```
            rownames(phen_res)[nrow(phen_res)] = fm_str
        } else {
            # fit broke
            temp = rep(NA, 10)
            phen_res = rbind(phen_res, temp)
            rownames(phen_res)[nrow(phen_res)] = fm_str
        }
    }
    phen_res = data.frame(phen_res)
    phen_res$formula = rownames(phen_res)
    phen_res$outcome = phen
    hold = rbind(hold, phen_res)
}
colnames(hold)[6:10] = c('logLik', 'AIC', 'BIC', 'brainVar', 'modtype')
write.csv(hold, file=out_fname, row.names=F)
```

Let's see if that changes our FDR results:

```
res =
read.csv('~/data/baseline_prediction/prs_start/univar_medianClinDiff1_4groupOrdered_lmeAgeAndSex.csv')
res = res[res$modtype=='linear',]
# keep only top PRS
prs_rows = which(grepl(res$brainVar, pattern='^ADHD') &
                 grepl(res$outcome, pattern='_inatt_'))
inatt_best = prs_rows[which.min(res[prs_rows, 'p.value'])]
prs_rows = which(grepl(res$brainVar, pattern='^ADHD') &
                 grepl(res$outcome, pattern='_hi_'))
hi_best = prs_rows[which.min(res[prs_rows, 'p.value'])]
res_inatt = rbind(res[!grepl(res$brainVar, pattern='^ADHD') &
grepl(res$outcome, pattern='_inatt_'),],
                  res[inatt_best, ])
p2_inatt = p.adjust(res_inatt[, 'p.value'], method='fdr')
res_hi = rbind(res[!grepl(res$brainVar, pattern='^ADHD') &
grepl(res$outcome, pattern='_hi_'),],
                  res[hi_best, ])
p2_hi = p.adjust(res_hi[, 'p.value'], method='fdr')
print(res_inatt[p2_inatt<.05,c('brainVar', 'outcome', 'p.value')])
print(res_hi[p2_hi<.05,c('brainVar', 'outcome', 'p.value')])
print(res_inatt[p2_inatt<.1,c('brainVar', 'outcome', 'p.value')])
print(res_hi[p2_hi<.1,c('brainVar', 'outcome', 'p.value')])
```

```
> print(res_inatt[p2_inatt<.05,c('brainVar', 'outcome', 'p.value')])
     brainVar                    outcome      p.value
85  VMI.beery ORDthreshMED_inatt_GE6_wp05 6.364996e-05
106      FSIQ ORDthreshMED_inatt_GE6_wp05 2.742290e-05
112  base_age ORDthreshMED_inatt_GE6_wp05 1.674481e-03
> print(res_hi[p2_hi<.05,c('brainVar', 'outcome', 'p.value')])
            brainVar                  outcome      p.value
```

```
172              OFC ORDthreshMED_hi_GE6_wp05 3.670474e-03
178           ATR_fa ORDthreshMED_hi_GE6_wp05 6.308342e-03
181           CST_fa ORDthreshMED_hi_GE6_wp05 1.142843e-02
190           IFO_fa ORDthreshMED_hi_GE6_wp05 8.076652e-03
202        VMI.beery ORDthreshMED_hi_GE6_wp05 2.243446e-05
220            VM.wj ORDthreshMED_hi_GE6_wp05 7.631343e-03
223             FSIQ ORDthreshMED_hi_GE6_wp05 7.202085e-05
121 ADHD_PRS0.001000 ORDthreshMED_hi_GE6_wp05 1.119771e-02
> print(res_inatt[p2_inatt<.1,c('brainVar', 'outcome', 'p.value')])
            brainVar                    outcome      p.value
55               OFC ORDthreshMED_inatt_GE6_wp05 1.315436e-02
61            ATR_fa ORDthreshMED_inatt_GE6_wp05 1.348709e-02
70             CC_fa ORDthreshMED_inatt_GE6_wp05 2.003963e-02
85         VMI.beery ORDthreshMED_inatt_GE6_wp05 6.364996e-05
103            VM.wj ORDthreshMED_inatt_GE6_wp05 1.930910e-02
106             FSIQ ORDthreshMED_inatt_GE6_wp05 2.742290e-05
112         base_age ORDthreshMED_inatt_GE6_wp05 1.674481e-03
4   ADHD_PRS0.001000 ORDthreshMED_inatt_GE6_wp05 1.255169e-02
> print(res_hi[p2_hi<.1,c('brainVar', 'outcome', 'p.value')])
            brainVar                  outcome      p.value
172              OFC ORDthreshMED_hi_GE6_wp05 3.670474e-03
178           ATR_fa ORDthreshMED_hi_GE6_wp05 6.308342e-03
181           CST_fa ORDthreshMED_hi_GE6_wp05 1.142843e-02
190           IFO_fa ORDthreshMED_hi_GE6_wp05 8.076652e-03
202        VMI.beery ORDthreshMED_hi_GE6_wp05 2.243446e-05
220            VM.wj ORDthreshMED_hi_GE6_wp05 7.631343e-03
223             FSIQ ORDthreshMED_hi_GE6_wp05 7.202085e-05
121 ADHD_PRS0.001000 ORDthreshMED_hi_GE6_wp05 1.119771e-02
```

