Logistic models

Code

2019-01-09

It made sense to transfer the logistic analysis to here since it's mostly in R.

inattention, volume

Hide

Hide

```
library(gdata)
library(nnet)
load('~/data/baseline_prediction/combined_descriptives_12172018.RData.gz')
clin = read.csv('~/data/baseline_prediction/long_clin_11302018.csv')
df = merge(clin, data, by='MRN')
idx = df$diag_group != 'new_onset'
idx2 = !is.na(df$inatt_vol_lh) | !is.na(df$inatt_AD_clu1) | !is.na(df$inatt_melodic_D
MN)
imaging = df[idx & idx2,]
imaging[imaging$OLS_inatt_slope <= -.33, 'OLS_inatt_categ'] = 'marked'
imaging[imaging$OLS_inatt_slope > -.33 & imaging$OLS_inatt_slope <= 0, 'OLS_inatt_cat
eg'] = 'mild'
imaging[imaging$OLS_inatt_slope > 0, 'OLS_inatt_categ'] = 'deter'
imaging[imaging$DX == 'NV', 'OLS_inatt_categ'] = 'NV'
imaging$OLS_inatt_categ = as.factor(imaging$OLS_inatt_categ)
imaging$OLS_inatt_categ = relevel(imaging$OLS_inatt_categ, ref='NV')
load('~/data/baseline_prediction/combined_descriptives_12172018.RData.gz')
clin = read.csv('~/data/baseline_prediction/long_clin_11302018.csv')
df = merge(clin, data, by='MRN')
idx = df$diag_group != 'new_onset'
struct = df[!is.na(df$HI_vol_rh) & idx,]
load('~/data/baseline_prediction/struct_volume_11142018_260timeDiff12mo.RData.gz')
struct = merge(struct, data, by='MRN') # put mask ids in combined dataset
mprage = read.xls('~/data/baseline_prediction/long_scans_08072018.xlsx',
                  sheet='mprage')
struct = merge(struct, mprage, by.x='mask.id', by.y='Mask.ID...Scan') # get demograph
ics
qc = read.csv('~/data/baseline_prediction/master_qc.csv')
struct = merge(struct, qc, by.x='mask.id', by.y='Mask.ID') # get QC scores
df = merge(struct, imaging, by='MRN')
dim(df)
```

```
[1]  241 5244
```

First, let's check if any of the variables we used as covariates before has a relationship with the categories:

Hide

Hide

```
for (t in c('age_at_scan', 'I(age_at_scan^2)', 'ext_avg_freesurfer5.3', 'int_avg_free
surfer5.3', 'mprage_QC', 'as.numeric(Sex...Subjects)')) {
  fm_str = sprintf('%s ~ OLS_inatt_categ', t)
  print(fm_str)
  print(summary(aov(lm(as.formula(fm_str), data=df))))
}
```

```
[1] "age_at_scan ~ OLS_inatt_categ"
                 Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ   3   36.5  12.167   2.392 0.0693 .
Residuals       237 1205.5   5.087
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "I(age_at_scan^2) ~ OLS_inatt_categ"
                 Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ   3  13843    4614   2.328 0.0752 .
Residuals       237 469671    1982
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "ext_avg_freesurfer5.3 ~ OLS_inatt_categ"
                 Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ   3  0.498   0.166   1.395  0.245
Residuals       237 28.205   0.119
[1] "int_avg_freesurfer5.3 ~ OLS_inatt_categ"
                 Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ   3  1.214  0.4048   3.303  0.021 *
Residuals       237 29.050  0.1226
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "mprage_QC ~ OLS_inatt_categ"
                 Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ   3   0.41  0.1382   0.672   0.57
Residuals       237  48.73  0.2056
[1] "as.numeric(Sex...Subjects) ~ OLS_inatt_categ"
                 Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ   3   1.93  0.6425   2.976 0.0323 *
Residuals       237  51.18  0.2159
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

So, we should probably include the variables that have some relationship to the inattention categories:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_vol_lh.x) + int_avg_freesurfer5.3 + Se
x...Subjects, data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 334.096941
iter  10 value 301.875869
final  value 298.896079
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
        (Intercept) scale(inatt_vol_lh.x) int_avg_freesurfer5.3 Sex...SubjectsMale
deter  3.981309e-05            0.03627413           0.003261793       0.0009316589
marked 1.685679e-02            0.02457266           0.126635988       0.4863639387
mild   3.849646e-03            0.84629240           0.082773587       0.0964661215
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_vol_lh.x) +
    int_avg_freesurfer5.3 + Sex...Subjects, data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_vol_lh.x) int_avg_freesurfer5.3 Sex...SubjectsMale
deter    -4.382657           -0.41982308             1.5212826          1.3446484
marked   -2.437644            0.40816544             0.7875303          0.2736658
mild     -3.334581           -0.04138496             0.9946383          0.7358081

Residual Deviance: 597.7922
AIC: 621.7922
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_vol_lh.x) int_avg_freesurfer5.3 Sex...SubjectsMale
deter   0.01249213             0.6571631              4.578093          3.836837
marked  0.08736645             1.5040560              2.197961          1.314775
mild    0.03562951             0.9594597              2.703746          2.087168
```

OK, so these make sense. If we assume all Freesurfer QC is good, what does it look like?

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_vol_lh.x) + Sex...Subjects, data = df,
na.action=na.omit)
```

```
# weights:  16 (9 variable)
initial  value 334.096941
iter  10 value 305.628140
final  value 303.887882
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
        (Intercept) scale(inatt_vol_lh.x) Sex...SubjectsMale
deter  2.044054e-05            0.01752835        0.001227767
marked 1.573282e-03            0.03542537        0.543345916
mild   5.089723e-05            0.71917032        0.115643570
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_vol_lh.x) +
    Sex...Subjects, data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_vol_lh.x) Sex...SubjectsMale
deter   -1.4822006           -0.46517638          1.2901887
marked  -0.9644105            0.37741879          0.2373413
mild    -1.4628664           -0.07584044          0.6910465

Residual Deviance: 607.7758
AIC: 625.7758
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_vol_lh.x) Sex...SubjectsMale
deter    0.2271373            0.6280243           3.633472
marked   0.3812079            1.4585150           1.267874
mild     0.2315716            0.9269641           1.995803
```

Not as good. Alright, let's interpret the better model then:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_vol_lh.x) + int_avg_freesurfer5.3 + Se
x...Subjects, data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 334.096941
iter  10 value 301.875869
final  value 298.896079
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
        (Intercept) scale(inatt_vol_lh.x) int_avg_freesurfer5.3 Sex...SubjectsMale
deter  3.981309e-05           0.03627413           0.003261793       0.0009316589
marked 1.685679e-02           0.02457266           0.126635988       0.4863639387
mild   3.849646e-03           0.84629240           0.082773587       0.0964661215
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_vol_lh.x) +
    int_avg_freesurfer5.3 + Sex...Subjects, data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_vol_lh.x) int_avg_freesurfer5.3 Sex...SubjectsMale
deter    -4.382657           -0.41982308             1.5212826          1.3446484
marked   -2.437644            0.40816544             0.7875303          0.2736658
mild     -3.334581           -0.04138496             0.9946383          0.7358081

Residual Deviance: 597.7922
AIC: 621.7922
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_vol_lh.x) int_avg_freesurfer5.3 Sex...SubjectsMale
deter   0.01249213             0.6571631              4.578093           3.836837
marked  0.08736645             1.5040560              2.197961           1.314775
mild    0.03562951             0.9594597              2.703746           2.087168
```

Let's just worry about the two significant categories, deterioration and marked improvement:

- A one-unit increase in the volume of the left hemisphere cluster variable (i.e. increase by 1 SD) is associated with a decrease in the log odds of deteriorating (vs normals) in the amount of .42. That one-unit increase is also associated with a .41 increase in the log odds of a marked improvement.
- In terms of relative risk ratio, a one-unit decrease in the the volume of that brain cluster yields a relative risk ratio of .66 of deterioration (vs normals). The relative risk ratio for a one-unit increase is 1.50 for marked improvement. In other words, the odds of marked improvement, compared to normals, is 1.5 times higher for every 1 SD we increase in that brain region.

Let's make a simpler (but not too far off) model, and look at probabilities:

