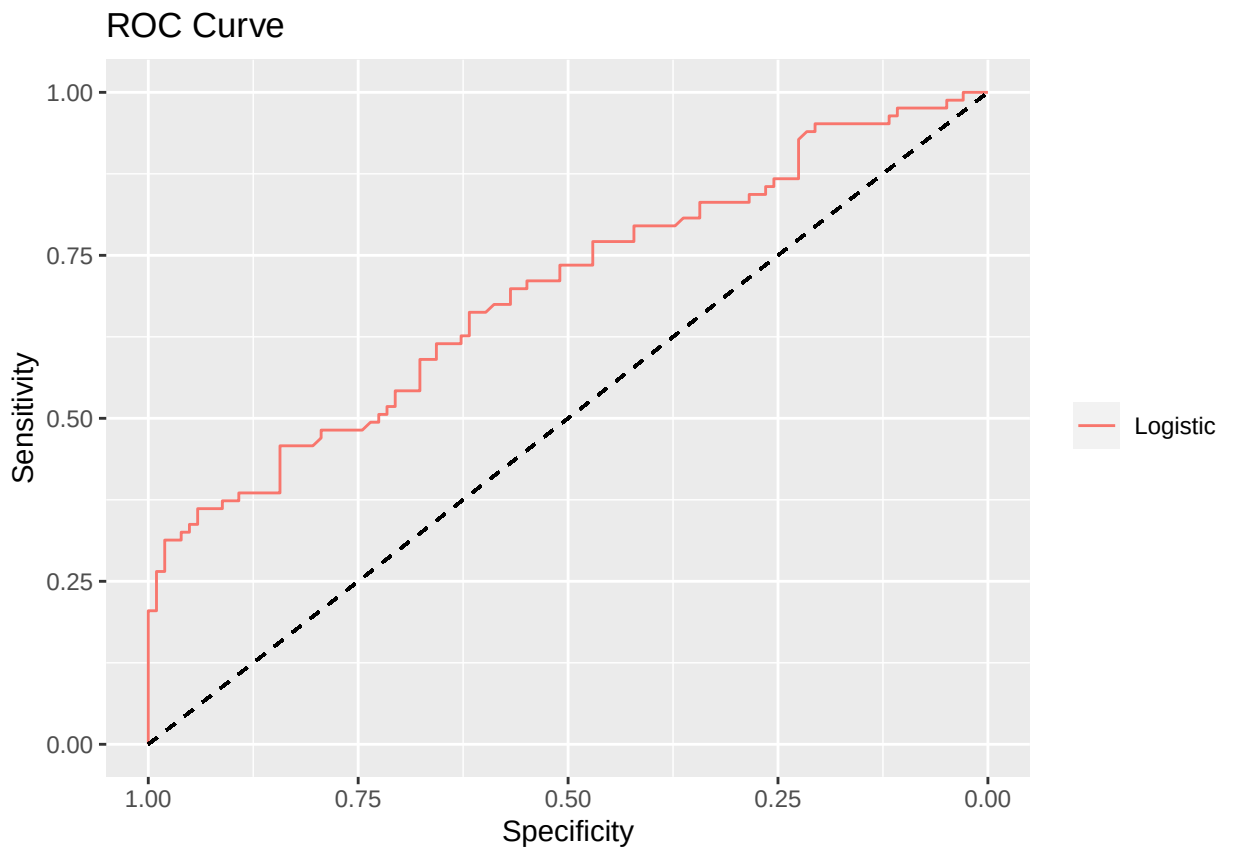
```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
roc_logit <- roc(cases_comp$survival_status, mod.result$fitted.values)
```

```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases
```

```
ggroc(list("Logistic" = roc_logit)) +
  theme(legend.title = element_blank()) +
  geom_segment(aes(x = 1, xend = 0, y = 0, yend = 1), color = "black", linetype = "dashed") +
  xlab("Specificity") +
  ylab("Sensitivity") +
  ggtitle("ROC Curve")
```

