N741 Spring 2018 - Homework 7

Homework 7 - DUE WED April 11, 2018

Rosemary Kinuthia

April 6, 2018

Homework 7

Use these variables from HELP dataset for Homework 07

	Variable Label
age	Age at baseline (in years)
female	Gender of respondent
pss_fr	Perceived Social Support - friends
homeless	One or more nights on the street or shelter in past 6 months
pcs	SF36 Physical Composite Score - Baseline
mcs	SF36 Mental Composite Score - Baseline
cesd	CESD total score - Baseline
cesd_gte16	Indicator of Depression
mcs_lt45	Indicator of Poor Mental Health

Homework 7 Assignment

```
#Load packages needed for Homework 7
library(rpart)
library(partykit)
library(reshape2)
library(party)
library(tidyverse)
library(randomForestSRC)
library(ggRandomForests)
```

PROBLEM 1: Regression Tree for MCS

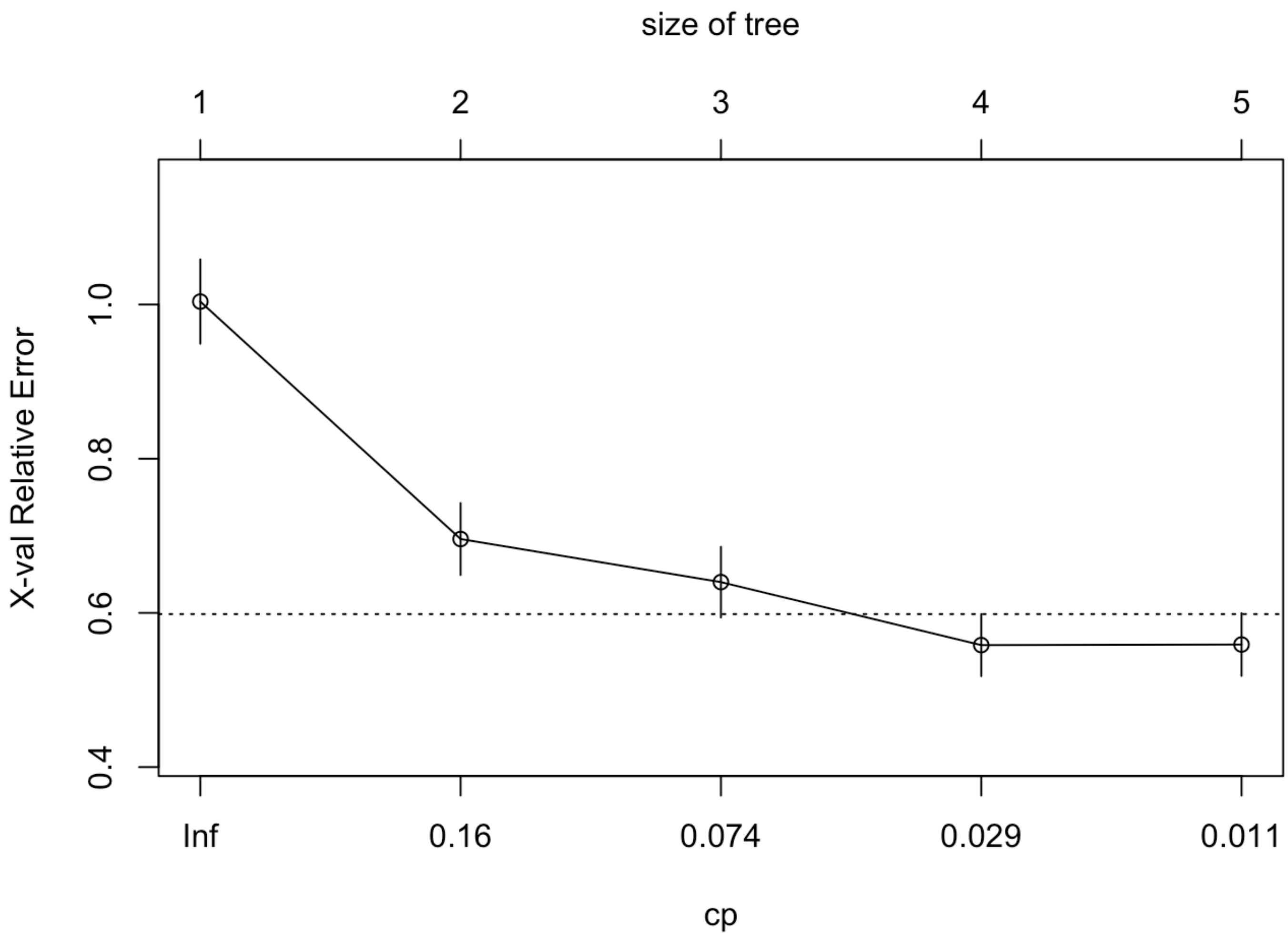
Using the code above, fit a regression tree model where the `mcs` is the outcome and the `cesd` is the predictor and complete the following:

- fit a regression tree to the `mcs` based on only the `cesd` scores from the `h1` dataset;
- display the results
- plot the cross-validated results
- provide a summary of the model fit
- and plot the regression tree

```
# fit a regression tree model to the mcs as the outcome
# and using the cesd as the only predictor
fitmcs <- rpart::rpart(mcs ~ cesd, data = h1)
rpart::printcp(fitmcs) # Display the results
```

```
##
## Regression tree:
## rpart::rpart(formula = mcs ~ cesd, data = h1)
##
## Variables actually used in tree construction:
## [1] cesd
##
## Root node error: 74512/453 = 164.48
##
## n= 453
##
##          CP nsplit rel error  xerror     xstd
## 1 0.325298      0   1.00000 1.00384 0.054675
## 2 0.081349      1   0.67470 0.69582 0.046808
## 3 0.066496      2   0.59335 0.63994 0.045806
## 4 0.012496      3   0.52686 0.55821 0.040217
## 5 0.010000      4   0.51436 0.55896 0.040694
```

```
rpart::plotcp(fitmcs) # Visualize cross-validation results
```

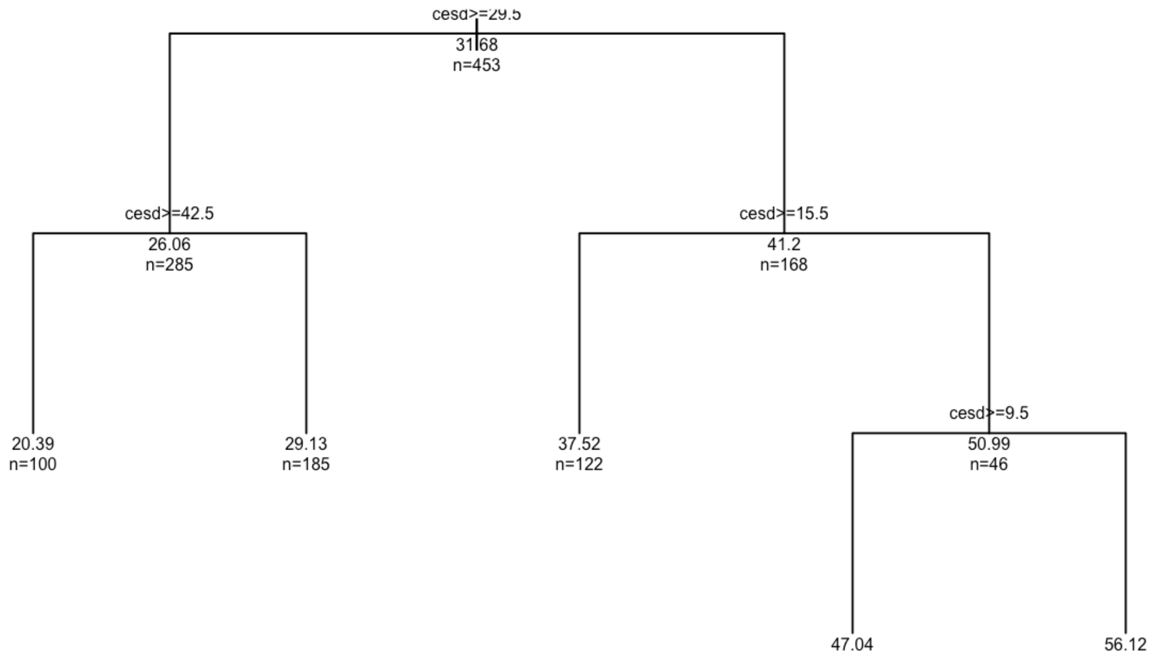



```
summary(fitmcs) # Detailed summary of fit
```

```
## Call:
## rpart::rpart(formula = mcs ~ cesd, data = h1)
##   n= 453
##
##           CP nsplit rel error    xerror       xstd
## 1 0.32529813      0 1.0000000 1.0038378 0.05467458
## 2 0.08134904      1 0.6747019 0.6958172 0.04680821
## 3 0.06649553      2 0.5933528 0.6399356 0.04580625
## 4 0.01249609      3 0.5268573 0.5582119 0.04021732
## 5 0.01000000      4 0.5143612 0.5589562 0.04069373
##
## Variable importance
## cesd
##  100
##
## Node number 1: 453 observations,    complexity param=0.3252981
##   mean=31.67668, MSE=164.4847
```

```
##   left son=2 (285 obs) right son=3 (168 obs)
##   Primary splits:
##       cesd < 29.5 to the right, improve=0.3252981, (0 missing)
##
## Node number 2: 285 observations,    complexity param=0.06649553
##   mean=26.06057, MSE=100.1894
##   left son=4 (100 obs) right son=5 (185 obs)
##   Primary splits:
##       cesd < 42.5 to the right, improve=0.17352, (0 missing)
##
## Node number 3: 168 observations,    complexity param=0.08134904
##   mean=41.20401, MSE=129.2805
##   left son=6 (122 obs) right son=7 (46 obs)
##   Primary splits:
##       cesd < 15.5 to the right, improve=0.2790834, (0 missing)
##
## Node number 4: 100 observations
##   mean=20.38941, MSE=43.95751
##
## Node number 5: 185 observations
##   mean=29.12606, MSE=103.8029
##
## Node number 6: 122 observations
##   mean=37.51566, MSE=103.6988
##
## Node number 7: 46 observations,    complexity param=0.01249609
##   mean=50.98616, MSE=65.35702
##   left son=14 (26 obs) right son=15 (20 obs)
##   Primary splits:
##       cesd < 9.5  to the right, improve=0.3097046, (0 missing)
##
## Node number 14: 26 observations
##   mean=47.04024, MSE=67.29195
##
## Node number 15: 20 observations
##   mean=56.11586, MSE=16.28645
```