Hide

Hide

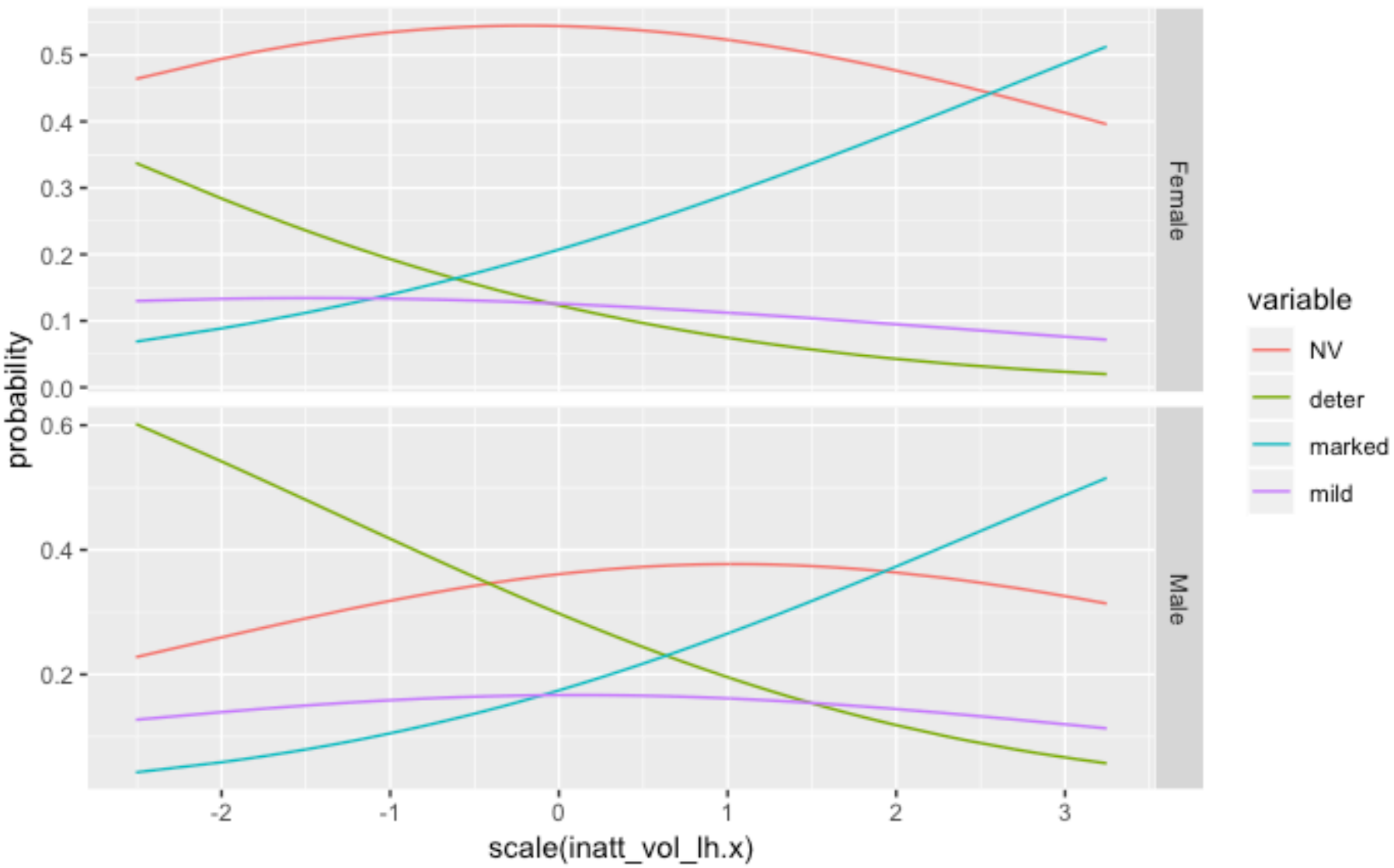
```
library(reshape2)
library(ggplot2)
fit <- multinom(OLS_inatt_categ ~ scale(inatt_vol_lh.x) + Sex...Subjects, data = df,
na.action=na.omit)
```

```
# weights:  16 (9 variable)
initial  value 334.096941
iter  10 value 305.628140
final  value 303.887882
converged
```

Hide

Hide

```
z <- summary(fit)$coefficients/summary(fit)$standard.errors
p <- (1 - pnorm(abs(z), 0, 1)) * 2
dbrain = data.frame(Sex...Subjects=rep(c('Male', 'Female'), length(df$inatt_vol_lh.x)
), inatt_vol_lh.x=rep(df$inatt_vol_lh.x, 2))
pp.dbrain = cbind(dbrain, predict(fit, newdata = dbrain, type='probs', se=T))
lpp = melt(pp.dbrain, id.vars=c('Sex...Subjects', 'inatt_vol_lh.x'), value.name='prob
ability')
ggplot(lpp, aes(x = scale(inatt_vol_lh.x), y = probability, colour = variable)) + geo
m_line() + facet_grid(Sex...Subjects ~
  ., scales = "free")
```



Hide

Hide

```
print(p)
```

```
        (Intercept) scale(inatt_vol_lh.x) Sex...SubjectsMale
deter  2.044054e-05            0.01752835        0.001227767
marked 1.573282e-03            0.03542537        0.543345916
mild   5.089723e-05            0.71917032        0.115643570
```

Hide

Hide

```
print(fit$AIC)
```

```
[1] 625.7758
```

Still, we'll eventually have to compare all models, even though some have more subjects than others. As we're not doing this in a cross-validation framework, the next best thing is to check how well we can predict our training set. If we're doing too well, there's a high risk of overfitting. But we need to be doing somewhat well, to show some evidence of modeling the data correctly. So, let's do that with our best model:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_vol_lh.x) + int_avg_freesurfer5.3 + Se
x...Subjects, data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 334.096941
iter  10 value 301.875869
final  value 298.896079
converged
```

Hide

Hide

```
res.roc = multiclass.roc(df$OLS_inatt_categ, as.numeric(predict(fit1, newdata=df, typ
e='class')))
print(sprintf('AUC: %f', auc(res.roc)))
```

```
[1] "AUC: 0.513025"
```

HI, volume

Hide

Hide

```
imaging$OLS_HI_categ = NULL
imaging[imaging$OLS_HI_slope <= -.5, 'OLS_HI_categ'] = 'marked'
imaging[imaging$OLS_HI_slope > -.5 & imaging$OLS_HI_slope <= 0, 'OLS_HI_categ'] = 'mi
ld'
imaging[imaging$OLS_HI_slope > 0, 'OLS_HI_categ'] = 'deter'
imaging[imaging$DX == 'NV', 'OLS_HI_categ'] = 'NV'
imaging$OLS_HI_categ = as.factor(imaging$OLS_HI_categ)
imaging$OLS_HI_categ = relevel(imaging$OLS_HI_categ, ref='NV')
df = merge(struct, imaging, by='MRN')
for (t in c('age_at_scan', 'I(age_at_scan^2)', 'ext_avg_freesurfer5.3', 'int_avg_free
surfer5.3', 'mprage_QC', 'as.numeric(Sex...Subjects)')) {
  fm_str = sprintf('%s ~ OLS_HI_categ', t)
  print(fm_str)
  print(summary(aov(lm(as.formula(fm_str), data=df))))
}
```

```
[1] "age_at_scan ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   12.1   4.029   0.776  0.508
Residuals    237 1230.0   5.190
[1] "I(age_at_scan^2) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   6238    2080   1.033  0.379
Residuals    237 477276    2014
[1] "ext_avg_freesurfer5.3 ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3  0.446  0.1486   1.246  0.294
Residuals    237 28.257  0.1192
[1] "int_avg_freesurfer5.3 ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3  1.317  0.4389   3.593 0.0143 *
Residuals    237 28.947  0.1221
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "mprage_QC ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   1.04  0.3462   1.706  0.167
Residuals    237  48.11  0.2030
[1] "as.numeric(Sex...Subjects) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   1.66  0.5542   2.553 0.0562 .
Residuals    237  51.44  0.2171
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Internal Freesurfer popped up again, but Sex is not as impressive. Let’s try it both ways to see what happens:

Hide

Hide

```
fit1 <- multinom(OLS_HI_categ ~ scale(HI_vol_rh.x) + Sex...Subjects + int_avg_freesur
fer5.3, data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 334.096941
iter  10 value 290.989670
final  value 288.994497
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
fit2 <- multinom(OLS_HI_categ ~ scale(HI_vol_rh.x) + int_avg_freesurfer5.3, data = df
, na.action=na.omit)
```

```
# weights:  16 (9 variable)
initial  value 334.096941
iter  10 value 293.649403
final  value 293.597393
converged
```

Hide

Hide

```
z2 <- summary(fit2)$coefficients/summary(fit2)$standard.errors
p2 <- (1 - pnorm(abs(z2), 0, 1)) * 2
rr2 = exp(coef(fit2))
pp2 = fitted(fit2)
print(p1)
```

```
        (Intercept) scale(HI_vol_rh.x) Sex...SubjectsMale int_avg_freesurfer5.3
deter  3.712725e-03        0.006498882         0.41154165           0.082029393
marked 1.230771e-02        0.022285877         0.01244109           0.109228896
mild   2.062869e-05        0.875740854         0.02062377           0.001377105
```

Hide

Hide

```
print(p2)
```

```
        (Intercept) scale(HI_vol_rh.x) int_avg_freesurfer5.3
deter  0.0059996209        0.003244911           0.093054454
marked 0.0663592914        0.053784816           0.148893703
mild   0.0001423472        0.520253260           0.001699101
```

Hide

Hide

```
print(fit1$AIC)
```

```
[1] 601.989
```

Hide

Hide

```
print(fit2$AIC)
```

```
[1] 605.1948
```

Using Sex is definitely better, and it's actuall better as it is more similar to the inattention model:

Hide

Hide

```
fit <- multinom(OLS_HI_categ ~ scale(HI_vol_rh.x) + Sex...Subjects + int_avg_freesurf
er5.3, data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 334.096941
iter  10 value 290.989670
final  value 288.994497
converged
```

Hide

Hide

```
z <- summary(fit)$coefficients/summary(fit)$standard.errors
p <- (1 - pnorm(abs(z), 0, 1)) * 2
print(p)
```

```
        (Intercept) scale(HI_vol_rh.x) Sex...SubjectsMale int_avg_freesurfer5.3
deter  3.712725e-03        0.006498882         0.41154165           0.082029393
marked 1.230771e-02        0.022285877         0.01244109           0.109228896
mild   2.062869e-05        0.875740854         0.02062377           0.001377105
```

Hide

Hide

```
print(fit$AIC)
```

```
[1] 601.989
```

So, what does it mean?