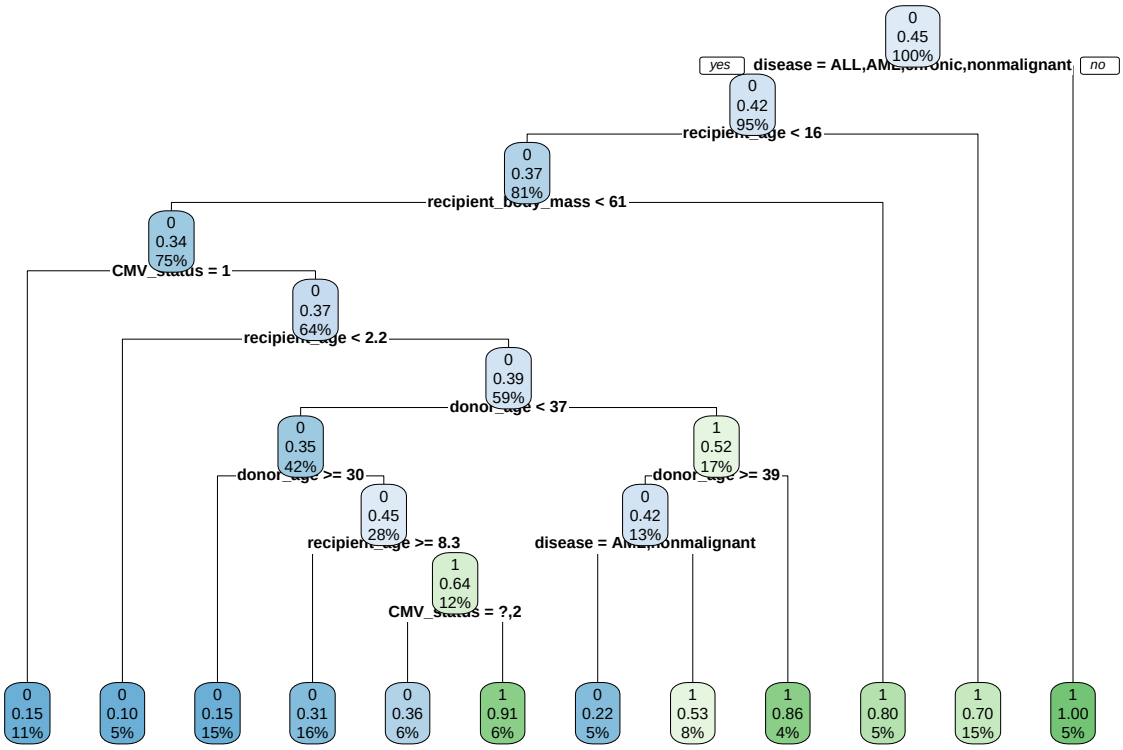



```
auc(roc_logit) # Want this to be >0.7
```

```
## Area under the curve: 0.6926
```

Next we can try decision trees. We see here that the predictive accuracy is much higher now! We have an accuracy of 77.3% and a balanced sensitivity and specificity.

```
cases_comp <- cases_comp %>%
  select(c(survival_status, recipient_age, recipient_body_mass,
           recipient_rh, disease, donor_age, stem_cell_source, disease, gender_match, ABO_match, CMV_sta



# will fit the tree
fit <- rpart(as.factor(survival_status) ~ ., data = cases_comp)

# will plot the tree
rpart.plot(fit, cex = 0.5)
```




```
# create new dataframe without survival outcomes
df <- subset(cases_comp, select = -c(survival_status))

#make predicitons and cutoffs
mypreds <- predict(fit, df)
predictions <- ifelse(mypreds[,2] > 0.5, 1, 0)


confusionMatrix(data = as.factor(predictions),
                reference = as.factor(cases_comp$survival_status),
                positive = "1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 83 23
##          1 19 60
##
##                Accuracy : 0.773
##                  95% CI : (0.7058, 0.8312)
##     No Information Rate : 0.5514
##     P-Value [Acc > NIR] : 2.984e-10
##
##                   Kappa : 0.539
##
```

```
##  Mcnemar's Test P-Value : 0.6434
##
##             Sensitivity : 0.7229
##             Specificity : 0.8137
##          Pos Pred Value : 0.7595
##          Neg Pred Value : 0.7830
##              Prevalence : 0.4486
##          Detection Rate : 0.3243
##    Detection Prevalence : 0.4270
##       Balanced Accuracy : 0.7683
##
##        'Positive' Class : 1
##
```

Secondary Analysis

We can also look at some secondary analyses. We read a journal article that found that patients who received peripheral stem cell transplants were more likely to develop graft-versus-host disease (GVHD). We can easily check to see if this holds true in our data.

```
# more people who received peripheral stem cells survived, meaning that they were
# more likely to have time to develop GVHD
table(bone$survival_status, bone$stem_cell_source) %>%
  prop.table(margin = 2)
```

```
##
##     bone_marrow peripheral_blood
##   0   0.4285714        0.5793103
##   1   0.5714286        0.4206897
```

```
# proportion with GVHD
table(bone$acute_GvHD_II_III_IV, bone$stem_cell_source) %>%
  prop.table(margin = 2)
```

```
##
##       bone_marrow peripheral_blood
##   no    0.4523810        0.3862069
##   yes   0.5476190        0.6137931
```

```
# what if we only look at people who survived?
survived <- filter(bone, survival_status == 0)
table(survived$acute_GvHD_II_III_IV, survived$stem_cell_source) %>%
  prop.table(margin = 2)
```

```
##
##       bone_marrow peripheral_blood
##   no    0.5000000        0.4166667
##   yes   0.5000000        0.5833333
```

```
# note: there are only a few people included in this table

# could try to model GVHD based on stem cell source, adjusted for potential confounders
```