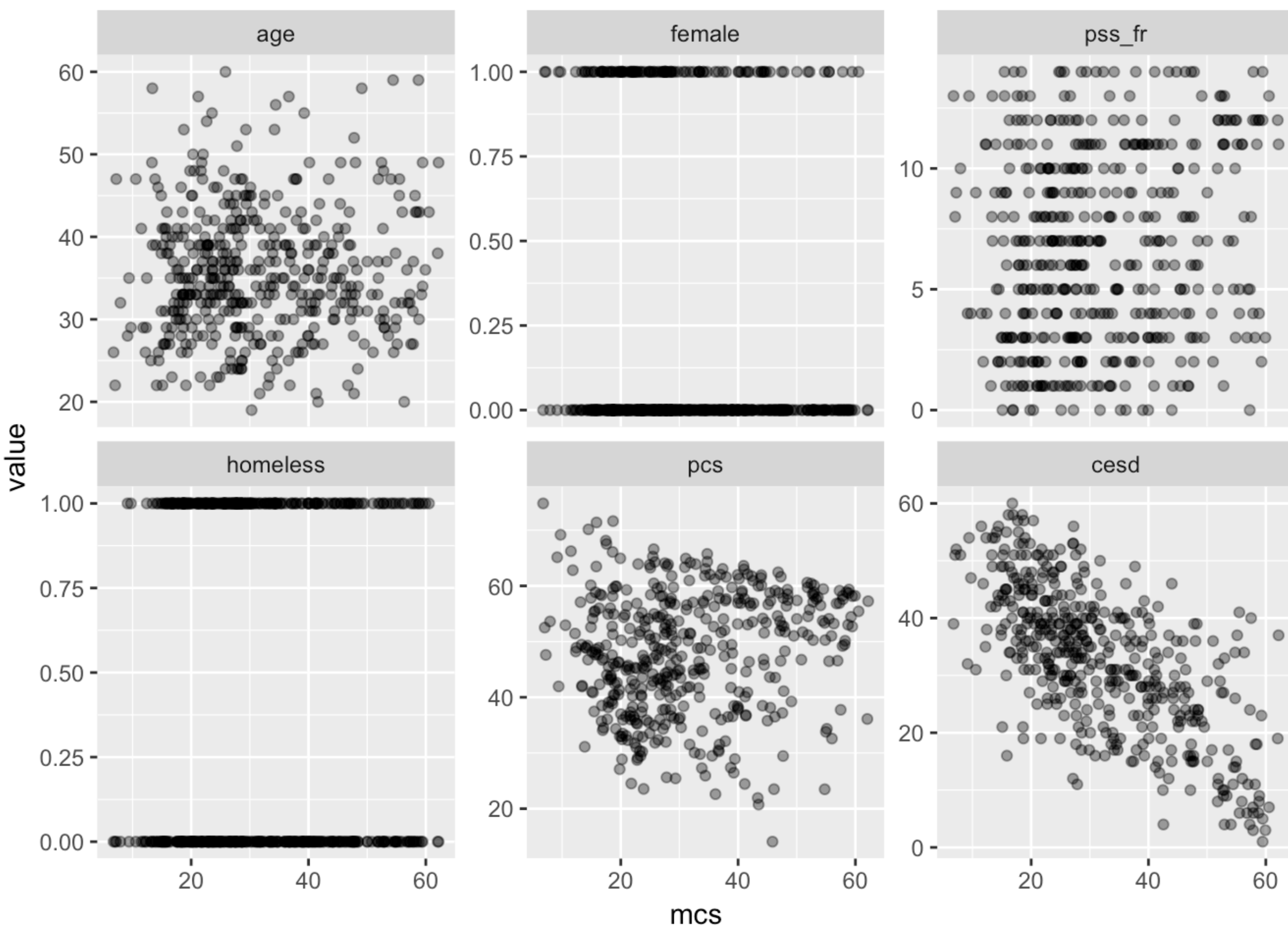
```
# plot tree
plot(fitmcs, uniform = TRUE, compress = FALSE)
text(fitmcs, use.n = TRUE, all = TRUE, cex = 0.5)
```



PROBLEM 2: Matrix Scatterplot of Other Variables with MCS

Using the code above as a guide,swap out `mcs` for `cesd` and redo the scatterplots compared to the `mcs`. HINT: You can begin with the data subset `h1a`, but you will need to modify the code for `h1m` and for the `ggplot()` code lines.

```
# all vars except the dichotomous cesd_gte16 and mcs_lt45
h1a <- h1[,1:7]

# Melt the other variables down and link to mcs
h1m <- reshape2::melt(h1a, id.vars = "mcs")

# Plot panels for each covariate
ggplot(h1m, aes(x=mcs, y=value)) +
  geom_point(alpha=0.4)+
  scale_color_brewer(palette="Set2")+
  facet_wrap(~variable, scales="free_y", ncol=3)
```



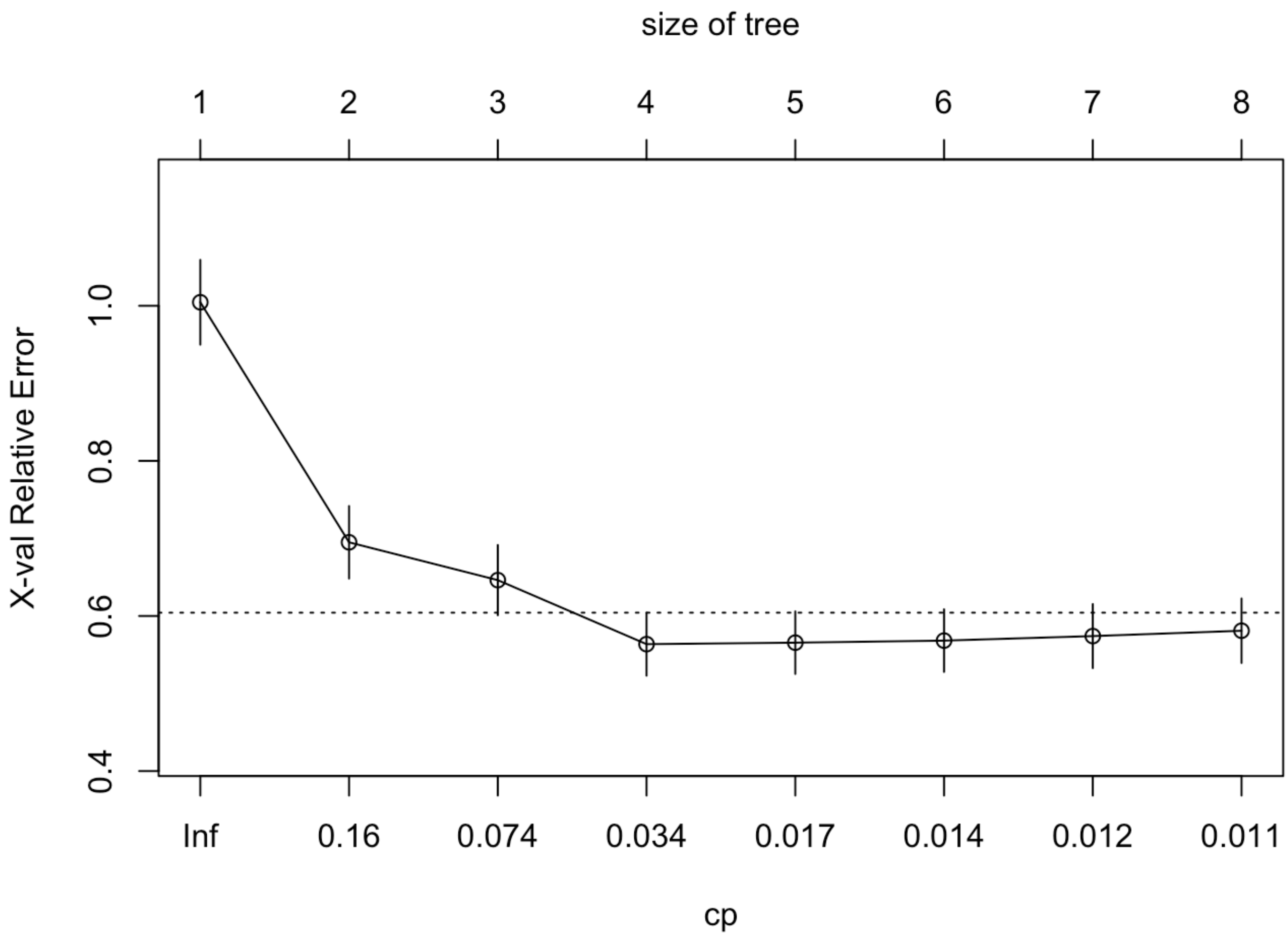
PROBLEM 3: Regression Tree for MCS Using Rest of Variables

Using the code above as a guide, swap out `mcs` for `cesd` and redo the regression tree for `mcs` using the rest of the variables in the data subset `h1a`.

```
# fit a regression tree with all vars
fitall <- rpart::rpart(mcs ~ ., data = h1a)

# equivalent code statement without the shorthand
# using the period for the "rest of the variables"
# this time each variable to be included is listed
# individually putting a plus + in between each
# variable added to the model

fitall <- rpart::rpart(mcs~ age + female + pss_fr +
                              homeless + pcs + cesd,
                              data = h1a)

# Now let's look at fitall
rpart::printcp(fitall) # Display the results
```

```
##
## Regression tree:
## rpart::rpart(formula = mcs ~ age + female + pss_fr + homeless +
##     pcs + cesd, data = h1a)
##
## Variables actually used in tree construction:
## [1] cesd pcs
##
## Root node error: 74512/453 = 164.48
##
## n= 453
##
##         CP nsplit rel error  xerror     xstd
## 1 0.325298      0   1.00000 1.00455 0.054665
## 2 0.081349      1   0.67470 0.69502 0.046745
## 3 0.066496      2   0.59335 0.64622 0.045287
## 4 0.017717      3   0.52686 0.56366 0.040613
## 5 0.015767      4   0.50914 0.56559 0.040336
## 6 0.012496      5   0.49337 0.56824 0.040370
## 7 0.012258      6   0.48088 0.57406 0.041279
## 8 0.010000      7   0.46862 0.58098 0.041580
```

```
rpart::plotcp(fitall) # Visualize cross-validation results
```




```r
summary(fitall) # Detailed summary of fit
```