This variable selection might look nicer too, as sex had big weights in the 2-class model and won't be there anymore. We also lost temporal from hi, which loked a bit funky too.

So, let's re-run the big-models then:

```r
library(caret)
library(nnet)
library(pROC)

data = readRDS('~/data/baseline_prediction/prs_start/complete_massagedRawNeuropsychResidsNoComorbiditiesIRMI_clinDiffGE1_03062020.rds')

# clinDiff1
inatt_vars = c('OFC', 'ATR_fa', 'CC_fa', 'VMI.beery', 'VM.wj', 'FSIQ', 'base_age',
               'ADHD_PRS0.001000')
hi_vars = c('OFC', 'ATR_fa', 'CST_fa', 'IFO_fa',
            'VMI.beery', 'VM.wj', 'FSIQ', 'ADHD_PRS0.001000')

covars = c()
min_sx = 6
```

```
# 4 classes
for (sx in c('inatt', 'hi')) {
    set.seed(42)
    phen = sprintf('threshMED_%s_GE%d_wp05', sx, min_sx)
    eval(parse(text=sprintf('this_data = data[, c(phen, %s_vars,
covars)]',
                            sx)))

    scale_me = c()
    for (v in colnames(this_data)) {
        if (!is.factor(this_data[, v])) {
            scale_me = c(scale_me, v)
        }
    }
    this_data[, scale_me] = scale(this_data[, scale_me])

    eval(parse(text=sprintf('predictors_str=paste(%s_vars, collapse="+")',
sx)))
    if (length(covars) > 0) {
        fm_str = paste(phen, " ~ ", predictors_str, ' + ',
               paste(covars, collapse='+'),
               sep="")
    } else {
        fm_str = paste(phen, " ~ ", predictors_str, sep="")
    }

    fit = multinom(as.formula(fm_str), data=this_data, maxit=2000)
    preds = predict(fit, type='prob')
    print(sx)
    print(varImp(fit))
    print(multiclass.roc(this_data[, phen], preds))
}

# 3 classes
for (sx in c('inatt', 'hi')) {
    set.seed(42)
    phen = sprintf('threshMED_%s_GE%d_wp05', sx, min_sx)
    eval(parse(text=sprintf('this_data = data[, c(phen, %s_vars,
covars)]',
                            sx)))

    this_data = this_data[this_data[, phen] != 'nv012',]
    this_data[, phen] = factor(this_data[, phen], ordered=F)
    this_data[, phen] = relevel(this_data[, phen], ref='notGE6adhd')

    scale_me = c()
    for (v in colnames(this_data)) {
        if (!is.factor(this_data[, v])) {
            scale_me = c(scale_me, v)
        }
    }
    this_data[, scale_me] = scale(this_data[, scale_me])
    eval(parse(text=sprintf('predictors_str=paste(%s_vars, collapse="+")',
sx)))
    if (length(covars) > 0) {
        fm_str = paste(phen, " ~ ", predictors_str, ' + ',
               paste(covars, collapse='+'),
               sep="")
    } else {
        fm_str = paste(phen, " ~ ", predictors_str, sep="")
    }
    fit = multinom(as.formula(fm_str), data=this_data, maxit=2000)
    preds = predict(fit, type='prob')
    print(sx)
    print(varImp(fit))
    print(multiclass.roc(this_data[, phen], preds))
}

# 2 classes
for (sx in c('inatt', 'hi')) {
    set.seed(42)
    phen = sprintf('threshMED_%s_GE%d_wp05', sx, min_sx)
    eval(parse(text=sprintf('this_data = data[, c(phen, %s_vars,
covars)]',
                            sx)))

    this_data = this_data[this_data[, phen] != 'nv012',]
    this_data = this_data[this_data[, phen] != 'notGE6adhd',]
    this_data[, phen] = factor(this_data[, phen], ordered=F)
    this_data[, phen] = relevel(this_data[, phen], ref='nonimp')

    scale_me = c()
    for (v in colnames(this_data)) {
        if (!is.factor(this_data[, v])) {
            scale_me = c(scale_me, v)
        }
    }
    this_data[, scale_me] = scale(this_data[, scale_me])

    eval(parse(text=sprintf('predictors_str=paste(%s_vars, collapse="+")',
sx)))
    if (length(covars) > 0) {
        fm_str = paste(phen, " ~ ", predictors_str, ' + ',
               paste(covars, collapse='+'),
               sep="")
    } else {
        fm_str = paste(phen, " ~ ", predictors_str, sep="")
    }
    fit = multinom(as.formula(fm_str), data=this_data, maxit=2000)
    preds = predict(fit, type='prob')
    print(sx)
    print(varImp(fit))
    print(multiclass.roc(this_data[, phen], preds))
}

covars = c('base_inatt', 'base_hi')
```

```
# repeat 2 classes!
```