Hide

Hide

```
rr = exp(coef(fit))
print('p-values')
```

```
[1] "p-values"
```

Hide

Hide

```
print(p)
```

```
        (Intercept) scale(HI_vol_rh.x) Sex...SubjectsMale int_avg_freesurfer5.3
deter  3.712725e-03        0.006498882         0.41154165           0.082029393
marked 1.230771e-02        0.022285877         0.01244109           0.109228896
mild   2.062869e-05        0.875740854         0.02062377           0.001377105
```

Hide

Hide

```
print(fit)
```

```
Call:
multinom(formula = OLS_HI_categ ~ scale(HI_vol_rh.x) + Sex...Subjects +
    int_avg_freesurfer5.3, data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(HI_vol_rh.x) Sex...SubjectsMale int_avg_freesurfer5.3
deter    -3.772418         0.54726305          0.3986516             1.1309769
marked   -2.300565        -0.43837027          0.8649990             0.7403711
mild     -5.095517         0.03253334          1.0324728             1.8448801

Residual Deviance: 577.989
AIC: 601.989
```

Hide

Hide

```
print('risk ratio')
```

```
[1] "risk ratio"
```

Hide

Hide

```
print(rr)
```

```
       (Intercept) scale(HI_vol_rh.x) Sex...SubjectsMale int_avg_freesurfer5.3
deter   0.02299639          1.7285157           1.489814             3.098682
marked  0.10020218          0.6450869           2.375004             2.096713
mild    0.00612414          1.0330683           2.808001             6.327341
```

Again, we focus on the two significant categories, deterioration and marked improvement:

- A one-unit increase in the volume of the right hemisphere cluster variable (i.e. increase by 1 SD) is associated with an increase in the log odds of deteriorating (vs normals) in the amount of .55. That one-unit increase is also associated with a .44 decrease in the log odds of a marked improvement.
- In terms of relative risk ratio, a one-unit increase in the volume of that brain cluster yields a relative risk ratio of 1.73 of deterioration (vs normals). The relative risk ratio for a one-unit decrease is .65 for marked improvement. In other words, the odds of deterioration, compared to normals, is 1.74 times higher for every 1 SD we increase in that brain region; conversely, the odds of marked improvement is .65 lower for every 1 SD we increase in that brain region.

Let’s make a much simpler (but not too far off) model again, and look at probabilities:

Hide

Hide

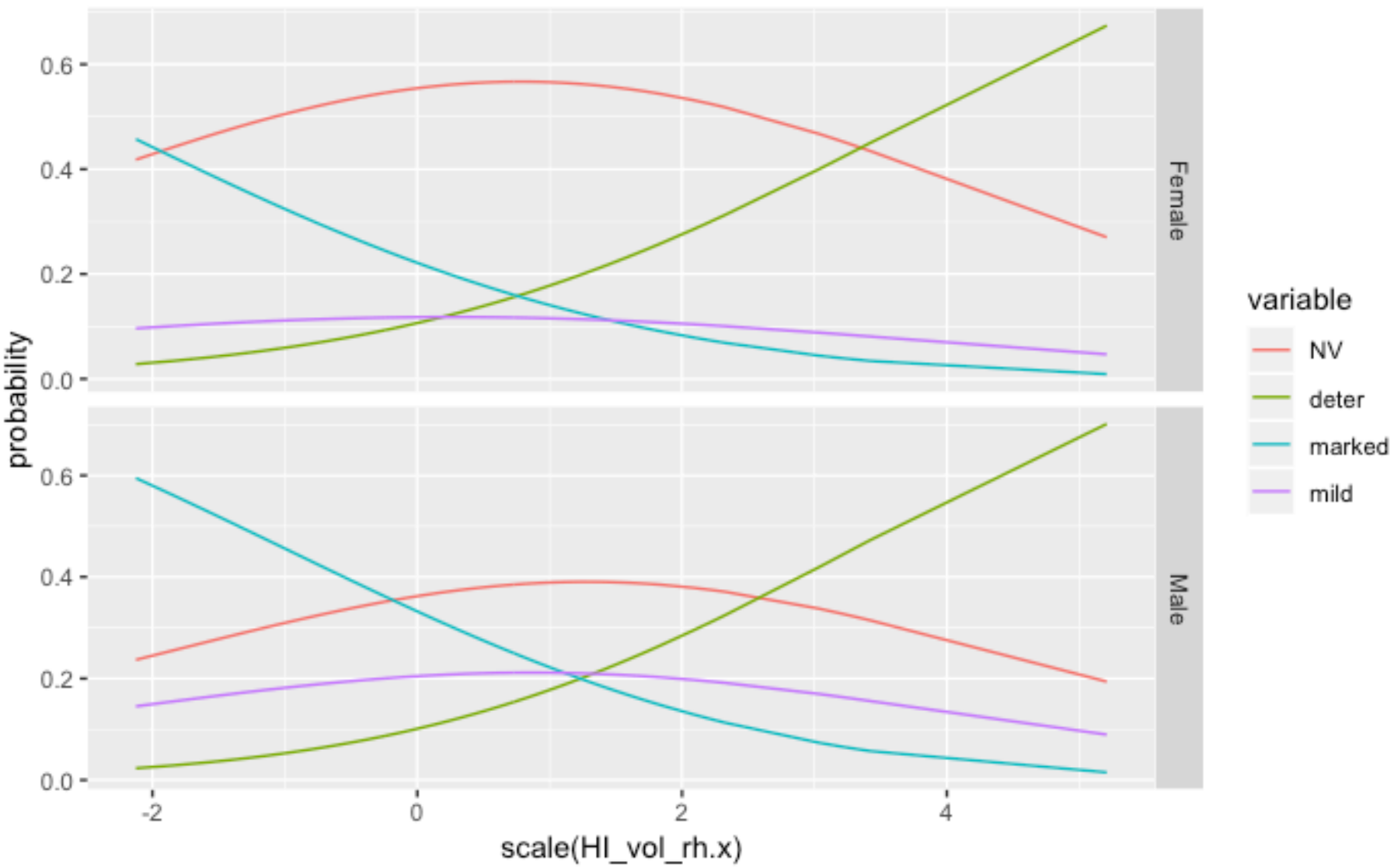
```
library(reshape2)
library(ggplot2)
fit <- multinom(OLS_HI_categ ~ scale(HI_vol_rh.x) + Sex...Subjects, data = df, na.act
ion=na.omit)
```

```
# weights:  16 (9 variable)
initial  value 334.096941
iter  10 value 295.589314
final  value 294.983811
converged
```

Hide

Hide

```
z <- summary(fit)$coefficients/summary(fit)$standard.errors
p <- (1 - pnorm(abs(z), 0, 1)) * 2
dbrain = data.frame(Sex...Subjects=rep(c('Male', 'Female'), length(df$HI_vol_rh.x)),
HI_vol_rh.x=rep(df$HI_vol_rh.x, 2))
pp.dbrain = cbind(dbrain, predict(fit, newdata = dbrain, type='probs', se=T))
lpp = melt(pp.dbrain, id.vars=c('Sex...Subjects', 'HI_vol_rh.x'), value.name='probabi
lity')
ggplot(lpp, aes(x = scale(HI_vol_rh.x), y = probability, colour = variable)) + geom_l
ine() + facet_grid(Sex...Subjects ~
    ., scales = "free")
```



Hide

Hide

```
print(p)
```

```
        (Intercept) scale(HI_vol_rh.x) Sex...SubjectsMale
deter  2.568256e-05         0.01319032         0.44013566
marked 1.351576e-03         0.01267609         0.01540593
mild   2.890596e-05         0.84717628         0.02477473
```

Hide

Hide

```
print(fit$AIC)
```

```
[1] 607.9676
```

And for future comparisons, this is the model AUC on training data:

Hide

Hide

```
fit1 <- multinom(OLS_HI_categ ~ scale(HI_vol_rh.x) + Sex...Subjects + int_avg_freesur
fer5.3, data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 334.096941
iter  10 value 290.989670
final  value 288.994497
converged
```

Hide

Hide

```
res.roc = multiclass.roc(df$OLS_inatt_categ, as.numeric(predict(fit1, newdata=df, typ
e='class')))
print(sprintf('AUC: %f', auc(res.roc)))
```

```
[1] "AUC: 0.509377"
```

inattention, DTI

Hide

Hide

```
load('~/data/baseline_prediction/combined_descriptives_12172018.RData.gz')
clin = read.csv('~/data/baseline_prediction/long_clin_11302018.csv')
df = merge(clin, data, by='MRN')
idx = df$diag_group != 'new_onset'
dti = df[!is.na(df$inatt_AD_clu1) & idx,]
load('~/data/baseline_prediction/dti_rd_voxelwise_n272_09212018.RData.gz')
dti = merge(dti, data, by='MRN') # put mask ids in combined dataset
mprage = read.xls('~/data/baseline_prediction/long_scans_08072018.xlsx',
                  sheet='dti')
dti = merge(dti, mprage, by.x='mask.id', by.y='Mask.ID') # get demographics
qc = read.csv('~/data/baseline_prediction/master_qc.csv')
dti = merge(dti, qc, by.x='mask.id', by.y='Mask.ID') # get QC scores
df = merge(dti, imaging, by='MRN')
df$mvmt = rowMeans(scale(df$norm.trans), scale(df$norm.rot))
dim(df)
```

```
[1]   253 12111
```