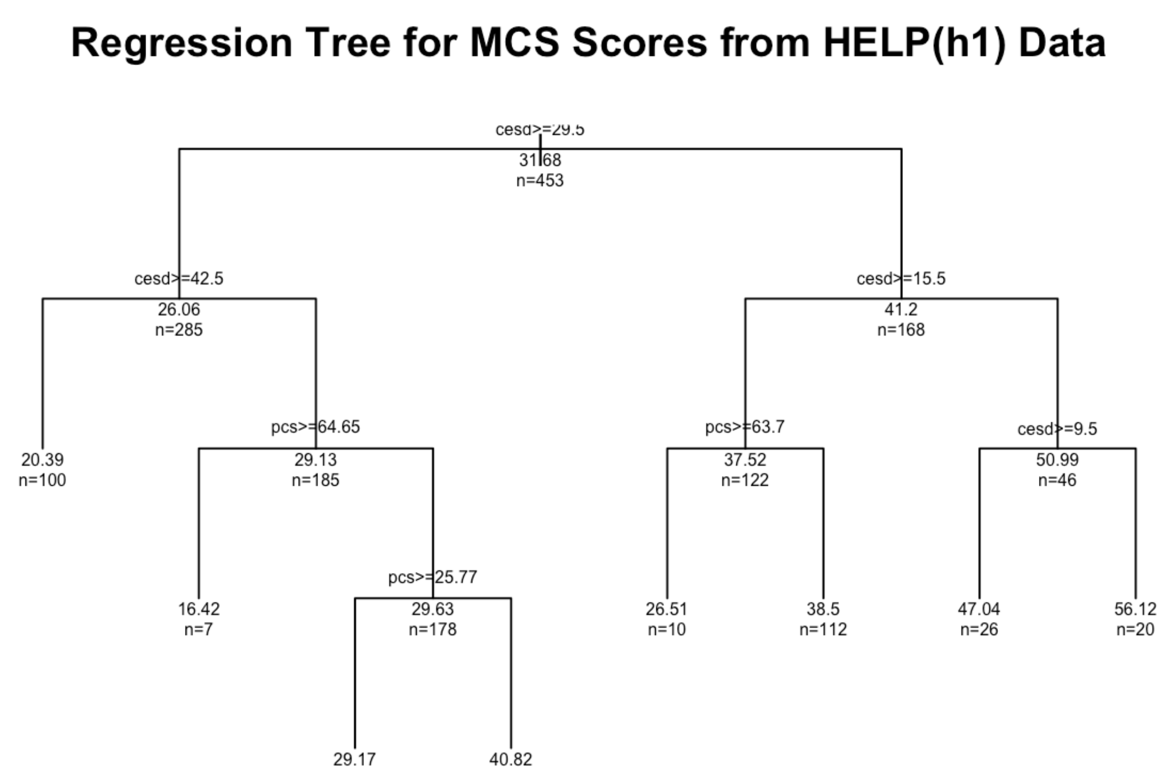
```
## Call:
## rpart::rpart(formula = mcs ~ age + female + pss_fr + homeless +
##     pcs + cesd, data = h1a)
##   n= 453
##
##           CP nsplit rel error    xerror       xstd
## 1 0.32529813      0 1.0000000 1.0045478 0.05466529
## 2 0.08134904      1 0.6747019 0.6950200 0.04674484
## 3 0.06649553      2 0.5933528 0.6462180 0.04528689
## 4 0.01771736      3 0.5268573 0.5636589 0.04061322
## 5 0.01576737      4 0.5091399 0.5655929 0.04033625
## 6 0.01249609      5 0.4933726 0.5682405 0.04037050
## 7 0.01225792      6 0.4808765 0.5740602 0.04127898
## 8 0.01000000      7 0.4686186 0.5809767 0.04157976
##
## Variable importance
##   cesd    pcs    age pss_fr
```

```
##     83     14      1      1
##
## Node number 1: 453 observations,    complexity param=0.3252981
##   mean=31.67668, MSE=164.4847
##   left son=2 (285 obs) right son=3 (168 obs)
##   Primary splits:
##       cesd   < 29.5     to the right, improve=0.325298100, (0 missing)
##       pcs    < 49.46132 to the left,  improve=0.064711670, (0 missing)
##       pss_fr < 10.5     to the left,  improve=0.039318510, (0 missing)
##       female < 0.5      to the right, improve=0.014091560, (0 missing)
##       age    < 42.5     to the left,  improve=0.005473724, (0 missing)
##   Surrogate splits:
##       pcs < 56.34591 to the left,  agree=0.669, adj=0.107, (0 split)
##       age < 57.5     to the left,  agree=0.631, adj=0.006, (0 split)
##
## Node number 2: 285 observations,    complexity param=0.06649553
##   mean=26.06057, MSE=100.1894
##   left son=4 (100 obs) right son=5 (185 obs)
##   Primary splits:
##       cesd   < 42.5     to the right, improve=0.173520000, (0 missing)
##       pcs    < 24.47511 to the right, improve=0.057879990, (0 missing)
##       pss_fr < 10.5     to the left,  improve=0.015219690, (0 missing)
##       age    < 22.5     to the right, improve=0.005742931, (0 missing)
##       female < 0.5      to the right, improve=0.001903900, (0 missing)
##   Surrogate splits:
##       pss_fr < 0.5      to the left,  agree=0.660, adj=0.03, (0 split)
##       pcs    < 68.64778 to the right, agree=0.653, adj=0.01, (0 split)
##
## Node number 3: 168 observations,    complexity param=0.08134904
##   mean=41.20401, MSE=129.2805
##   left son=6 (122 obs) right son=7 (46 obs)
##   Primary splits:
##       cesd   < 15.5     to the right, improve=0.279083400, (0 missing)
##       pcs    < 62.7532  to the right, improve=0.113215200, (0 missing)
##       pss_fr < 10.5     to the left,  improve=0.053187210, (0 missing)
##       age    < 48.5     to the left,  improve=0.036737610, (0 missing)
##       female < 0.5      to the right, improve=0.007177787, (0 missing)
##   Surrogate splits:
##       age < 58.5     to the left,  agree=0.738, adj=0.043, (0 split)
##
## Node number 4: 100 observations
##   mean=20.38941, MSE=43.95751
##
## Node number 5: 185 observations,    complexity param=0.01576737
##   mean=29.12606, MSE=103.8029
##   left son=10 (7 obs) right son=11 (178 obs)
##   Primary splits:
```

```
##       pcs    < 64.65134 to the right, improve=0.061178900, (0 missing)
##       age    < 22.5     to the right, improve=0.031248410, (0 missing)
##       cesd   < 37.5     to the right, improve=0.020833690, (0 missing)
##       pss_fr < 10.5     to the left,  improve=0.015175680, (0 missing)
##       female < 0.5      to the left,  improve=0.004355548, (0 missing)
##
## Node number 6: 122 observations,    complexity param=0.01771736
##   mean=37.51566, MSE=103.6988
##   left son=12 (10 obs) right son=13 (112 obs)
##   Primary splits:
##       pcs    < 63.69606 to the right, improve=0.10434930, (0 missing)
##       age    < 47.5     to the left,  improve=0.02626159, (0 missing)
##       cesd   < 24.5     to the right, improve=0.02348926, (0 missing)
##       female < 0.5      to the right, improve=0.02256241, (0 missing)
##       pss_fr < 2.5      to the right, improve=0.01295167, (0 missing)
##
## Node number 7: 46 observations,    complexity param=0.01249609
##   mean=50.98616, MSE=65.35702
##   left son=14 (26 obs) right son=15 (20 obs)
##   Primary splits:
##       cesd     < 9.5      to the right, improve=0.30970460, (0 missing)
##       pcs      < 59.57495 to the right, improve=0.16249370, (0 missing)
##       pss_fr   < 11.5     to the left,  improve=0.13099300, (0 missing)
##       age      < 40       to the left,  improve=0.06604375, (0 missing)
##       homeless < 0.5      to the left,  improve=0.00873942, (0 missing)
##   Surrogate splits:
##       pss_fr   < 11.5     to the left,  agree=0.674, adj=0.25, (0 split)
##       pcs      < 54.5861  to the left,  agree=0.652, adj=0.20, (0 split)
##       age      < 46       to the left,  agree=0.609, adj=0.10, (0 split)
##       homeless < 0.5      to the left,  agree=0.609, adj=0.10, (0 split)
##
## Node number 10: 7 observations
##   mean=16.41837, MSE=35.31025
##
## Node number 11: 178 observations,    complexity param=0.01225792
##   mean=29.6258, MSE=99.89614
##   left son=22 (171 obs) right son=23 (7 obs)
##   Primary splits:
##       pcs      < 25.77119 to the right, improve=0.051365510, (0 missing)
##       age      < 22.5     to the right, improve=0.029936490, (0 missing)
##       pss_fr   < 10.5     to the left,  improve=0.022699840, (0 missing)
##       cesd     < 37.5     to the right, improve=0.020642200, (0 missing)
##       homeless < 0.5      to the right, improve=0.002448012, (0 missing)
##
## Node number 12: 10 observations
##   mean=26.50685, MSE=30.97799
##
```

```
## Node number 13: 112 observations
##   mean=38.49859, MSE=98.40465
##
## Node number 14: 26 observations
##   mean=47.04024, MSE=67.29195
##
## Node number 15: 20 observations
##   mean=56.11586, MSE=16.28645
##
## Node number 22: 171 observations
##   mean=29.16748, MSE=95.51594
##
## Node number 23: 7 observations
##   mean=40.8217, MSE=76.41866
```

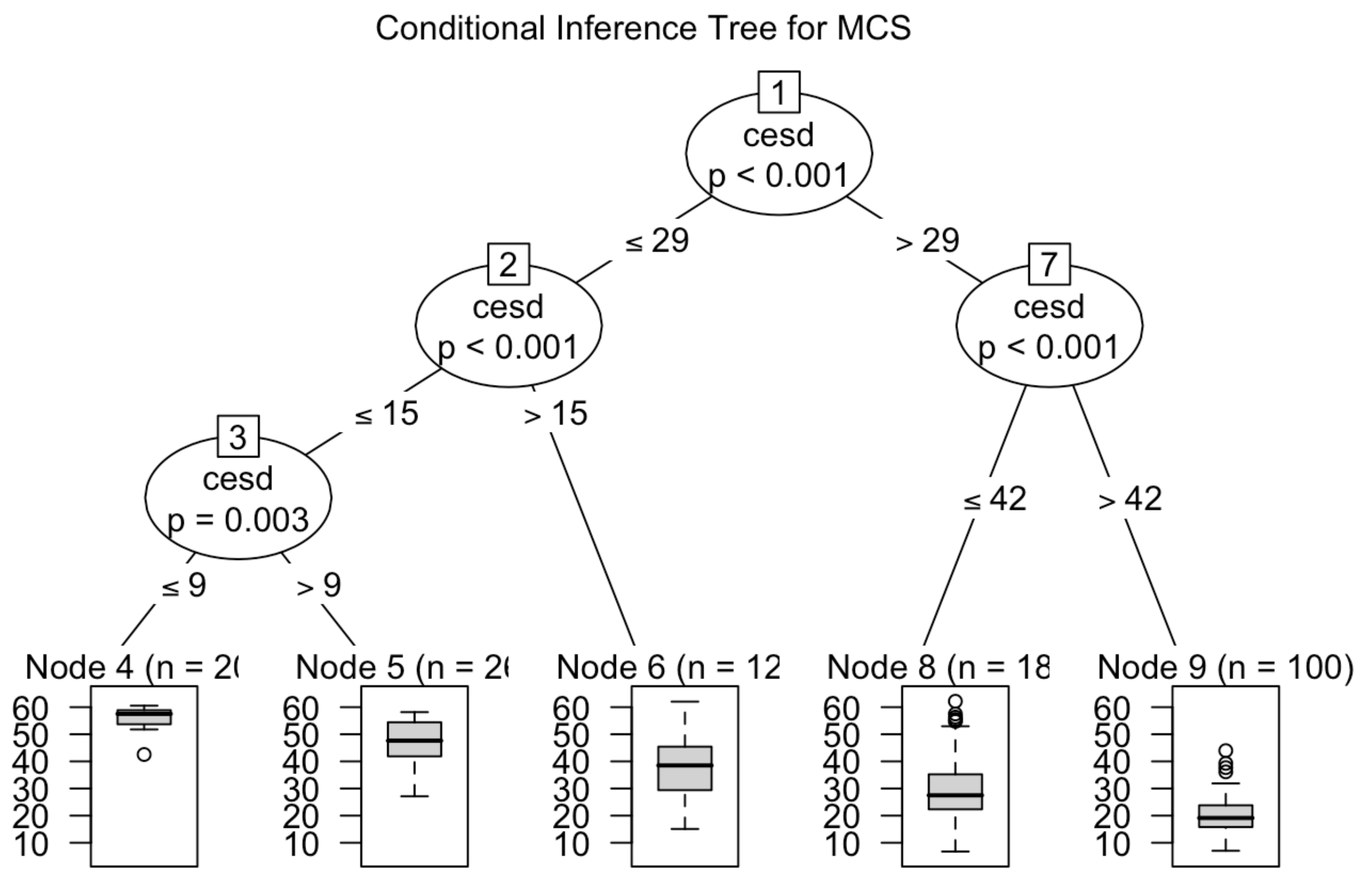
```
plot(fitall, uniform = TRUE, compress = FALSE, main = "Regression Tree for MCS Scores
from HELP(h1) Data")
text(fitall, use.n = TRUE, all = TRUE, cex = 0.5)
```



PROBLEM 4: Fit a Conditional Regression Tree for MCS

Using the code above, swap out `mcs` for `cesd` to fit a confitional regression tree for `mcs` predicted by the other variables in the dataset `h1a` .

```
fitallp <- party::ctree(mcs ~ ., data = h1a)
plot(fitallp, main = "Conditional Inference Tree for MCS")
```



PROBLEM 5: Fit a Logistic Regression Model for MCS < 45

The mental component (or composite) scale of the SF36 instrument is a measure of mental health. The scores are created relative to population norms. The population norm for the `mcs` of the SF36 is 50 with a standard deviation of 10. A difference of a "half" of a standard deviation - in other words a difference of 5 points - is considered to be clinically meaningful. So, people with MCS scores greater than 55 are considered to have better than average mental health and those with MCS scores less than 45 are considered to have worse than average mental health scores. So, in the dataset `h1` above, we included an indicator variable called `mcs_lt45` where a value of 1 indicates people with MCS < 45 ("poor mental health") and a value of 0 ("normal or better than normal mental health") is for people with MCS scores => 45.