And these are the new weights:

```
[1] "inatt"
                   Overall
OFC              0.8367455
ATR_fa           0.8195108
CC_fa            0.6742910
VMI.beery        0.8237748
VM.wj            1.8142522
FSIQ             1.0345247
base_age         1.2131446
ADHD_PRS0.001000 0.5876132
Data: multivariate predictor preds with 4 levels of this_data[, phen]:
nv012, imp, nonimp, notGE6adhd.
Multi-class area under the curve: 0.734

[1] "hi"
                   Overall
OFC              0.7987530
ATR_fa           0.5377048
CST_fa           0.7324253
IFO_fa           0.8339179
VMI.beery        0.9524689
VM.wj            1.8213355
FSIQ             1.0785322
ADHD_PRS0.001000 0.7009769
Data: multivariate predictor preds with 4 levels of this_data[, phen]:
nv012, imp, nonimp, notGE6adhd.
Multi-class area under the curve: 0.7244


[1] "inatt"
                   Overall
OFC              0.6499210
ATR_fa           0.7808217
CC_fa            0.6369187
VMI.beery        0.6483071
VM.wj            1.0741005
FSIQ             2.1332369
base_age         1.0379867
ADHD_PRS0.001000 1.0960322
Data: multivariate predictor preds with 3 levels of this_data[, phen]:
notGE6adhd, imp, nonimp.
Multi-class area under the curve: 0.7599

[1] "hi"
                   Overall
OFC              0.6165721
ATR_fa           0.5419530
```

```
CST_fa           0.7750120
IFO_fa           0.9976900
VMI.beery        0.5432081
VM.wj            0.9974110
FSIQ             1.9307809
ADHD_PRS0.001000 1.4039214
Data: multivariate predictor preds with 3 levels of this_data[, phen]:
notGE6adhd, imp, nonimp.
Multi-class area under the curve: 0.7534


[1] "inatt"
                    Overall
OFC              0.30150387
ATR_fa           0.24065790
CC_fa            0.58822510
VMI.beery        0.07100498
VM.wj            0.04721154
FSIQ             0.45019041
base_age         0.81722046
ADHD_PRS0.001000 0.17706220
Data: preds with 2 levels of this_data[, phen]: nonimp, imp.
Multi-class area under the curve: 0.7349

[1] "hi"
                    Overall
OFC              0.40404437
ATR_fa           0.15883720
CST_fa           0.82841940
IFO_fa           0.08032946
VMI.beery        0.08718376
VM.wj            0.06586944
FSIQ             0.61603773
ADHD_PRS0.001000 0.20064926
Data: preds with 2 levels of this_data[, phen]: nonimp, imp.
Multi-class area under the curve: 0.74
```

And we add in base_sx for the clinical domain:

```
[1] "inatt"
                   Overall
OFC              0.2226160
ATR_fa           0.5727578
CC_fa            0.6268315
VMI.beery        0.0765019
VM.wj            0.1496885
FSIQ             0.2279085
base_age         0.1872046
ADHD_PRS0.001000 0.4410067
base_inatt       1.4675412
base_hi          0.7176022
Data: preds with 2 levels of this_data[, phen]: nonimp, imp.
```

```
Multi-class area under the curve: 0.8582

[1] "hi"
                    Overall
OFC             0.305137722
ATR_fa          0.115171477
CST_fa          0.619097177
IFO_fa          0.003244446
VMI.beery       0.227187475
VM.wj           0.127799922
FSIQ            0.768889097
ADHD_PRS0.001000 0.109051519
base_inatt      0.332364391
base_hi         1.493837846
Setting direction: controls < cases
Data: preds with 2 levels of this_data[, phen]: nonimp, imp.
Multi-class area under the curve: 0.834
```