Hide

Hide

```
for (t in c('age_at_scan', 'I(age_at_scan^2)', 'mvmt', 'I(mvmt^2)', 'as.numeric(Sex)'
)) {
  fm_str = sprintf('%s ~ OLS_inatt_categ', t)
  print(fm_str)
  print(summary(aov(lm(as.formula(fm_str), data=df))))
}
```

```
[1] "age_at_scan ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   49.4  16.453   3.074 0.0283 *
Residuals      249 1332.8   5.353
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "I(age_at_scan^2) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3  19177    6392   3.128 0.0264 *
Residuals      249 508812    2043
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "mvmt ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   3.13  1.0425   1.043  0.374
Residuals      249 248.87  0.9995
[1] "I(mvmt^2) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   16.1   5.356   1.564  0.199
Residuals      249  852.7   3.424
[1] "as.numeric(Sex) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   1.39  0.4629   2.133 0.0966 .
Residuals      249  54.03  0.2170
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

So, it looks like for this only the two age terms are significant (not even sex). So:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_AD_clu1.x) + scale(inatt_AD_clu2.x) +
age_at_scan + I(age_at_scan^2), data = df, na.action=na.omit)
```

```
# weights:  24 (15 variable)
initial  value 350.732473
iter  10 value 326.394548
iter  20 value 319.650278
final  value 319.643551
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
       (Intercept) scale(inatt_AD_clu1.x) scale(inatt_AD_clu2.x) age_at_scan I(age_at
_scan^2)
deter   0.39022534             0.0303269             0.3341888   0.2712942       0.
11959343
marked  0.08958495             0.5073408             0.4402750   0.1453827       0.
14510595
mild    0.03702642             0.3943330             0.4553216   0.0694384       0.
06435903
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_AD_clu1.x) +
    scale(inatt_AD_clu2.x) + age_at_scan + I(age_at_scan^2),
    data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_AD_clu1.x) scale(inatt_AD_clu2.x) age_at_scan I(age_at
_scan^2)
deter    -1.896797            -0.3821876            -0.1716772   0.5582614      -0.
04417232
marked   -3.791111             0.1213238             0.1256194   0.6886507      -0.
03519388
mild     -6.066505            -0.1746022            -0.1549381   1.1112369      -0.
05780350

Residual Deviance: 639.2871
AIC: 669.2871
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_AD_clu1.x) scale(inatt_AD_clu2.x) age_at_scan I(age_at
_scan^2)
deter  0.150048405              0.682367              0.8422510    1.747631        0
.9567891
marked 0.022570507              1.128990              1.1338506    1.991027        0
.9654182
mild   0.002319265              0.839791              0.8564682    3.038114        0
.9438354
```

But it also doesn't look like cluster 2 is doing well. Let's try it without it:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_AD_clu1.x) + age_at_scan + I(age_at_sc
an^2), data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 350.732473
iter  10 value 325.749611
iter  20 value 321.103424
final  value 321.103412
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
       (Intercept) scale(inatt_AD_clu1.x) age_at_scan I(age_at_scan^2)
deter   0.39550904             0.02335906  0.27569321       0.12214400
marked  0.09018374             0.49939611  0.14298245       0.14052052
mild    0.03629942             0.37171970  0.06849908       0.06417739
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_AD_clu1.x) +
    age_at_scan + I(age_at_scan^2), data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_AD_clu1.x) age_at_scan I(age_at_scan^2)
deter    -1.868077             -0.3973156   0.5502304      -0.04357863
marked   -3.767196              0.1229367   0.6903125      -0.03550221
mild     -6.096876             -0.1828337   1.1149854      -0.05776710

Residual Deviance: 642.2068
AIC: 666.2068
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_AD_clu1.x) age_at_scan I(age_at_scan^2)
deter  0.154420375              0.6721219    1.733652        0.9573573
marked 0.023116791              1.1308129    1.994339        0.9651206
mild   0.002249885              0.8329066    3.049524        0.9438697
```

It also looks like the two age terms are not contributing much… is it better without them?

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_AD_clu1.x), data = df, na.action=na.om
it)
```

```
# weights:  12 (6 variable)
initial  value 350.732473
iter  10 value 330.712069
final  value 330.711102
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
        (Intercept) scale(inatt_AD_clu1.x)
deter  8.659209e-04              0.1352950
marked 1.776564e-04              0.3921643
mild   7.883483e-08              0.4679238
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_AD_clu1.x),
    data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_AD_clu1.x)
deter   -0.5484969             -0.2443413
marked  -0.6351228              0.1486078
mild    -1.0445154             -0.1415168

Residual Deviance: 661.4222
AIC: 673.4222
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_AD_clu1.x)
deter    0.5778177              0.7832203
marked   0.5298704              1.1602178
mild     0.3518623              0.8680406
```

Not really. So let's keep them:

Hide

Hide

```
fit <- multinom(OLS_inatt_categ ~ scale(inatt_AD_clu1.x) + age_at_scan + I(age_at_sca
n^2), data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 350.732473
iter  10 value 325.749611
iter  20 value 321.103424
final  value 321.103412
converged
```

Hide

Hide

```
z <- summary(fit)$coefficients/summary(fit)$standard.errors
p <- (1 - pnorm(abs(z), 0, 1)) * 2
rr = exp(coef(fit))
pp1 = fitted(fit)
print('p-values')
```

```
[1] "p-values"
```

Hide

Hide

```
print(p)
```

```
       (Intercept) scale(inatt_AD_clu1.x) age_at_scan I(age_at_scan^2)
deter   0.39550904             0.02335906  0.27569321       0.12214400
marked  0.09018374             0.49939611  0.14298245       0.14052052
mild    0.03629942             0.37171970  0.06849908       0.06417739
```

Hide

Hide

```
print(fit)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_AD_clu1.x) +
    age_at_scan + I(age_at_scan^2), data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_AD_clu1.x) age_at_scan I(age_at_scan^2)
deter    -1.868077             -0.3973156   0.5502304      -0.04357863
marked   -3.767196              0.1229367   0.6903125      -0.03550221
mild     -6.096876             -0.1828337   1.1149854      -0.05776710

Residual Deviance: 642.2068
AIC: 666.2068
```

Hide

Hide

```
print('risk ratio')
```

```
[1] "risk ratio"
```

Hide

Hide

```
print(rr)
```

```
       (Intercept) scale(inatt_AD_clu1.x) age_at_scan I(age_at_scan^2)
deter  0.154420375              0.6721219    1.733652        0.9573573
marked 0.023116791              1.1308129    1.994339        0.9651206
mild   0.002249885              0.8329066    3.049524        0.9438697
```

And it looks like deterioration is the only significant category:

- A one-unit increase in AD for that cluster (i.e. increase by 1 SD) is associated with a decrease in the log odds of deteriorating (vs normals) in the amount of .40.
- In terms of relative risk ratio, a one-unit decrease in the AD of that brain cluster yields a relative risk ratio of .67 of deterioration (vs normals). In other words, the odds of deterioration, compared to normals, is .67 times higher for every 1 SD we decrease in that brain region.

Let's make a much simpler (but not too far off) model, and look at probabilities:

Hide

Hide

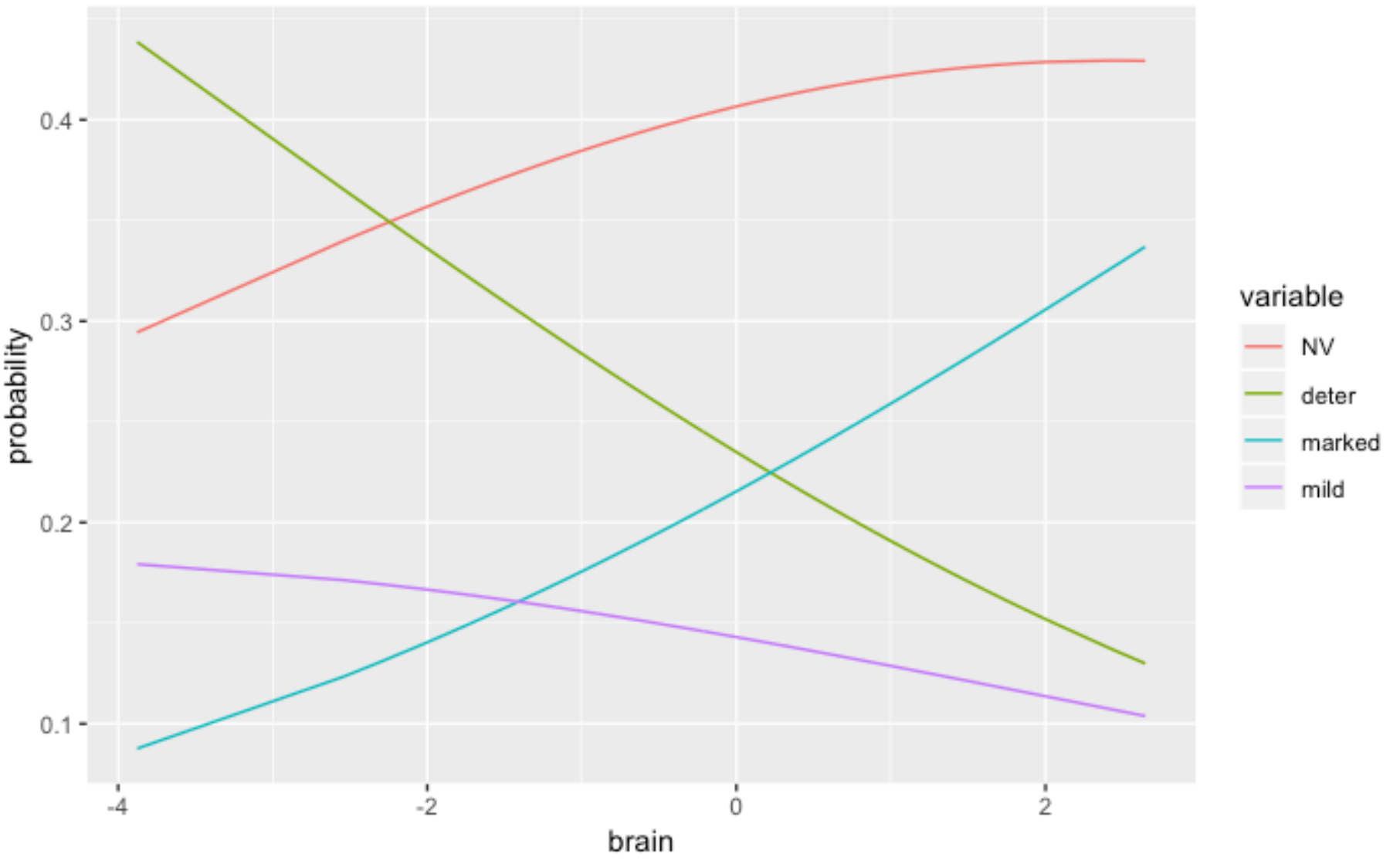
```
library(reshape2)
library(ggplot2)
fit <- multinom(OLS_inatt_categ ~ scale(inatt_AD_clu1.x), data = df, na.action=na.omi
t)
```

```
# weights:  12 (6 variable)
initial  value 350.732473
iter  10 value 330.712069
final  value 330.711102
converged
```