Use the dataset `h1` and the code above to fit a logistic regression model for `mcs_lt45` based on the predictors of

- `age`
- `female`
- `pss_fr`
- `homeless`
- `pcs`
- `cesd`

Is this model similar to the model for `cesd_gte16` or not - what is similar? what is different?

```
# begin with a logistic regression - poor mental health or not (normal or better than
normal mental health)
glm1 <- glm(mcs_lt45 ~ age + female + pss_fr + homeless +
              pcs + cesd, data = h1)
summary(glm1)
```

```
##
## Call:
## glm(formula = mcs_lt45 ~ age + female + pss_fr + homeless + pcs +
##     cesd, data = h1)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.96035  -0.10332   0.08078   0.21806   0.62498
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.3611168  0.1386939   2.604  0.00953 **
## age         -0.0023080  0.0021130  -1.092  0.27529
## female       0.0202380  0.0382212   0.529  0.59672
## pss_fr      -0.0036606  0.0040882  -0.895  0.37104
## homeless     0.0172706  0.0323939   0.533  0.59420
## pcs          0.0005446  0.0015809   0.344  0.73064
## cesd         0.0158725  0.0013519  11.741  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1114291)
##
##     Null deviance: 68.424  on 452  degrees of freedom
## Residual deviance: 49.697  on 446  degrees of freedom
## AIC: 300.46
##
## Number of Fisher Scoring iterations: 2
```

```
#The mcs_lt45 model does appear to be different from the cesd_gte16. In the cesd_gte1
6 model all the the predictors lower the cesd. However, only pcs and mcs are statisti
cally significant. In the mcs_lt45 model, with the exception of age and pss_fr which
lower the mcl, the other predictor variables increase the mcs_lt45, and only cesd is
statistically significant.
```

PROBLEM 6: Fit a Classification Tree for MCS < 45

Use the `rpart` package to fit a classification tree to the poor mental health indicator `mcs_lt45`.

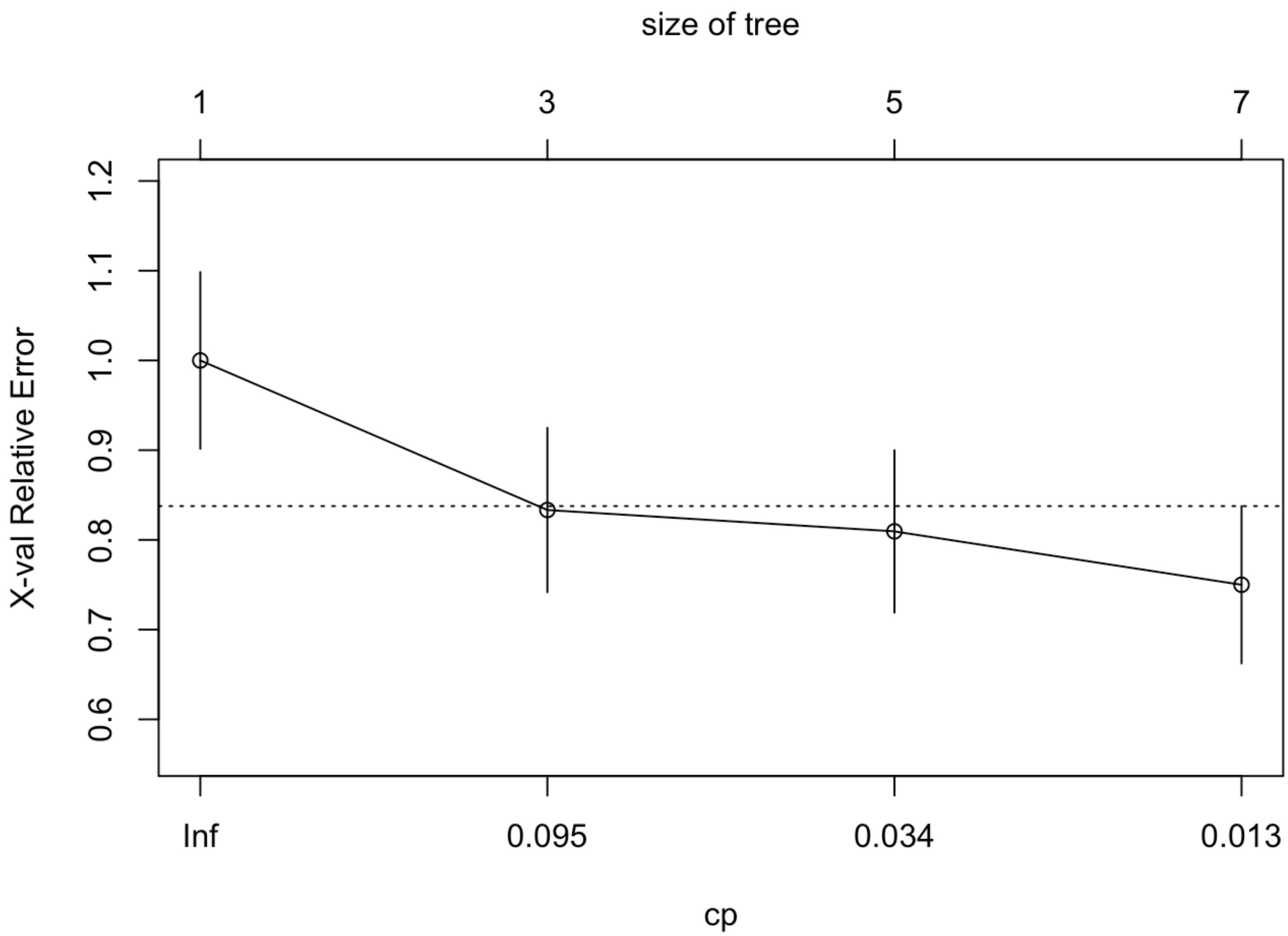
```
fitk <- rpart::rpart(mcs_lt45 ~ age + female + pss_fr +
                       homeless + pcs + cesd,
                     method = "class", data = h1)
class(fitk)
```

```
## [1] "rpart"
```

```
# Display the results
rpart::printcp(fitk)
```

```
##
## Classification tree:
## rpart::rpart(formula = mcs_lt45 ~ age + female + pss_fr + homeless +
##     pcs + cesd, data = h1, method = "class")
##
## Variables actually used in tree construction:
## [1] age  cesd pcs
##
## Root node error: 84/453 = 0.18543
##
## n= 453
##
##         CP nsplit rel error  xerror     xstd
## 1 0.136905      0   1.00000 1.00000 0.098475
## 2 0.065476      2   0.72619 0.83333 0.091584
## 3 0.017857      4   0.59524 0.80952 0.090502
## 4 0.010000      6   0.55952 0.75000 0.087675
```

```
#Visualize the cross-validation results
rpart::plotcp(fitk)
```




```
# Get a detailed summary of the splits
summary(fitk)
```

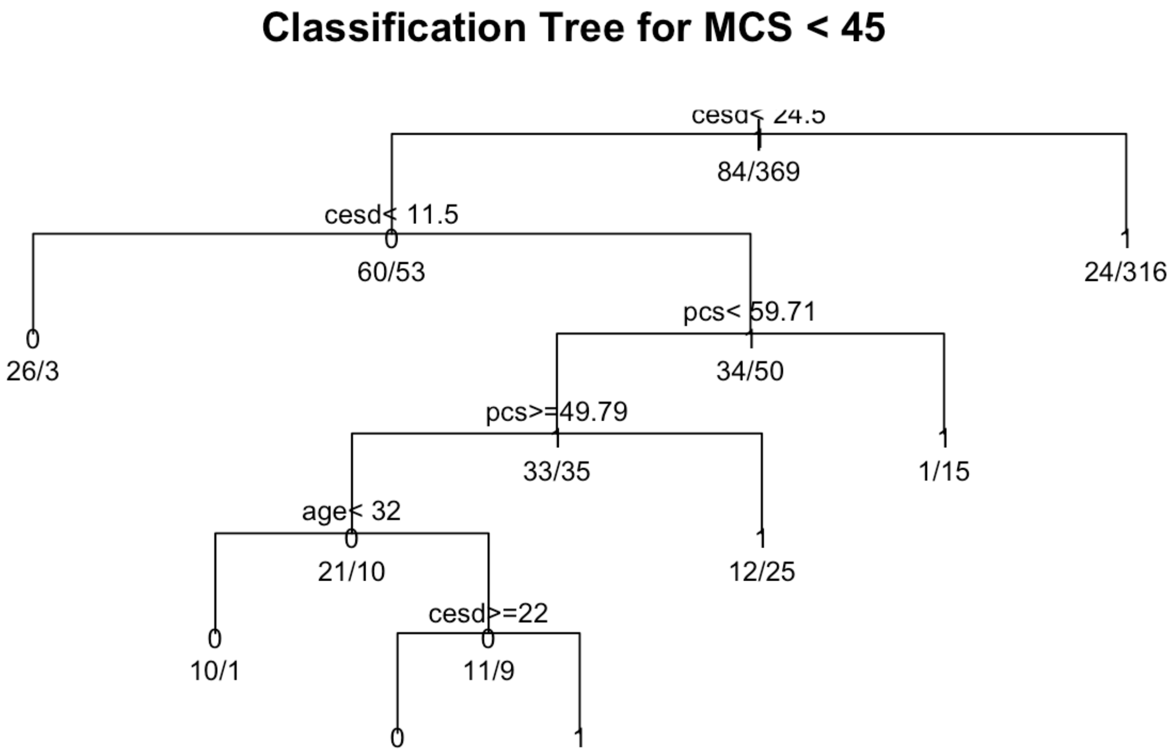
```
## Call:
## rpart::rpart(formula = mcs_lt45 ~ age + female + pss_fr + homeless +
##     pcs + cesd, data = h1, method = "class")
##   n= 453
##
##           CP nsplit rel error    xerror       xstd
## 1 0.13690476      0 1.0000000 1.0000000 0.09847465
## 2 0.06547619      2 0.7261905 0.8333333 0.09158409
## 3 0.01785714      4 0.5952381 0.8095238 0.09050164
## 4 0.01000000      6 0.5595238 0.7500000 0.08767468
##
## Variable importance
##   cesd    pcs    age pss_fr
##     76     16      6      2
##
## Node number 1: 453 observations,    complexity param=0.1369048
```

```
##   predicted class=1  expected loss=0.1854305  P(node) =1
##     class counts:    84   369
##    probabilities: 0.185 0.815
##   left son=2 (113 obs) right son=3 (340 obs)
##   Primary splits:
##       cesd   < 24.5     to the left,  improve=35.952730, (0 missing)
##       pcs    < 49.46132 to the right, improve= 7.907014, (0 missing)
##       pss_fr < 10.5     to the right, improve= 4.386206, (0 missing)
##       female < 0.5      to the left,  improve= 1.504589, (0 missing)
##       age    < 48.5     to the right, improve= 1.425056, (0 missing)
##   Surrogate splits:
##       age < 57.5     to the right, agree=0.753, adj=0.009, (0 split)
##       pcs < 70.77019 to the right, agree=0.753, adj=0.009, (0 split)
##
## Node number 2: 113 observations,    complexity param=0.1369048
##   predicted class=0  expected loss=0.4690265  P(node) =0.2494481
##     class counts:    60    53
##    probabilities: 0.531 0.469
##   left son=4 (29 obs) right son=5 (84 obs)
##   Primary splits:
##       cesd   < 11.5     to the left,  improve=10.427690, (0 missing)
##       pcs    < 60.7539  to the left,  improve= 8.921666, (0 missing)
##       pss_fr < 11.5     to the right, improve= 2.105364, (0 missing)
##       female < 0.5      to the left,  improve= 1.591788, (0 missing)
##       age    < 47.5     to the right, improve= 1.587768, (0 missing)
##   Surrogate splits:
##       age < 58.5     to the right, agree=0.761, adj=0.069, (0 split)
##
## Node number 3: 340 observations
##   predicted class=1  expected loss=0.07058824  P(node) =0.7505519
##     class counts:    24   316
##    probabilities: 0.071 0.929
##
## Node number 4: 29 observations
##   predicted class=0  expected loss=0.1034483  P(node) =0.06401766
##     class counts:    26     3
##    probabilities: 0.897 0.103
##
## Node number 5: 84 observations,    complexity param=0.06547619
##   predicted class=1  expected loss=0.4047619  P(node) =0.1854305
##     class counts:    34    50
##    probabilities: 0.405 0.595
##   left son=10 (68 obs) right son=11 (16 obs)
##   Primary splits:
##       pcs    < 59.71077 to the left,  improve=4.6306020, (0 missing)
##       female < 0.5      to the left,  improve=1.8658960, (0 missing)
##       cesd   < 21.5     to the right, improve=1.7155130, (0 missing)
```

```
##       age    < 38.5     to the left,  improve=0.2586838, (0 missing)
##       pss_fr < 11.5     to the right, improve=0.2539683, (0 missing)
##
## Node number 10: 68 observations,    complexity param=0.06547619
##   predicted class=1  expected loss=0.4852941  P(node) =0.1501104
##     class counts:    33    35
##    probabilities: 0.485 0.515
##   left son=20 (31 obs) right son=21 (37 obs)
##   Primary splits:
##       pcs    < 49.7901  to the right, improve=4.2059850, (0 missing)
##       female < 0.5      to the left,  improve=2.0824760, (0 missing)
##       cesd   < 16.5     to the left,  improve=1.1284830, (0 missing)
##       age    < 43.5     to the left,  improve=0.4790628, (0 missing)
##       pss_fr < 7.5      to the left,  improve=0.2761438, (0 missing)
##   Surrogate splits:
##       age      < 27.5     to the left,  agree=0.588, adj=0.097, (0 split)
##       pss_fr   < 13.5     to the right, agree=0.588, adj=0.097, (0 split)
##       homeless < 0.5      to the right, agree=0.559, adj=0.032, (0 split)
##       cesd     < 12.5     to the left,  agree=0.559, adj=0.032, (0 split)
##
## Node number 11: 16 observations
##   predicted class=1  expected loss=0.0625  P(node) =0.03532009
##     class counts:     1    15
##    probabilities: 0.062 0.938
##
## Node number 20: 31 observations,    complexity param=0.01785714
##   predicted class=0  expected loss=0.3225806  P(node) =0.06843267
##     class counts:    21    10
##    probabilities: 0.677 0.323
##   left son=40 (11 obs) right son=41 (20 obs)
##   Primary splits:
##       age      < 32       to the left,  improve=1.830205000, (0 missing)
##       pss_fr   < 8.5      to the left,  improve=1.607211000, (0 missing)
##       cesd     < 21.5     to the right, improve=1.462673000, (0 missing)
##       pcs      < 57.31713 to the right, improve=1.274883000, (0 missing)
##       homeless < 0.5      to the left,  improve=0.004527448, (0 missing)
##   Surrogate splits:
##       pcs  < 59.00035 to the right, agree=0.710, adj=0.182, (0 split)
##       cesd < 16.5     to the left,  agree=0.677, adj=0.091, (0 split)
##
## Node number 21: 37 observations
##   predicted class=1  expected loss=0.3243243  P(node) =0.0816777
##     class counts:    12    25
##    probabilities: 0.324 0.676
##
## Node number 40: 11 observations
##   predicted class=0  expected loss=0.09090909  P(node) =0.02428256
```

```
##     class counts:    10     1
##    probabilities: 0.909 0.091
##
## Node number 41: 20 observations,    complexity param=0.01785714
##   predicted class=0  expected loss=0.45  P(node) =0.04415011
##     class counts:    11     9
##    probabilities: 0.550 0.450
##   left son=82 (7 obs) right son=83 (13 obs)
##   Primary splits:
##       cesd     < 22       to the right, improve=2.0318680, (0 missing)
##       age      < 39.5     to the right, improve=0.5813187, (0 missing)
##       pcs      < 57.14784 to the right, improve=0.5813187, (0 missing)
##       pss_fr   < 7        to the left,  improve=0.4454545, (0 missing)
##       homeless < 0.5      to the right, improve=0.1000000, (0 missing)
##   Surrogate splits:
##       pss_fr < 7        to the left,  agree=0.80, adj=0.429, (0 split)
##       age    < 45       to the right, agree=0.75, adj=0.286, (0 split)
##       pcs    < 57.34769 to the right, agree=0.75, adj=0.286, (0 split)
##
## Node number 82: 7 observations
##   predicted class=0  expected loss=0.1428571  P(node) =0.01545254
##     class counts:     6     1
##    probabilities: 0.857 0.143
##
## Node number 83: 13 observations
##   predicted class=1  expected loss=0.3846154  P(node) =0.02869757
##     class counts:     5     8
##    probabilities: 0.385 0.615
```

```
# Plot the tree
plot(fitk, uniform = TRUE,
     main = "Classification Tree for MCS < 45 ")
text(fitk, use.n = TRUE, all = TRUE, cex = 0.8)
```