Hide

Hide

```
z <- summary(fit)$coefficients/summary(fit)$standard.errors
p <- (1 - pnorm(abs(z), 0, 1)) * 2
pp = fitted(fit)
a = cbind(pp, scale(df$inatt_AD_clu1.x))
colnames(a)[5] = 'brain'
lpp = melt(as.data.frame(a), value.name='probability', id.vars=c('brain'))
ggplot(lpp, aes(x=brain, y=probability, color=variable)) + geom_line()
```



Hide

Hide

```
print(p)
```

```
        (Intercept) scale(inatt_AD_clu1.x)
deter  8.659209e-04              0.1352950
marked 1.776564e-04              0.3921643
mild   7.883483e-08              0.4679238
```

Hide

Hide

```
print(fit$AIC)
```

```
[1] 673.4222
```

And for future comparisons, this is the model AUC on training data:

Hide

Hide

```
fit <- multinom(OLS_inatt_categ ~ scale(inatt_AD_clu1.x) + age_at_scan + I(age_at_sca
n^2), data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 350.732473
iter  10 value 325.749611
iter  20 value 321.103424
final  value 321.103412
converged
```

Hide

Hide

```
res.roc = multiclass.roc(df$OLS_inatt_categ, as.numeric(predict(fit1, newdata=df, typ
e='class')))
print(sprintf('AUC: %f', auc(res.roc)))
```

```
[1] "AUC: 0.491993"
```

HI, DTI

Hide

Hide

```
for (t in c('age_at_scan', 'I(age_at_scan^2)', 'mvmt', 'I(mvmt^2)', 'as.numeric(Sex)'
)) {
  fm_str = sprintf('%s ~ OLS_HI_categ', t)
  print(fm_str)
  print(summary(aov(lm(as.formula(fm_str), data=df))))
}
```

```
[1] "age_at_scan ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   22.9   7.632   1.398  0.244
Residuals    249 1359.3   5.459
[1] "I(age_at_scan^2) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3  11110    3703   1.784  0.151
Residuals    249 516880    2076
[1] "mvmt ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   1.71  0.5707   0.568  0.637
Residuals    249 250.29  1.0052
[1] "I(mvmt^2) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3    8.1   2.708   0.783  0.504
Residuals    249  860.6   3.456
[1] "as.numeric(Sex) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   1.24  0.4142   1.904   0.13
Residuals    249  54.18  0.2176
```

Well, nothing seems to matter for the HI categories… this makes the model rather simple:

Hide

Hide

```
fit <- multinom(OLS_HI_categ ~ scale(HI_RD_clu1.x), data = df, na.action=na.omit)
```

```
# weights:  12 (6 variable)
initial  value 350.732473
iter  10 value 323.504725
final  value 323.493453
converged
```

Hide

Hide

```
z <- summary(fit)$coefficients/summary(fit)$standard.errors
p <- (1 - pnorm(abs(z), 0, 1)) * 2
pp = fitted(fit)
rr = exp(coef(fit))
print('p-values')
```

```
[1] "p-values"
```

Hide

Hide

```
print(p)
```

```
        (Intercept) scale(HI_RD_clu1.x)
deter  1.047996e-08           0.2171421
marked 5.832491e-02           0.6576175
mild   1.076369e-06           0.1336489
```

Hide

Hide

```
print(fit)
```

```
Call:
multinom(formula = OLS_HI_categ ~ scale(HI_RD_clu1.x), data = df,
    na.action = na.omit)

Coefficients:
       (Intercept) scale(HI_RD_clu1.x)
deter   -1.1760087          0.25249685
marked  -0.2872967         -0.06820104
mild    -0.9093761          0.27842161

Residual Deviance: 646.9869
AIC: 658.9869
```

Hide

Hide

```
print('risk ratio')
```

```
[1] "risk ratio"
```

Hide

Hide

```
print(rr)
```

```
       (Intercept) scale(HI_RD_clu1.x)
deter    0.3085076           1.2872354
marked   0.7502891           0.9340727
mild     0.4027755           1.3210430
```

Nothing seems significant here… I wonder if the relationship to continuous OLS wasn't strong enough to begin with, and then the categorization completely vanished it?

inattention, rsFMRI

Hide

Hide

```
load('~/data/baseline_prediction/combined_descriptives_12172018.RData.gz')
clin = read.csv('~/data/baseline_prediction/long_clin_11302018.csv')
df = merge(clin, data, by='MRN')
idx = df$diag_group != 'new_onset'
fmri = df[!is.na(df$inatt_melodic_limbic) & idx,]
load('~/data/baseline_prediction/melodic_inter_IC11_12142018.RData.gz')
fmri = merge(fmri, data, by='MRN') # put mask ids in combined dataset
mprage = read.xls('~/data/baseline_prediction/long_scans_08072018.xlsx',
                  sheet='mprage')
fmri = merge(fmri, mprage, by.x='mask.id', by.y='Mask.ID...Scan') # get demographics
qc = read.csv('~/data/baseline_prediction/master_qc.csv')
fmri = merge(fmri, qc, by.x='mask.id', by.y='Mask.ID') # get QC scores
df = merge(fmri, imaging, by='MRN')
dim(df)
```

```
[1]   196 44332
```

And we check the different covariates:

First, let's check if any of the variables we used as covariates before has a relationship with the categories:

Hide

Hide

```
for (t in c('age_at_scan', 'I(age_at_scan^2)', 'enormGoodTRs_fmri01', 'I(enormGoodTRs
_fmri01^2)', 'as.numeric(Sex...Subjects)')) {
  fm_str = sprintf('%s ~ OLS_inatt_categ', t)
  print(fm_str)
  print(summary(aov(lm(as.formula(fm_str), data=df))))
}
```

```
[1] "age_at_scan ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3    7.2   2.387   0.482  0.695
Residuals      192  950.3   4.950
[1] "I(age_at_scan^2) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   3234    1078   0.501  0.682
Residuals      192 413240    2152
[1] "enormGoodTRs_fmri01 ~ OLS_inatt_categ"
                Df  Sum Sq   Mean Sq F value Pr(>F)
OLS_inatt_categ  3 0.00403 0.0013441   3.615 0.0142 *
Residuals      192 0.07138 0.0003718
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "I(enormGoodTRs_fmri01^2) ~ OLS_inatt_categ"
                Df    Sum Sq   Mean Sq F value  Pr(>F)
OLS_inatt_categ  3 0.0000795 2.651e-05   3.993 0.00868 **
Residuals      192 0.0012746 6.639e-06
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "as.numeric(Sex...Subjects) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   0.42  0.1404   0.605  0.613
Residuals      192  44.58  0.2322
```

It looks like we need to keep the movement variables:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_melodic_limbic.x) + scale(inatt_melodi
c_DMN.x) + scale(inatt_melodic_VAN.x)  + enormGoodTRs_fmri01 + I(enormGoodTRs_fmri01^
2), data = df, na.action=na.omit)
```

```
# weights:  28 (18 variable)
initial  value 271.713695
iter  10 value 224.283842
iter  20 value 222.208775
iter  30 value 220.736295
final  value 220.731330
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
       (Intercept) scale(inatt_melodic_limbic.x) scale(inatt_melodic_DMN.x) scale(ina
tt_melodic_VAN.x) enormGoodTRs_fmri01
deter    0.2112566                   0.062460445                  0.5774985
0.005229275           0.7075459
marked   0.2068515                   0.007385301                  0.1279599
0.002201450           0.7777370
mild     0.3128992                   0.404850351                  0.3423221
0.883695459           0.8240821
       I(enormGoodTRs_fmri01^2)
deter                 0.9921428
marked                0.9958572
mild                  0.9986409
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_melodic_limbic.x) +
    scale(inatt_melodic_DMN.x) + scale(inatt_melodic_VAN.x) +
    enormGoodTRs_fmri01 + I(enormGoodTRs_fmri01^2), data = df,
    na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_melodic_limbic.x) scale(inatt_melodic_DMN.x) scale(ina
tt_melodic_VAN.x) enormGoodTRs_fmri01
deter    -2.546456                    0.4071590                 -0.1115318
-0.64200186            22.52745
marked   -2.471754                   -0.7143694                  0.3565107
0.77867479            17.61632
mild     -1.918519                   -0.2067465                  0.2163674
0.03506424            13.24864
       I(enormGoodTRs_fmri01^2)
deter                 4.2212763
marked                2.4774193
mild                  0.7656817

Residual Deviance: 441.4627
AIC: 477.4627
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_melodic_limbic.x) scale(inatt_melodic_DMN.x) scale(ina
tt_melodic_VAN.x) enormGoodTRs_fmri01
deter   0.07835890                     1.5025430                   0.894463
0.5262379        6075027257.6
marked  0.08443659                     0.4895007                   1.428337
2.1785833          44737351.1
mild    0.14682431                     0.8132258                   1.241558
1.0356862            567300.8
       I(enormGoodTRs_fmri01^2)
deter                  68.12037
marked                 11.91049
mild                    2.15046
```

They don't seem to be contributing much at all. Not even the DMN cluster. Let's then try to remove some of the variables ot get a better fit:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_melodic_limbic.x) + scale(inatt_melodi
c_VAN.x)  + enormGoodTRs_fmri01, data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 271.713695
iter  10 value 226.153226
iter  20 value 222.927817
iter  30 value 222.683536
iter  40 value 222.678794
iter  50 value 222.675646
final  value 222.675643
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
       (Intercept) scale(inatt_melodic_limbic.x) scale(inatt_melodic_VAN.x) enormGood
TRs_fmri01
deter  0.001167398                  0.060094102                0.005585332
0.03741378
marked 0.001131414                  0.004447235                0.001027206
0.08412984
mild   0.010817160                  0.366678675                0.856659104
0.21491807
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_melodic_limbic.x) +
    scale(inatt_melodic_VAN.x) + enormGoodTRs_fmri01, data = df,
    na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_melodic_limbic.x) scale(inatt_melodic_VAN.x) enormGood
TRs_fmri01
deter    -2.559749                   0.4089377                -0.63396416
23.34925
marked   -2.578128                  -0.7491121                 0.82241928
19.64140
mild     -1.969818                  -0.2235916                 0.04328261
14.12073

Residual Deviance: 445.3513
AIC: 469.3513
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_melodic_limbic.x) scale(inatt_melodic_VAN.x) enormGood
TRs_fmri01
deter   0.07732417                   1.5052180                 0.5304847         1
3818196098
marked  0.07591597                   0.4727861                 2.2759995
338961940
mild    0.13948217                   0.7996417                 1.0442330
1356923
```

Yeah, that's a better model. Let's interpret it, keeping in mind that both the limbic and VAN clusters are significant for the deterioration and marked improvement conditions:

- The results of the two clusters go in opposite ways. A one-unit increase in the limbic cluster (i.e. increase by 1 SD) is associated with an increase in the log odds of deteriorating (vs normals) in the amount of .41, and a decrease in the log odds ratio of marked improvement of .75. On the other hand, one-unit increase in the VAN cluster is associated with an decrease in the log odds of deteriorating in the amount of .64, and an increase in the log odds ratio of marked improvement of .82.
- In terms of relative risk ratio, a one-unit increase in the limbic cluster yields a relative risk ratio of 1.51 of deterioration (vs normals). In other words, the odds of deterioration compared to normals is 1.5 times higher for every 1 SD we decrease in that brain region, and .53 lower for a unit increase in the VAN cluster. Conversely, every unit increase in the VAN cluster makes the odds of marked improvement 2.28 times higher.

And this is the overall AUC:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_melodic_limbic.x) + scale(inatt_melodi
c_VAN.x)  + enormGoodTRs_fmri01, data = df, na.action=na.omit)
```

```
# weights:  20 (12 variable)
initial  value 271.713695
iter  10 value 226.153226
iter  20 value 222.927817
iter  30 value 222.683536
iter  40 value 222.678794
iter  50 value 222.675646
final  value 222.675643
converged
```

Hide

Hide

```
res.roc = multiclass.roc(df$OLS_inatt_categ, as.numeric(predict(fit1, newdata=df, typ
e='class')))
print(sprintf('AUC: %f', auc(res.roc)))
```

```
[1] "AUC: 0.524545"
```

The numbers are certainly not impressive, but fMRI is the best modality so far.

Note that there's no significant clusters for HI rsFMRI…

inattention, all imaging combined

The first step here is to impute the data across modalities. Then, we'll need to decide which age variable to run (maybe median age?), and finally decide on which variables and models to use. The main question is whether we get better models, and better predictions, when combining across modalities.

Let's start by making a Venn diagram to assess how much data we have across modalities.

Hide

Hide

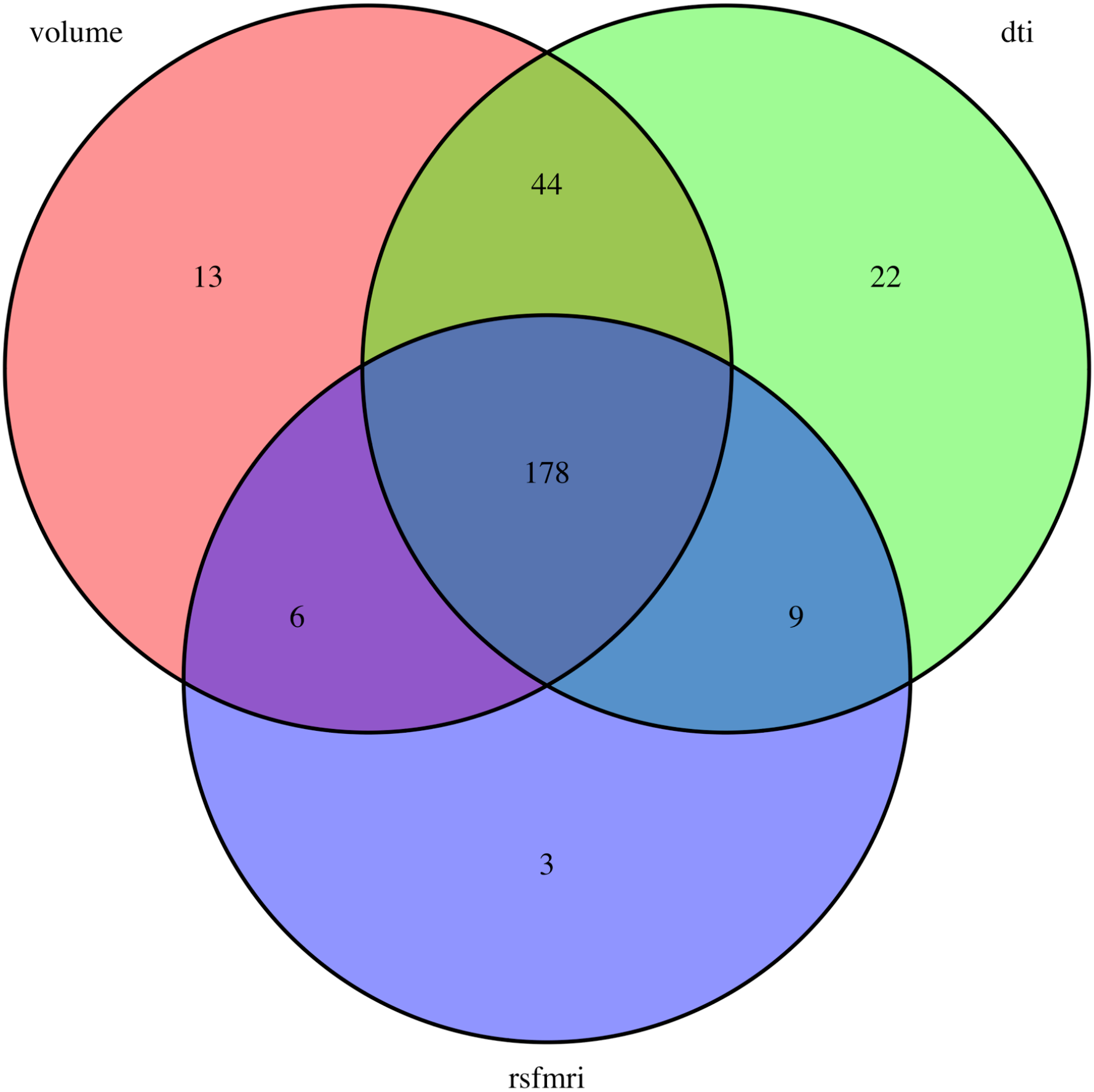
```
library(VennDiagram)
```

```
Loading required package: grid
Loading required package: futile.logger
```

Hide

Hide

```
venn.plot = venn.diagram(list(volume = which(!is.na(imaging$inatt_vol_lh)),
                              dti = which(!is.na(imaging$inatt_AD_clu1)),
                              rsfmri = which(!is.na(imaging$inatt_melodic_DMN))),
                         euler.d=TRUE, fill=c('red','green', 'blue'), filename='myven
n.tiff')
```



Here is the Venn plott

We might actually be able to test it with only the kids that have everything, and then try the imputed dataset.

Hide

Hide

```
idx = imaging$diag_group != 'new_onset'
idx2 = !is.na(imaging$inatt_AD_clu1) & !is.na(imaging$inatt_melodic_DMN) & !is.na(ima
ging$inatt_vol_lh)
df = imaging[idx2 & idx,]
load('~/data/baseline_prediction/dti_rd_voxelwise_n272_09212018.RData.gz')
df = merge(df, data[, 1:2], by='MRN') # put mask ids in combined dataset
demo = read.xls('~/data/baseline_prediction/long_scans_08072018.xlsx',
                sheet='dti')
df = merge(df, demo, by.x='mask.id', by.y='Mask.ID') # get demographics
qc = read.csv('~/data/baseline_prediction/master_qc.csv')
df = merge(df, qc, by.x='mask.id', by.y='Mask.ID') # get QC scores
df$mvmt = rowMeans(scale(df$norm.trans), scale(df$norm.rot))
load('~/data/baseline_prediction/melodic_inter_IC11_12142018.RData.gz')
df = merge(df, data[, 1:2], by='MRN', suffixes = c('.dti', '.rsfmri')) # put mask ids
in combined dataset
demo = read.xls('~/data/baseline_prediction/long_scans_08072018.xlsx',
                sheet='mprage')
df = merge(df, demo, by.x='mask.id.rsfmri', by.y='Mask.ID...Scan', suffixes = c('.dti
', '.rsfmri')) # get demographics
df = merge(df, qc, by.x='mask.id.rsfmri', by.y='Mask.ID', suffixes = c('.dti', '.rsfm
ri')) # get QC scores
load('~/data/baseline_prediction/struct_volume_11142018_260timeDiff12mo.RData.gz')
df = merge(df, data[, 1:2], by='MRN', suffixes = c('.dtiAndrsFMRI', '.vol')) # put ma
sk ids in combined dataset
df = merge(df, demo, by.x='mask.id', by.y='Mask.ID...Scan', suffixes = c('.dtiAndrsfm
ri', 'vol')) # get demographics
df = merge(df, qc, by.x='mask.id', by.y='Mask.ID', suffixes = c('.dtiAndrsfmri', 'vol
')) # get QC scores
dim(df)
```