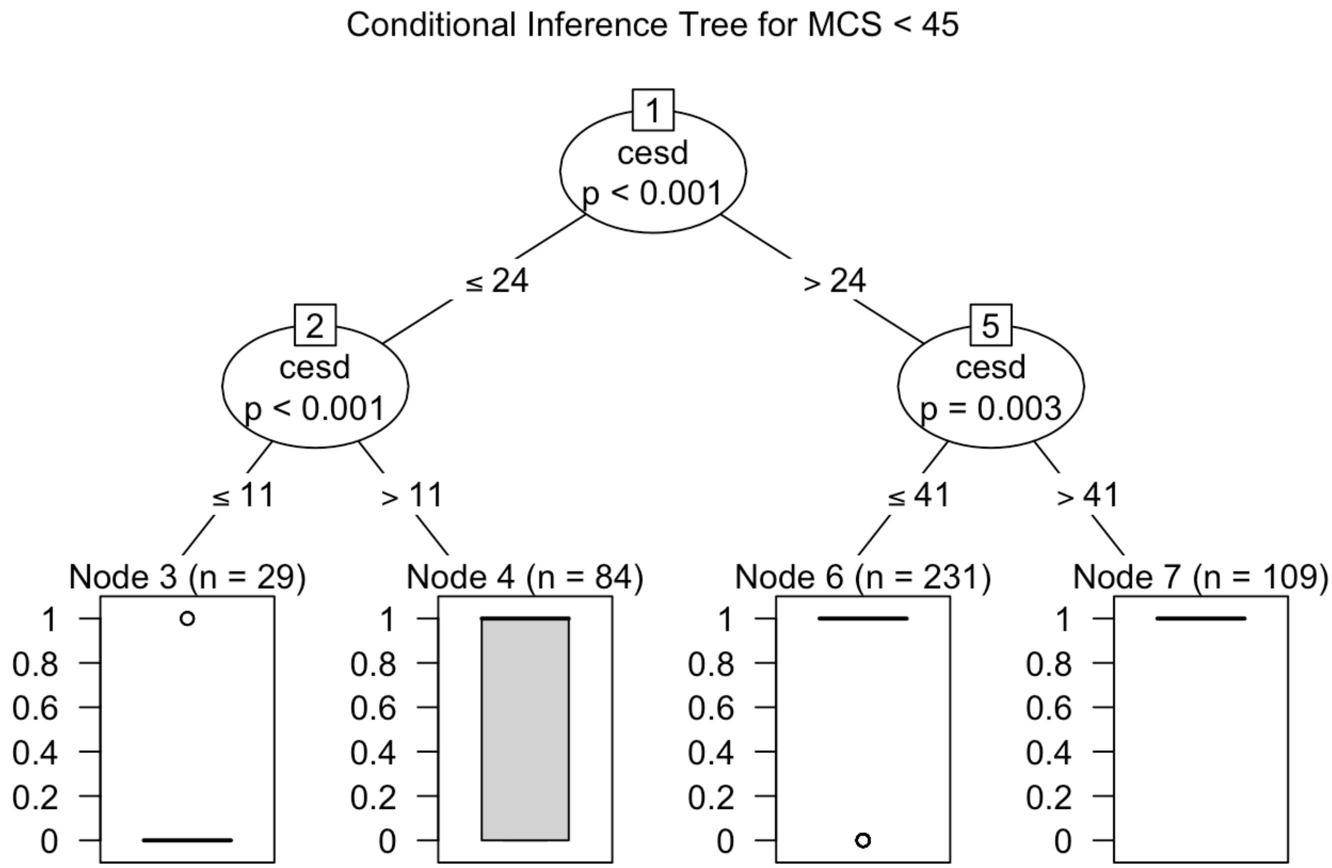
PROBLEM 7: Fit a Conditional Classification Tree for MCS < 45

Using the `party` package, we can fit a conditional classification tree using the `ctree()` function. Let's do one for the indicator of depression `mcs_lt45` given the other variables in the `h1` dataset: `age`, `female`, `pss_fr`, `homeless`, `pcs`, `cesd`.

```
# look at mcs_lt45 with ctree from party
fitallpk <- party::ctree(mcs_lt45 ~ age + female + pss_fr +
                           homeless + pcs + cesd, data = h1)
class(fitallpk)
```

```
## [1] "BinaryTree"
## attr(,"package")
## [1] "party"
```

```
plot(fitallpk, main = "Conditional Inference Tree for MCS < 45")
```



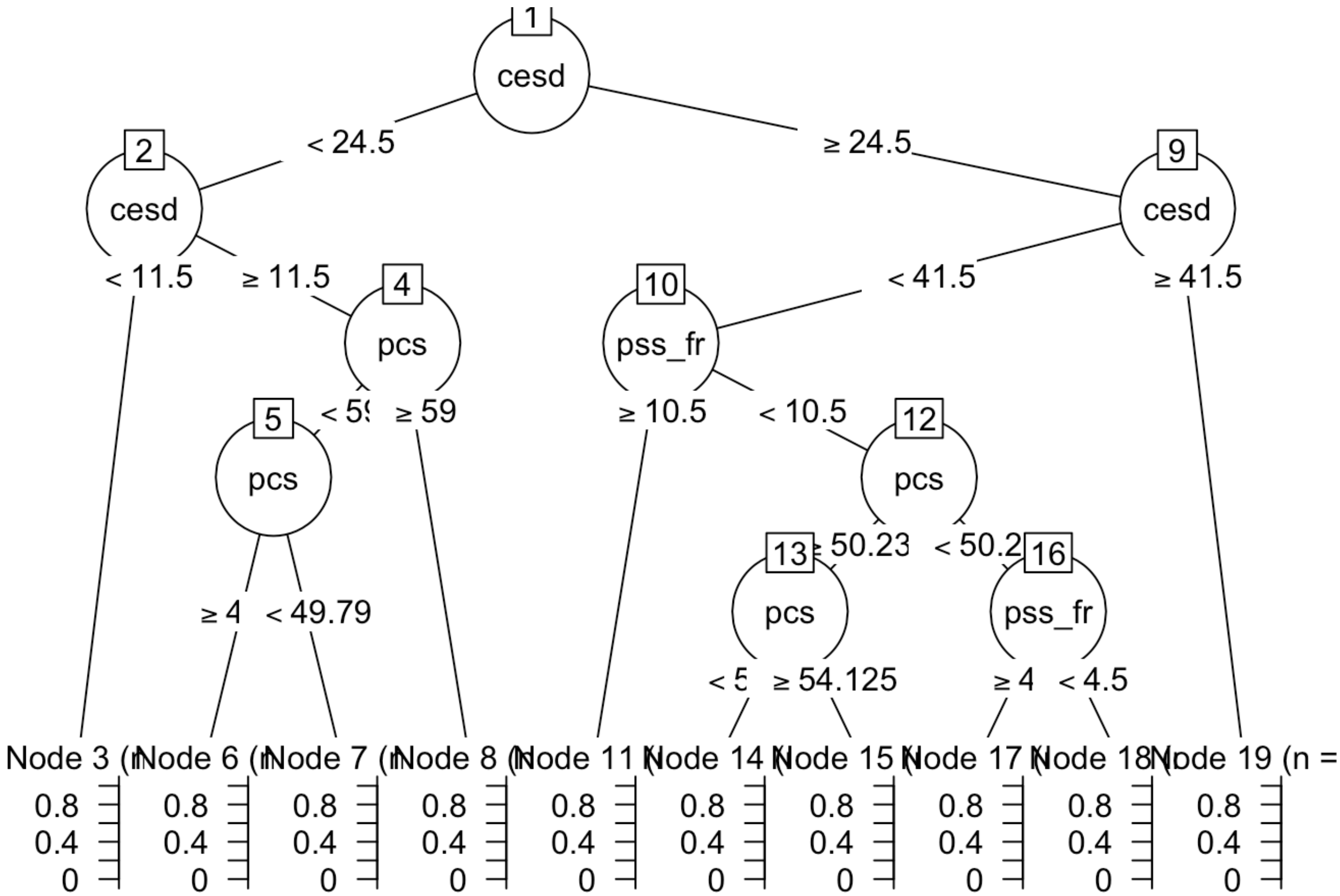
PROBLEM 8: Recursive Partitioning of Classification Tree for MCS < 45