```
[1] 178 152
```

Hide

Hide

```
for (t in c(# volume
            'age_at_scan', 'I(age_at_scan^2)', 'ext_avg_freesurfer5.3', 'int_avg_free
surfer5.3', 'mprage_QC', 'as.numeric(Sex)',
            # DTI
           'age_at_scan.dti', 'I(age_at_scan.dti^2)', 'mvmt', 'I(mvmt^2)', # 'as.nume
ric(Sex)',
          # rsfmri
           'age_at_scan.rsfmri', 'I(age_at_scan.rsfmri^2)', 'enormGoodTRs_fmri01', 'I
(enormGoodTRs_fmri01^2)')) { #}, 'as.numeric(Sex)')) {
  fm_str = sprintf('%s ~ OLS_inatt_categ', t)
  print(fm_str)
  print(summary(aov(lm(as.formula(fm_str), data=df))))
}
```

```
[1] "age_at_scan ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   25.1   8.360   1.523   0.21
Residuals      174  955.1   5.489
[1] "I(age_at_scan^2) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3  10185    3395   1.529  0.209
Residuals      174 386439    2221
[1] "ext_avg_freesurfer5.3 ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3  0.696  0.2319    1.98  0.119
Residuals      174 20.375  0.1171
[1] "int_avg_freesurfer5.3 ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3  0.825  0.2751   2.596  0.054 .
Residuals      174 18.440  0.1060
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "mprage_QC ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   0.07 0.02199   0.118  0.949
Residuals      174  32.34 0.18585
[1] "as.numeric(Sex) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   0.76  0.2520   1.082  0.358
Residuals      174  40.51  0.2328
[1] "age_at_scan.dti ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3   34.5  11.509   2.098  0.102
Residuals      174  954.6   5.486
[1] "I(age_at_scan.dti^2) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3  14298    4766   2.132  0.098 .
Residuals      174 389045    2236
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] "mvmt ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3    1.9  0.6322   0.628  0.598
Residuals      174  175.1  1.0063
[1] "I(mvmt^2) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3    2.6   0.868     0.2  0.896
Residuals      174  753.7   4.332
[1] "age_at_scan.rsfmri ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
OLS_inatt_categ  3    8.3   2.768   0.553  0.647
Residuals      174  871.1   5.006
[1] "I(age_at_scan.rsfmri^2) ~ OLS_inatt_categ"
                Df Sum Sq Mean Sq F value Pr(>F)
```

```
OLS_inatt_categ   3   3515    1172   0.538  0.657
Residuals       174 378592    2176
[1] "enormGoodTRs_fmri01 ~ OLS_inatt_categ"
                 Df  Sum Sq   Mean Sq F value Pr(>F)
OLS_inatt_categ   3 0.00415 0.0013822   2.904 0.0372 *
Residuals       133 0.06330 0.0004759
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
41 observations deleted due to missingness
[1] "I(enormGoodTRs_fmri01^2) ~ OLS_inatt_categ"
                 Df    Sum Sq   Mean Sq F value Pr(>F)
OLS_inatt_categ   3 0.0000961 3.204e-05   3.369 0.0206 *
Residuals       133 0.0012650 9.510e-06
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
41 observations deleted due to missingness
```

OK, so a couple of the QC variables seem significant. Let’s include them in the model:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_vol_lh) + scale(inatt_AD_clu1) + scale
(inatt_melodic_limbic) + scale(inatt_melodic_VAN) + int_avg_freesurfer5.3 + enormGood
TRs_fmri01, data = df, na.action=na.omit)
```

```
# weights:  32 (21 variable)
initial  value 189.922327
iter  10 value 153.530329
iter  20 value 149.878634
iter  30 value 147.479305
iter  40 value 147.374155
final  value 147.374091
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
       (Intercept) scale(inatt_vol_lh) scale(inatt_AD_clu1) scale(inatt_melodic_limbi
c) scale(inatt_melodic_VAN)
deter  0.003245379           0.4669615           0.91680525                  0.9963950
86              0.117563959
marked 0.010866914           0.5541645           0.09849125                  0.0015375
16              0.001309272
mild   0.248140539           0.4158516           0.78957907                  0.1981045
03              0.730657554
       int_avg_freesurfer5.3 enormGoodTRs_fmri01
deter             0.06583617          0.12341271
marked            0.20790485          0.07185099
mild              0.93000407          0.19434771
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_vol_lh) + scale(inatt_AD_clu1) +
    scale(inatt_melodic_limbic) + scale(inatt_melodic_VAN) +
    int_avg_freesurfer5.3 + enormGoodTRs_fmri01, data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_vol_lh) scale(inatt_AD_clu1) scale(inatt_melodic_limbi
c) scale(inatt_melodic_VAN)
deter    -5.322873          -0.2158985          -0.02529313                 0.0013417
23              -0.4692934
marked   -5.476507           0.1754786           0.55414723                -1.2056425
40               1.2293371
mild     -1.961252          -0.2465458           0.07077354                -0.4073259
35               0.1090576
       int_avg_freesurfer5.3 enormGoodTRs_fmri01
deter             1.62844142            19.09313
marked            1.17235353            24.42335
mild             -0.07421008            16.32576

Residual Deviance: 294.7482
AIC: 336.7482
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_vol_lh) scale(inatt_AD_clu1) scale(inatt_melodic_limbi
c) scale(inatt_melodic_VAN)
deter  0.004878719           0.8058171            0.9750241                    1.00134
26                0.6254441
marked 0.004183920           1.1918165            1.7404561                    0.29949
95                3.4189623
mild   0.140682130           0.7814956            1.0733381                    0.66542
73                1.1152266
       int_avg_freesurfer5.3 enormGoodTRs_fmri01
deter              5.0959261           195903521
marked             3.2295846         40450534570
mild               0.9284766            12308017
```

It's a completely different sample, so I'm not comfortable using AIC here. Let's see if AUC is any better:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_vol_lh) + scale(inatt_AD_clu1) + scale
(inatt_melodic_limbic) + scale(inatt_melodic_VAN) + int_avg_freesurfer5.3 + enormGood
TRs_fmri01, data = df, na.action=na.omit)
```

```
# weights:  32 (21 variable)
initial  value 189.922327
iter  10 value 153.530329
iter  20 value 149.878634
iter  30 value 147.479305
iter  40 value 147.374155
final  value 147.374091
converged
```

Hide

Hide

```
res.roc = multiclass.roc(df$OLS_inatt_categ, as.numeric(predict(fit1, newdata=df, typ
e='class')))
print(sprintf('AUC: %f', auc(res.roc)))
```

```
[1] "AUC: 0.652046"
```

Yep, considerably better AUC than any imaging modality by itself.

HI, all imaging combined

Hide

Hide

```
for (t in c(# volume
            'age_at_scan', 'I(age_at_scan^2)', 'ext_avg_freesurfer5.3', 'int_avg_free
surfer5.3', 'mprage_QC', 'as.numeric(Sex)',
            # DTI
           'age_at_scan.dti', 'I(age_at_scan.dti^2)', 'mvmt', 'I(mvmt^2)', #'as.numer
ic(Sex)',
           # rsfmri
           'age_at_scan.rsfmri', 'I(age_at_scan.rsfmri^2)', 'enormGoodTRs_fmri01', 'I
(enormGoodTRs_fmri01^2)')) { #}, 'as.numeric(Sex)')) {
  fm_str = sprintf('%s ~ OLS_HI_categ', t)
  print(fm_str)
  print(summary(aov(lm(as.formula(fm_str), data=df))))
}
```

```
[1] "age_at_scan ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3    6.3   2.111   0.377   0.77
Residuals    174  973.8   5.597
[1] "I(age_at_scan^2) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   4561    1520   0.675  0.569
Residuals    174 392063    2253
[1] "ext_avg_freesurfer5.3 ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3  0.305  0.1018   0.853  0.467
Residuals    174 20.766  0.1193
[1] "int_avg_freesurfer5.3 ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3  0.577  0.1924   1.791  0.151
Residuals    174 18.689  0.1074
[1] "mprage_QC ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   0.75  0.2489   1.368  0.254
Residuals    174  31.66  0.1819
[1] "as.numeric(Sex) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   1.03  0.3432   1.484  0.221
Residuals    174  40.23  0.2312
[1] "age_at_scan.dti ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   16.4   5.468   0.978  0.404
Residuals    174  972.7   5.590
[1] "I(age_at_scan.dti^2) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   9098    3033   1.338  0.264
Residuals    174 394245    2266
[1] "mvmt ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   1.85  0.6163   0.612  0.608
```

```
Residuals    174 175.15  1.0066
[1] "I(mvmt^2) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3    1.8   0.583   0.135  0.939
Residuals    174  754.6   4.337
[1] "age_at_scan.rsfmri ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   14.3   4.772    0.96  0.413
Residuals    174  865.1   4.972
[1] "I(age_at_scan.rsfmri^2) ~ OLS_HI_categ"
              Df Sum Sq Mean Sq F value Pr(>F)
OLS_HI_categ   3   6924    2308    1.07  0.363
Residuals    174 375182    2156
[1] "enormGoodTRs_fmri01 ~ OLS_HI_categ"
              Df  Sum Sq   Mean Sq F value Pr(>F)
OLS_HI_categ   3 0.00526 0.0017526   3.748 0.0126 *
Residuals    133 0.06219 0.0004676
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
41 observations deleted due to missingness
[1] "I(enormGoodTRs_fmri01^2) ~ OLS_HI_categ"
              Df    Sum Sq   Mean Sq F value Pr(>F)
OLS_HI_categ   3 0.0001043 3.477e-05   3.679 0.0138 *
Residuals    133 0.0012569 9.450e-06
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
41 observations deleted due to missingness
```

Only the fMRI QC variables were significant. But since we have no fMRI clusters, I won’t include them.

Hide

Hide

```
fit1 <- multinom(OLS_HI_categ ~ scale(HI_vol_rh) + scale(HI_RD_clu1), data = df, na.a
ction=na.omit)
```

```
# weights:  16 (9 variable)
initial  value 246.760396
iter  10 value 214.980127
final  value 214.499004
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
        (Intercept) scale(HI_vol_rh) scale(HI_RD_clu1)
deter  7.741101e-08       0.02553413         0.2172264
marked 6.141631e-02       0.04156384         0.7492994
mild   3.996606e-06       0.54053282         0.2471407
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_HI_categ ~ scale(HI_vol_rh) + scale(HI_RD_clu1),
    data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(HI_vol_rh) scale(HI_RD_clu1)
deter   -1.5620277        0.5208232       -0.33596904
marked  -0.3427908       -0.4346919       -0.05762521
mild    -1.0736779        0.1377555        0.26655054

Residual Deviance: 428.998
AIC: 446.998
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(HI_vol_rh) scale(HI_RD_clu1)
deter    0.2097104        1.6834129         0.7146452
marked   0.7097867        0.6474641         0.9440037
mild     0.3417493        1.1476949         1.3054536
```

Hide

Hide

```
fit1 <- multinom(OLS_HI_categ ~ scale(HI_vol_rh) + scale(HI_RD_clu1), data = df, na.a
ction=na.omit)
```

```
# weights:  16 (9 variable)
initial  value 246.760396
iter  10 value 214.980127
final  value 214.499004
converged
```

Hide

Hide

```
res.roc = multiclass.roc(df$OLS_inatt_categ, as.numeric(predict(fit1, newdata=df, typ
e='class')))
print(sprintf('AUC: %f', auc(res.roc)))
```

```
[1] "AUC: 0.506923"
```

Not much gained here.

inatt, all imaging imputed

Hide

Hide

```
idx = imaging$diag_group != 'new_onset'
library(VIM)
brain_cols = c("HI_vol_rh", "inatt_vol_lh", "HI_RD_clu1", "inatt_AD_clu1", "inatt_AD_
clu2", "inatt_melodic_limbic", "inatt_melodic_DMN", "inatt_melodic_VAN")
imputed_brain = irmi(imaging[idx, brain_cols])
df = imaging[idx, ]
df[, brain_cols] = imputed_brain
load('~/data/baseline_prediction/dti_rd_voxelwise_n272_09212018.RData.gz')
df = merge(df, data[, 1:2], by='MRN', all.x=T) # put mask ids in combined dataset
demo = read.xls('~/data/baseline_prediction/long_scans_08072018.xlsx',
                sheet='dti')
df = merge(df, demo, by.x='mask.id', by.y='Mask.ID', all.x=T) # get demographics
qc = read.csv('~/data/baseline_prediction/master_qc.csv')
df = merge(df, qc, by.x='mask.id', by.y='Mask.ID', all.x=T) # get QC scores
df$mvmt = rowMeans(scale(df$norm.trans), scale(df$norm.rot))
load('~/data/baseline_prediction/melodic_inter_IC11_12142018.RData.gz')
df = merge(df, data[, 1:2], by='MRN', suffixes = c('.dti', '.rsfmri'), all.x=T) # put
mask ids in combined dataset
demo = read.xls('~/data/baseline_prediction/long_scans_08072018.xlsx',
                sheet='mprage')
df = merge(df, demo, by.x='mask.id.rsfmri', by.y='Mask.ID...Scan', suffixes = c('.dti
', '.rsfmri'), all.x=T) # get demographics
df = merge(df, qc, by.x='mask.id.rsfmri', by.y='Mask.ID', suffixes = c('.dti', '.rsfm
ri'), all.x=T) # get QC scores
load('~/data/baseline_prediction/struct_volume_11142018_260timeDiff12mo.RData.gz')
df = merge(df, data[, 1:2], by='MRN', suffixes = c('.dtiAndrsFMRI', '.vol'), all.x=T)
# put mask ids in combined dataset
df = merge(df, demo, by.x='mask.id', by.y='Mask.ID...Scan', suffixes = c('.dtiAndrsfm
ri', 'vol'), all.x=T) # get demographics
df = merge(df, qc, by.x='mask.id', by.y='Mask.ID', suffixes = c('.dtiAndrsfmri', 'vol
'), all.x=T) # get QC scores
dim(df)
```

```
[1] 275 152
```

I dont' expect the covariate search to change here, as it's only between the category and the covariate, and we only imputed brain data. Let's jump straight into the model:

Hide

Hide

```
fit1 <- multinom(OLS_inatt_categ ~ scale(inatt_vol_lh) + scale(inatt_AD_clu1) + scale
(inatt_melodic_limbic) + scale(inatt_melodic_VAN) + int_avg_freesurfer5.3 + enormGood
TRs_fmri01, data = df, na.action=na.omit)
```

```
# weights:  32 (21 variable)
initial  value 230.124864
iter  10 value 192.404525
iter  20 value 189.469530
iter  30 value 187.176714
iter  40 value 186.945990
final  value 186.945617
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
        (Intercept) scale(inatt_vol_lh) scale(inatt_AD_clu1) scale(inatt_melodic_limb
ic) scale(inatt_melodic_VAN)
deter  0.0005193616           0.3587032           0.35302250                  0.971061
076               0.08985788
marked 0.0118716420           0.1801264           0.07693865                  0.007029
938               0.01595214
mild   0.3653559417           0.6007400           0.68710856                  0.315474
038               0.91070975
       int_avg_freesurfer5.3 enormGoodTRs_fmri01
deter              0.05375099          0.03085285
marked             0.26504528          0.17849219
mild               0.88710469          0.65645480
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_inatt_categ ~ scale(inatt_vol_lh) + scale(inatt_AD_clu1) +
    scale(inatt_melodic_limbic) + scale(inatt_melodic_VAN) +
    int_avg_freesurfer5.3 + enormGoodTRs_fmri01, data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(inatt_vol_lh) scale(inatt_AD_clu1) scale(inatt_melodic_limbi
c) scale(inatt_melodic_VAN)
deter    -5.103711          -0.2342078          -0.1981469                -0.0089711
17              -0.41168760
marked   -3.729061           0.3147460           0.4548694                -0.7695404
17               0.62031957
mild     -1.330394          -0.1437962           0.1007549                -0.2874102
77              -0.03122437
       int_avg_freesurfer5.3 enormGoodTRs_fmri01
deter              1.3614050           22.682478
marked             0.7661440           13.935879
mild              -0.1043757            4.989148

Residual Deviance: 373.8912
AIC: 415.8912
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(inatt_vol_lh) scale(inatt_AD_clu1) scale(inatt_melodic_limbi
c) scale(inatt_melodic_VAN)
deter  0.006074162           0.7911974           0.8202493                 0.99106
90                0.6625312
marked 0.024015367           1.3699113           1.5759676                 0.46322
59                1.8595222
mild   0.264373053           0.8660643           1.1060055                 0.75020
39                0.9692581
       int_avg_freesurfer5.3 enormGoodTRs_fmri01
deter              3.9016713        7.093733e+09
marked             2.1514543        1.127912e+06
mild               0.9008868        1.468113e+02
```

Hide

Hide

```
res.roc = multiclass.roc(df$OLS_inatt_categ, as.numeric(predict(fit1, newdata=df, typ
e='class')))
print(sprintf('AUC: %f', auc(res.roc)))
```

```
[1] "AUC: 0.531637"
```

Imputation doesn't seem to help inattention too much, and AUC is not as good.

HI, all imaging imputed

Hide

Hide

```
fit1 <- multinom(OLS_HI_categ ~ scale(HI_vol_rh) + scale(HI_RD_clu1), data = df, na.a
ction=na.omit)
```

```
# weights:  16 (9 variable)
initial  value 381.230949
iter  10 value 342.707677
final  value 342.395803
converged
```

Hide

Hide

```
z1 <- summary(fit1)$coefficients/summary(fit1)$standard.errors
p1 <- (1 - pnorm(abs(z1), 0, 1)) * 2
rr1 = exp(coef(fit1))
pp1 = fitted(fit1)
print(p1)
```

```
        (Intercept) scale(HI_vol_rh) scale(HI_RD_clu1)
deter  2.722027e-10       0.02594683         0.1847146
marked 1.754844e-02       0.03614871         0.8455991
mild   2.382043e-07       0.82077021         0.1083097
```

Hide

Hide

```
print(fit1)
```

```
Call:
multinom(formula = OLS_HI_categ ~ scale(HI_vol_rh) + scale(HI_RD_clu1),
    data = df, na.action = na.omit)

Coefficients:
       (Intercept) scale(HI_vol_rh) scale(HI_RD_clu1)
deter   -1.3604213        0.4074236        0.27034245
marked  -0.3538149       -0.3478596       -0.02904116
mild    -0.9188801       -0.0419776        0.28793459

Residual Deviance: 684.7916
AIC: 702.7916
```

Hide

Hide

```
print(rr1)
```

```
       (Intercept) scale(HI_vol_rh) scale(HI_RD_clu1)
deter    0.2565527        1.5029406         1.3104131
marked   0.7020049        0.7061980         0.9713765
mild     0.3989656        0.9588913         1.3336701
```

Hide

Hide

```
res.roc = multiclass.roc(df$OLS_inatt_categ, as.numeric(predict(fit1, newdata=df, typ
e='class')))
print(sprintf('AUC: %f', auc(res.roc)))
```

```
[1] "AUC: 0.514262"
```

Slightly better for HI, but not great.

Questions

- Should we do this using mixel effects given the family structure in the data (and also because that's how we obtained the clusters)? Or simply jump into ML and get the test data ready?
- Can we gain anything by selecting the clusters based on the categorical data? (i.e. use the same logistical model for cluster selection and here?)