Using the code above to do recursive partitioning of MCS < 45 (`mcs_lt45`) on `age` , `female` , `pss_fr` , `homeless` , `pcs` , `cesd` . Also use the `partykit` package to get prettier graphics for this classification tree.

```
# Recursive partitioning of MCS < 45 on age,
# female, pss_fr, homeless, pcs, cesd
WhoHasPoorMentalHealth <- rpart::rpart(mcs_lt45 ~ age + female +
                                pss_fr + homeless + pcs + cesd,
                              data = h1,
                              control = rpart.control(cp = 0.001,
                                                      minbucket = 20))

WhoHasPoorMentalHealth
```

```
## n= 453
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 453 68.423840 0.8145695
##    2) cesd< 24.5 113 28.141590 0.4690265
##      4) cesd< 11.5 29  2.689655 0.1034483 *
##      5) cesd>=11.5 84 20.238100 0.5952381
##       10) pcs< 59.00035 64 15.937500 0.5312500
##         20) pcs>=49.7901 27  6.000000 0.3333333 *
##         21) pcs< 49.7901 37  8.108108 0.6756757 *
##       11) pcs>=59.00035 20  3.200000 0.8000000 *
##    3) cesd>=24.5 340 22.305880 0.9294118
##      6) cesd< 41.5 231 21.506490 0.8961039
##       12) pss_fr>=10.5 52  8.076923 0.8076923 *
##       13) pss_fr< 10.5 179 12.905030 0.9217877
##         26) pcs>=50.23704 80  8.750000 0.8750000
##           52) pcs< 54.12466 25  4.560000 0.7600000 *
##           53) pcs>=54.12466 55  3.709091 0.9272727 *
##         27) pcs< 50.23704 99  3.838384 0.9595960
##           54) pss_fr>=4.5 55  3.709091 0.9272727 *
##           55) pss_fr< 4.5 44  0.000000 1.0000000 *
##      7) cesd>=41.5 109  0.000000 1.0000000 *
```

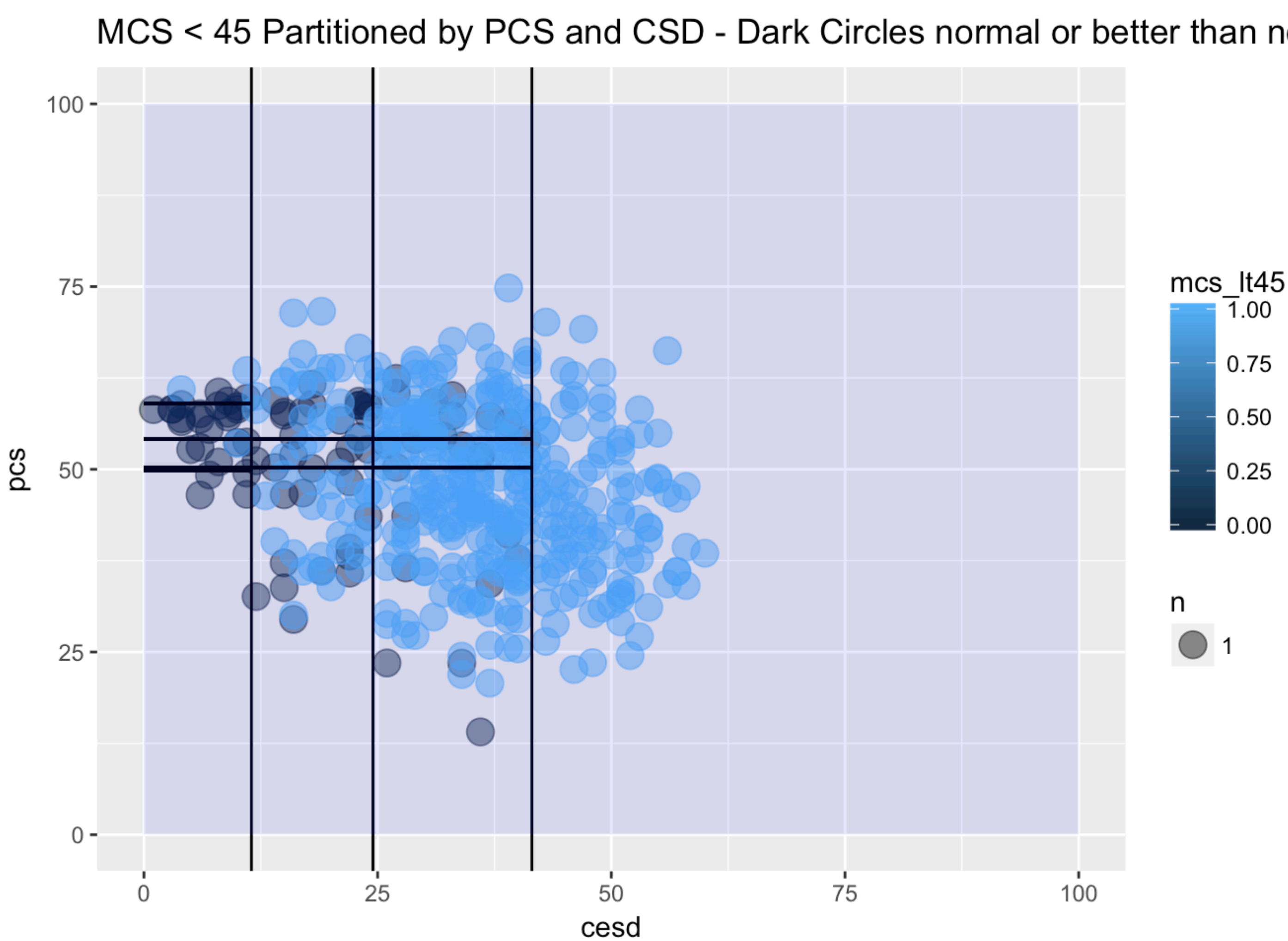
```
library(partykit)
# Plot the tree
plot(partykit::as.party(WhoHasPoorMentalHealth))
```



EXTRA CREDIT Scatterplot of recursive partitions for MCS < 45 for PCS and CESD

Using the code above, create a scatterplot of `pcs` and `cesd` where the points are colored by the indication of poor mental health `mcs_lt45`. Play with the `geom_vline()` or `geom_hline()` or `geom_segment()` to insert lines that best separate subjects with poor mental health (MCS < 45) from those with normal to better than average mental health (MCS > 45).

```
# Graph as partition
# using the break points shown from the
# conditional tree
ggplot(data = h1, aes(x = cesd, y = pcs)) +
  geom_count(aes(color = mcs_lt45), alpha = 0.5) +
  geom_vline(xintercept = 24.5) +
  geom_vline(xintercept = 11.5) +
  geom_vline(xintercept = 41.5) +
  geom_segment(x = 11.5, xend = 0, y = 59.00035, yend = 59.00035) +
  geom_segment(x = 11.5, xend = 0, y = 49.7901, yend = 49.7901) +
  geom_segment(x = 41.5, xend = 0, y = 50.23704, yend = 50.23704) +
  geom_segment(x = 41.5, xend = 0, y = 54.12466, yend = 54.12466)+
  annotate("rect", xmin = 0, xmax = 100, ymin = 0, ymax = 100, fill = "blue", alpha =
0.1) +
  ggtitle("MCS < 45 Partitioned by PCS and CSD - Dark Circles normal or better than n
ormal mental health")
```



PROBLEM 9: Fit a Random Forest Model for MCS

Now let's use a Random Forest approach for modeling the MCS by the other variables in the dataset:

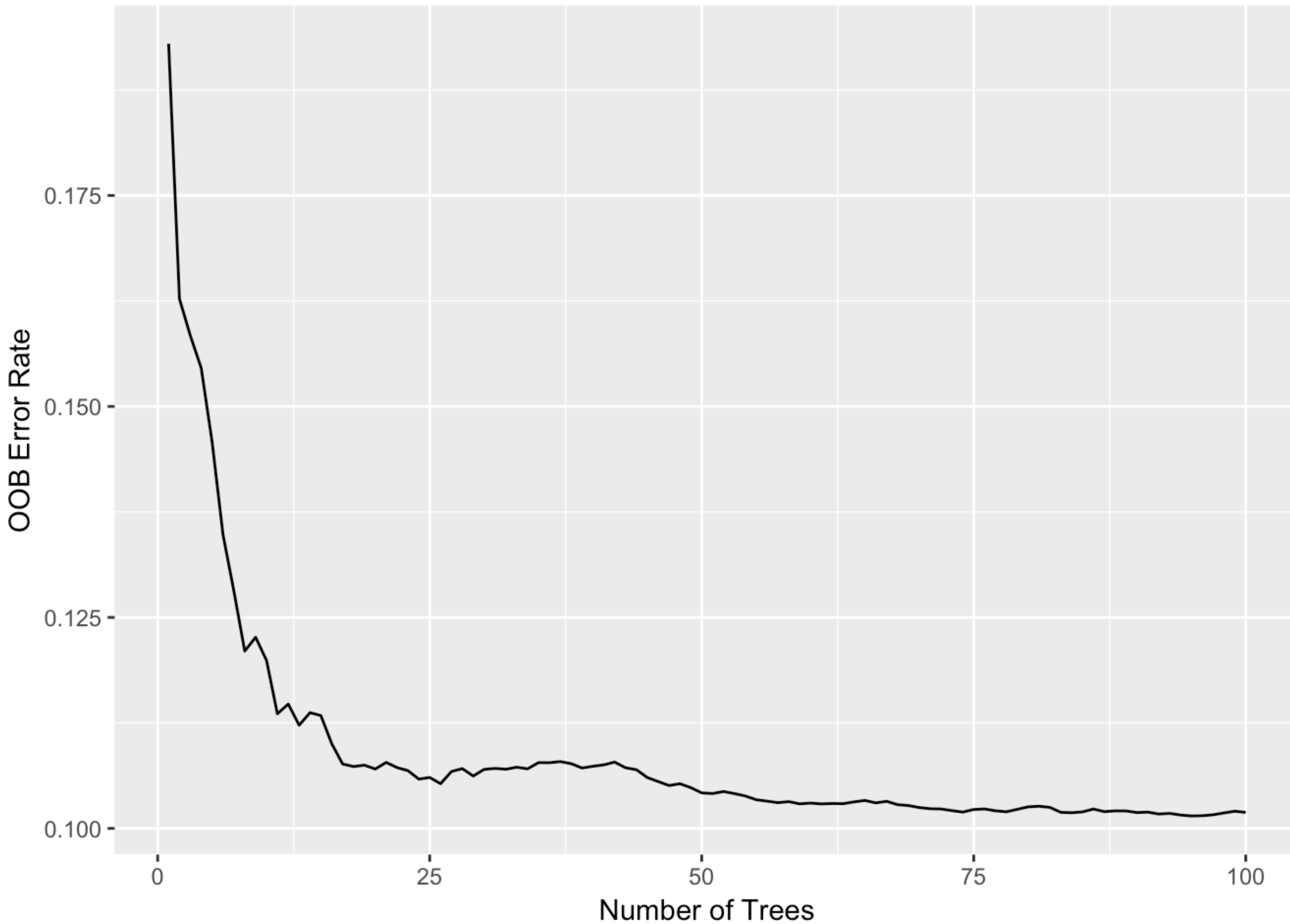
- `age`
- `female`
- `pss_fr`
- `homeless`
- `pcs`
- `cesd`

Use the code above to fit the model and explore how well the model converges and how well it does predicting MCS scores.

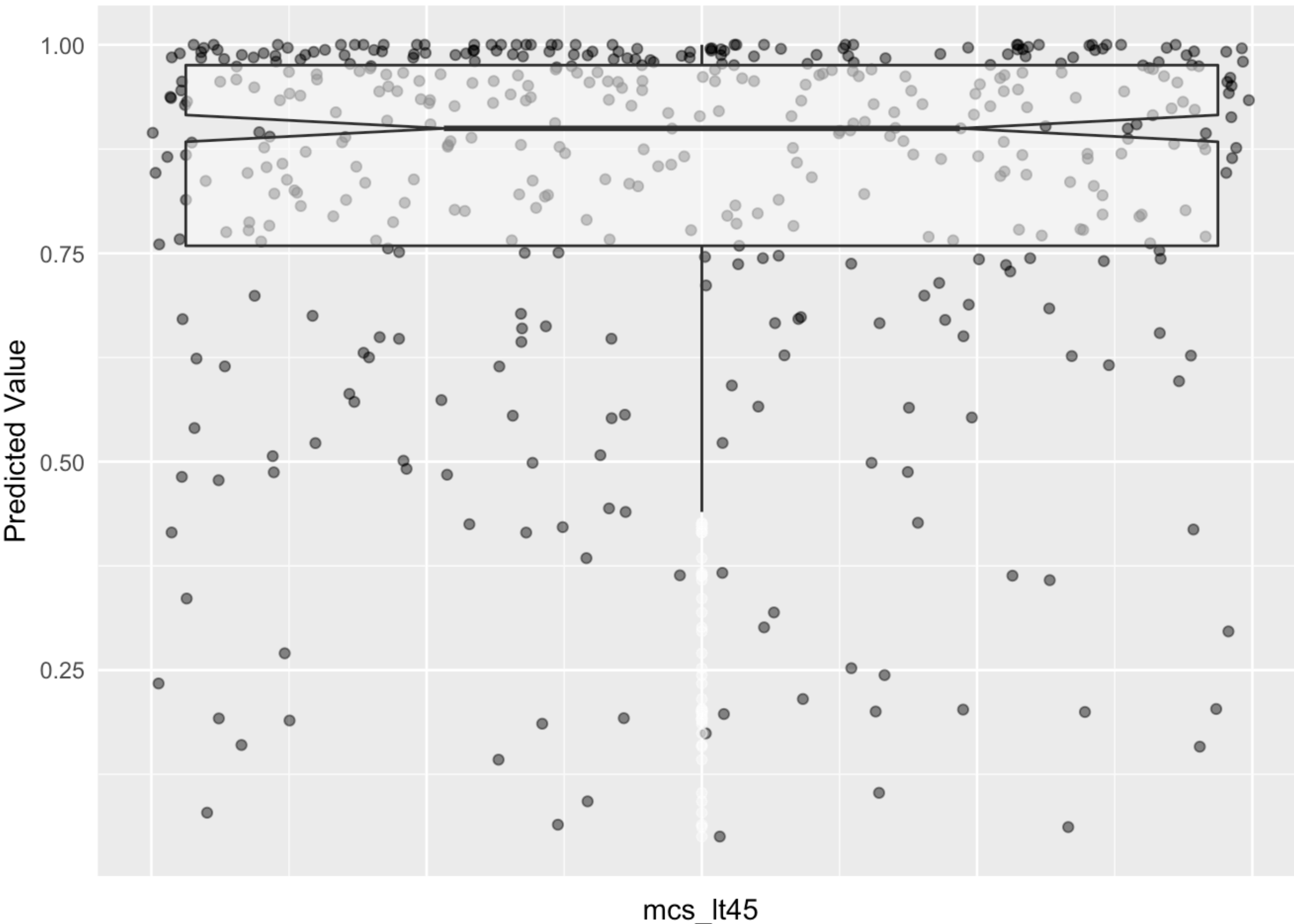
```
h1 <- as.data.frame(h1)
set.seed(131)
# Random Forest for the h1 dataset
fitallrf <- randomForestSRC::rfsrc(mcs_lt45 ~ age + female +
                                     pss_fr + homeless + pcs + cesd,
                                   data = h1, ntree = 100,
                                   tree.err=TRUE)
# view the results
fitallrf
```

```
##                          Sample size: 453
##                      Number of trees: 100
##            Forest terminal node size: 5
##        Average no. of terminal nodes: 38.69
## No. of variables tried at each split: 2
##               Total no. of variables: 6
##                             Analysis: RF-R
##                               Family: regr
##                       Splitting rule: mse
##                 % variance explained: 32.69
##                           Error rate: 0.1
```

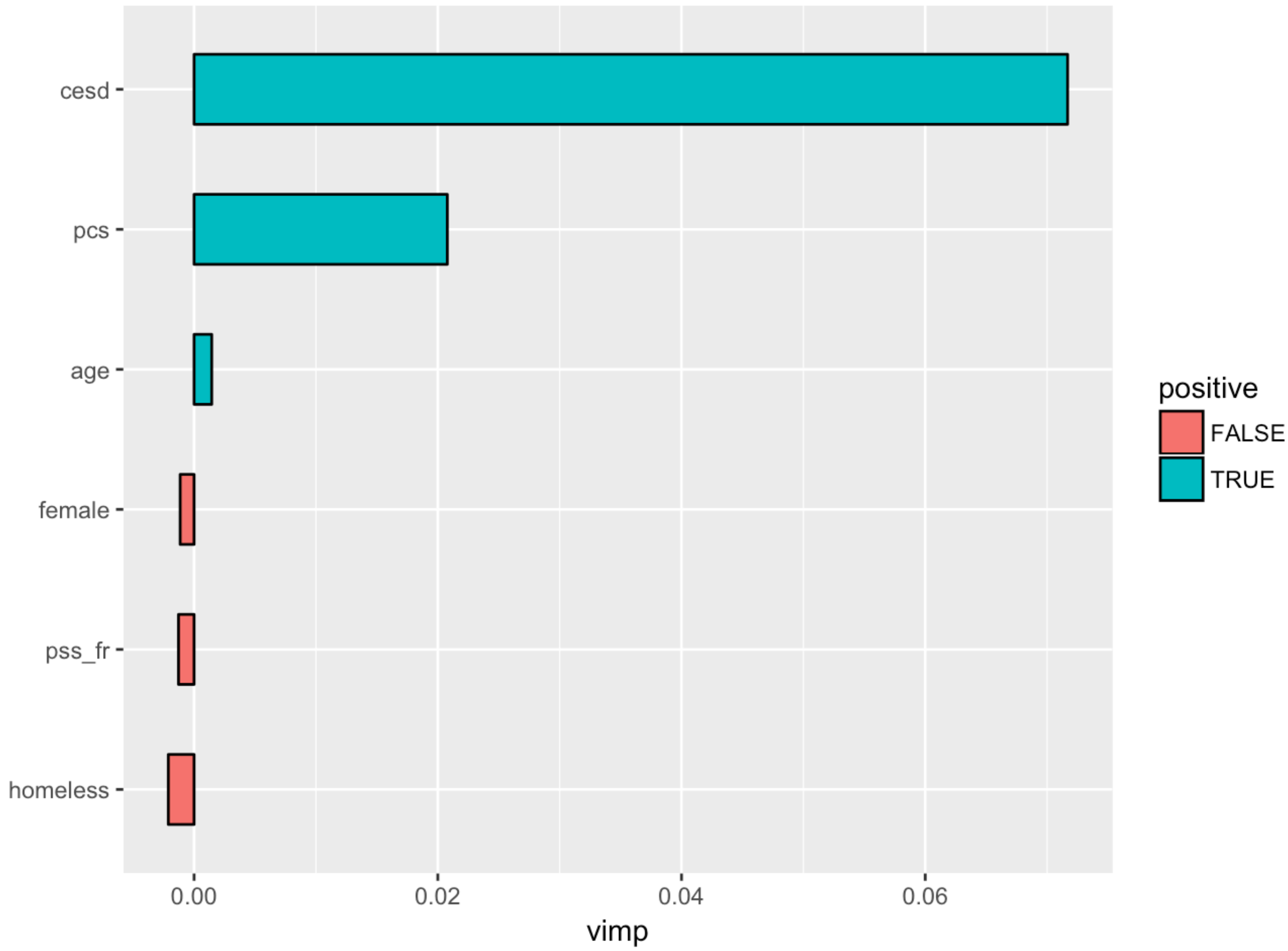
```
gg_e <- ggRandomForests::gg_error(fitallrf)
plot(gg_e)
```




```
# Plot the predicted cesd values
plot(ggRandomForests::gg_rfsrc(fitallrf), alpha = 0.5)
```




```
# Plot the VIMP rankins of independent variables
plot(ggRandomForests::gg_vimp(fitallrf))
```



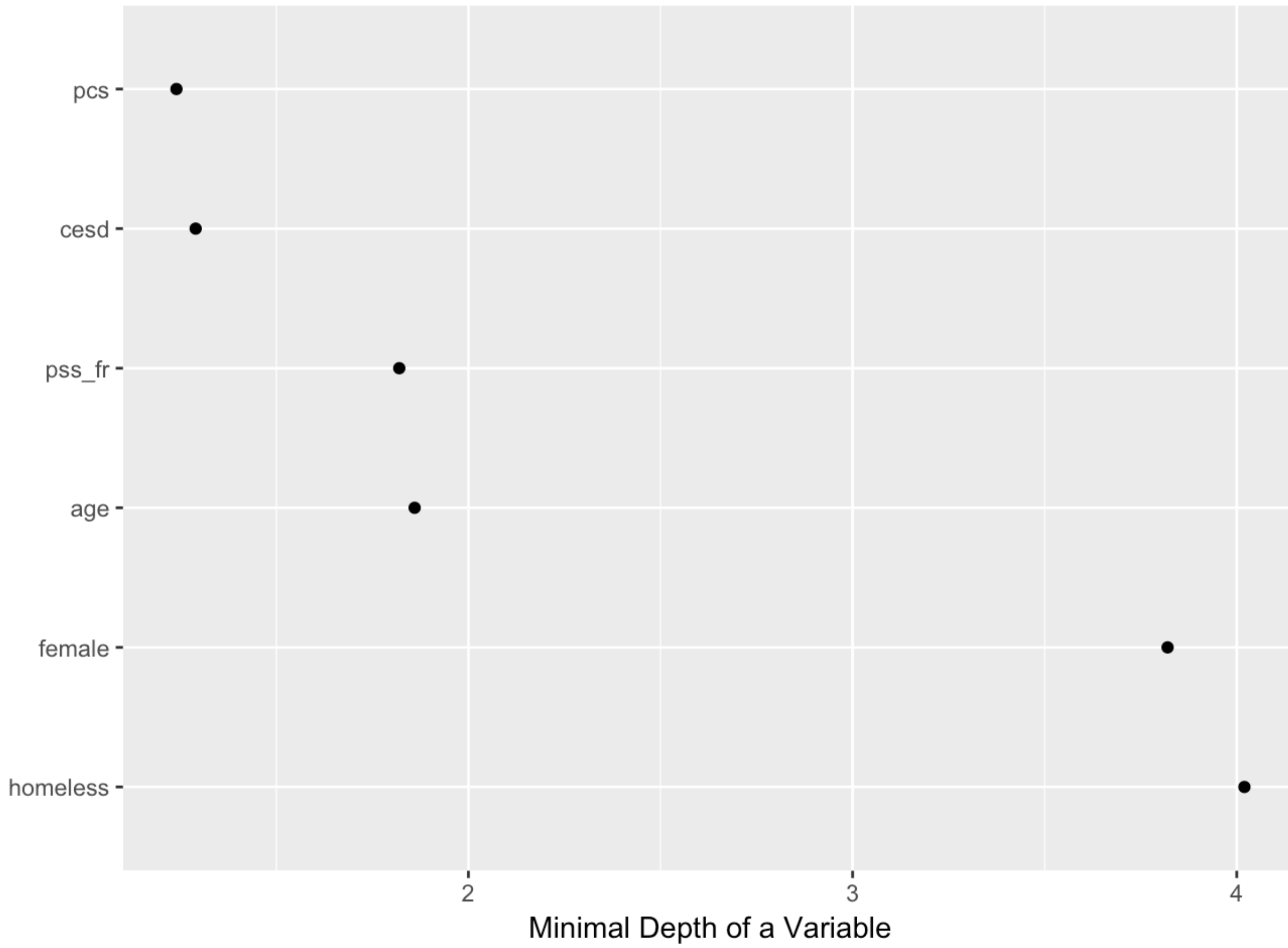

```
# Select the variables
varsel_cesd <- randomForestSRC::var.select(fitallrf)
```

```
## minimal depth variable selection ...
##
##
## -----------------------------------------------------------
## family             : regr
## var. selection     : Minimal Depth
## conservativeness   : medium
## x-weighting used?  : TRUE
## dimension          : 6
## sample size        : 453
## ntree              : 100
## nsplit             : 0
## mtry               : 2
## nodesize           : 5
## refitted forest    : FALSE
## model size         : 6
## depth threshold    : 4.557
## PE (true OOB)      : 0.1019
##
##
## Top variables:
##          depth vimp
## pcs       1.24   NA
## cesd      1.29   NA
## pss_fr    1.82   NA
## age       1.86   NA
## female    3.82   NA
## homeless  4.02   NA
## -----------------------------------------------------------
```

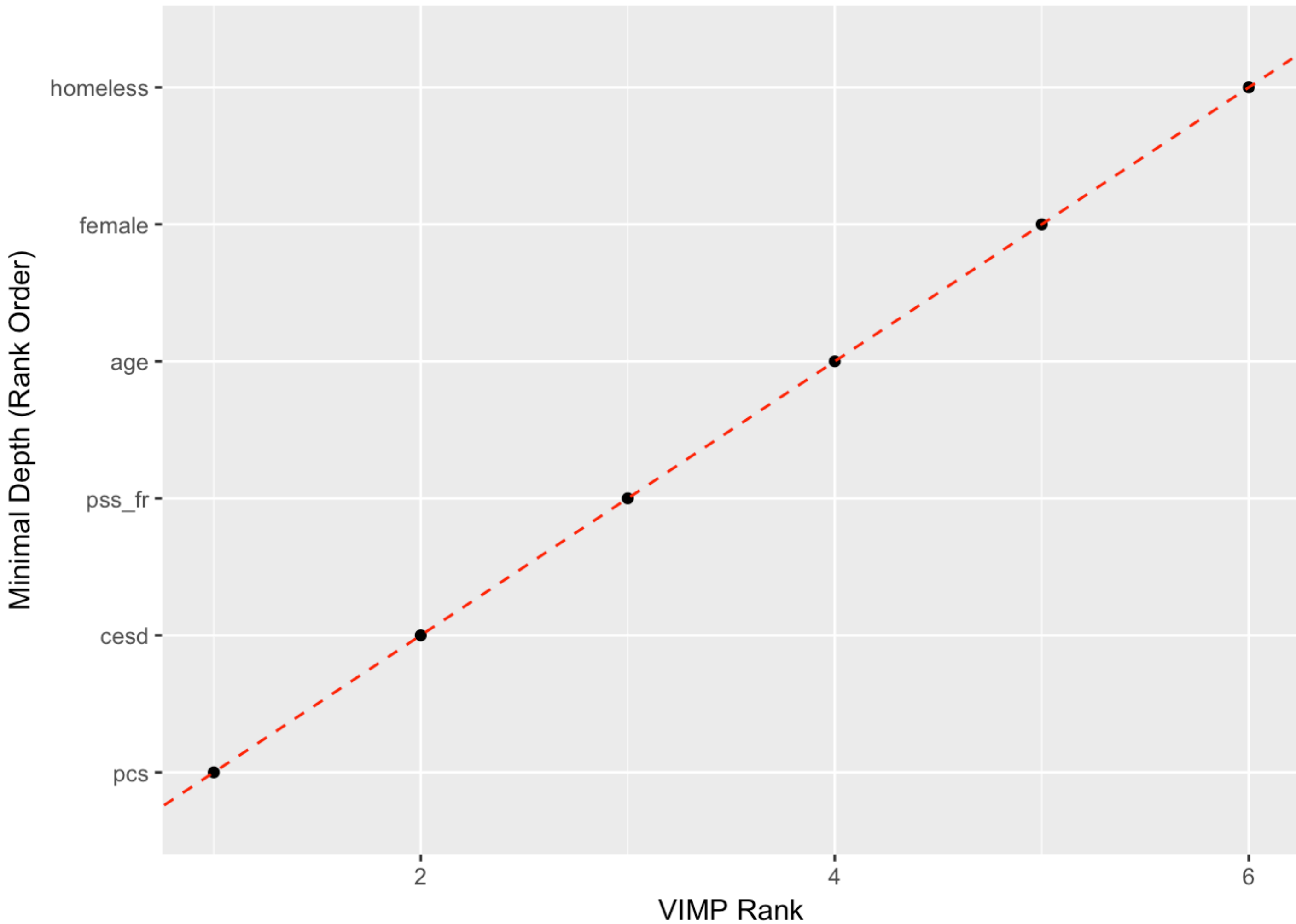
```
glimpse(varsel_cesd)
```

```
## List of 6
##  $ err.rate       : num 0.102
##  $ modelsize      : int 6
##  $ topvars        : chr [1:6] "pcs" "cesd" "pss_fr" "age" ...
##  $ varselect      :'data.frame': 6 obs. of  2 variables:
##   ..$ depth: num [1:6] 1.24 1.29 1.82 1.86 3.82 4.02
##   ..$ vimp : num [1:6] NA NA NA NA NA NA
##  $ rfsrc.refit.obj: NULL
##  $ md.obj         :List of 11
##   ..$ order                 : num [1:6, 1:2] 1.86 3.82 1.82 4.02 1.24 1.29 4.27 7.
63 5.66 7.47 ...
##   .. ..- attr(*, "dimnames")=List of 2
##   ..$ count                 : Named num [1:6] 0.1772 0.0918 0.1494 0.1019 0.1482 .
..
##   .. ..- attr(*, "names")= chr [1:6] "age" "female" "pss_fr" "homeless" ...
##   ..$ nodes.at.depth        : num [1:10000, 1:100] 2 3 6 8 10 8 5 2 0 NA ...
##   ..$ sub.order             : NULL
##   ..$ threshold             : num 4.56
##   ..$ threshold.1se         : num 4.69
##   ..$ topvars               : chr [1:6] "age" "female" "pss_fr" "homeless" ...
##   ..$ topvars.1se           : chr [1:6] "age" "female" "pss_fr" "homeless" ...
##   ..$ percentile            : Named num [1:6] 0.179 0.382 0.183 0.4 0.124 ...
##   .. ..- attr(*, "names")= chr [1:6] "age" "female" "pss_fr" "homeless" ...
##   ..$ density               : Named num [1:15] 0.0693 0.0998 0.1255 0.129 0.1133 .
..
##   .. ..- attr(*, "names")= chr [1:15] "0" "1" "2" "3" ...
##   ..$ second.order.threshold: num 7.05
```

```
# Save the gg_minimal_depth object for later use
gg_md <- ggRandomForests::gg_minimal_depth(varsel_cesd)
# Plot the object
plot(gg_md)
```



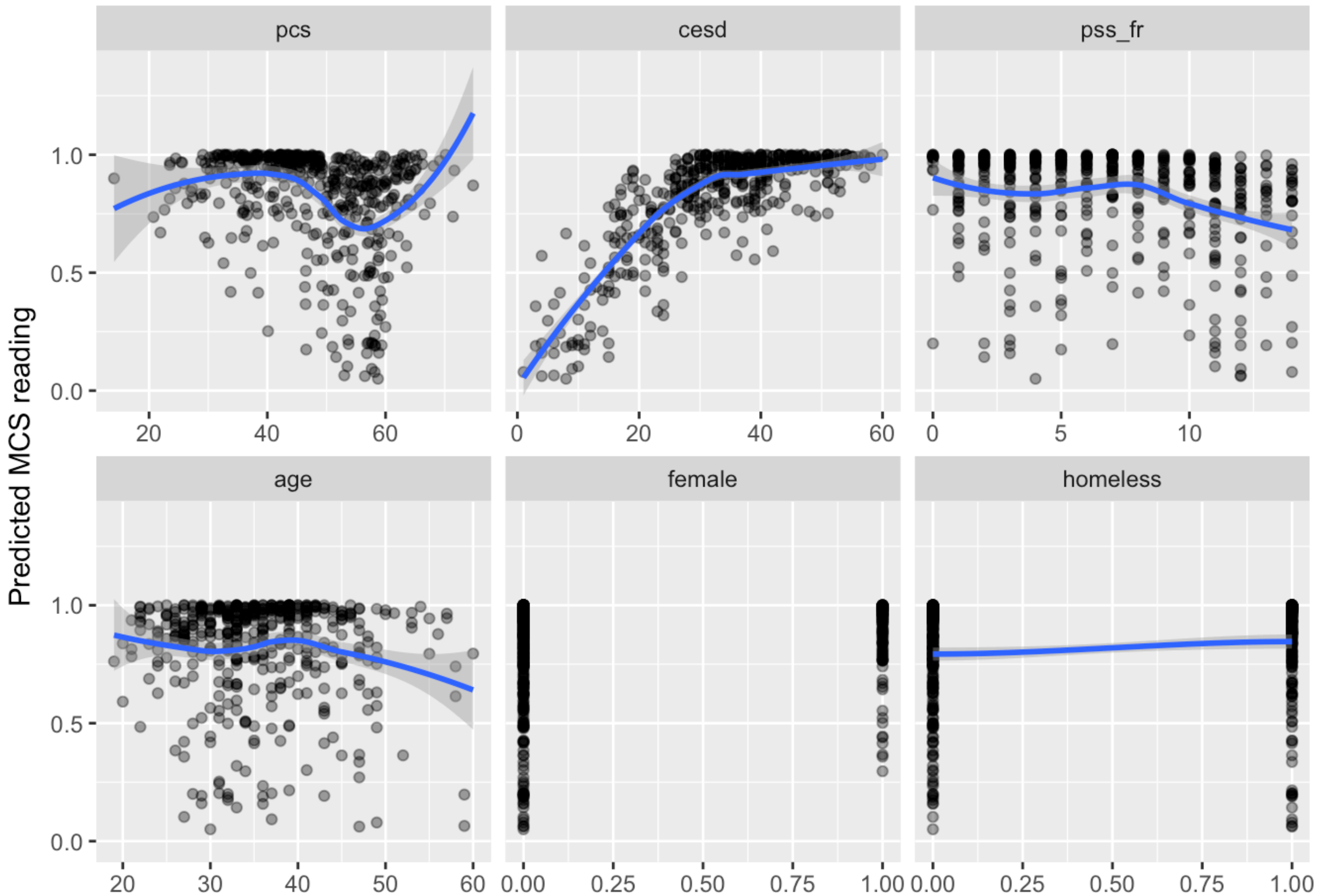

```
# Plot minimal depth v VIMP
gg_mdVIMP <- ggRandomForests::gg_minimal_vimp(gg_md)
plot(gg_mdVIMP)
```



PROBLEM 10: Create Plots of How Well Each Variable Predicts CESD*

Using the code above, see how well each variable predicts MCS scores given the other variables in the dataset `h1`.

```
#Create the variable dependence object from the random forest
gg_v <- ggRandomForests::gg_variable(fitallrf)

# Use the top ranked minimal depth variables only, plotted in minimal depth rank orde
r
xvar <- gg_md$topvars

# Plot the variable list in a single panel plot
plot(gg_v, xvar = xvar, panel = TRUE, alpha = 0.4) +
  labs(y="Predicted MCS reading", x="")
```



The github repository for this assignment can be accessed via this link (https://github.com/RosemaryKinuthia/N741Spring2018_Homework7.git (https://github.com/RosemaryKinuthia/N741Spring2018_Homework7.git)) [https://github.com/RosemaryKinuthia/N741Spring2018_Homework7.git (https://github.com/RosemaryKinuthia/N741Spring2018_Homework7.git)]
