Code

fMRI Biomarker for Deep Brain Stimulation

Trenton Jerde

Hide

```
library(dplyr)
library(ggplot2)
library(tidyr)
library(readr)
library(feather)
library(readxl)
library(rpart)
library(rpart.plot)
```

Hide

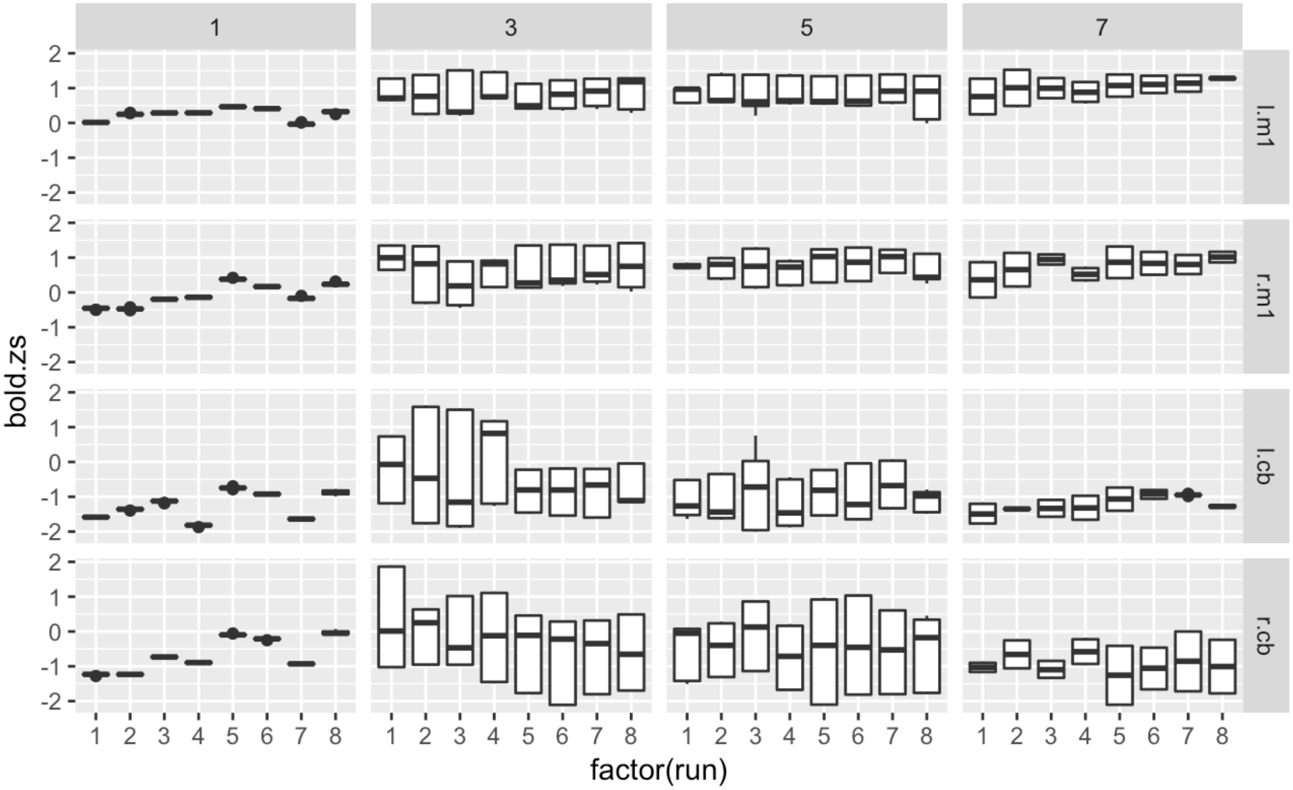
```
data.vlt <-
  read_excel("~/ds_stuff/projecdt_capstone/dbs_fmri/data/bold_4volt_3subs.xlsx")
data.frq <-
  read_delim("~/ds_stuff/projecdt_capstone/dbs_fmri/data/bold_4freq_3subs_new.csv
",
             "\t", escape_double = FALSE, trim_ws = TRUE)
names(data.frq) <-
  make.names(names(data.frq), unique = TRUE)
data.frq <-
  data.frq %>%
  rename(subj = X.subj)
```

Hide

```
data.frq <-
  data.frq %>%
  mutate(group = "frequency") %>%
  gather(run, bold, run1:run6)
data.vlt <-
  data.vlt %>%
  rename(voltage = volt) %>%
  mutate(freq = 130L) %>%
  mutate(group = "voltage") %>%
  gather(run, bold, run1:run8)
data <-
  bind_rows(
    data.frq,
    select_(
      data.vlt, .dots =
        c("subj", "area", "run", "dbsOnOff", "block", "contact", "voltage", "freq
", "group", "bold")
    )
  ) %>%
  select(group, everything()) %>%
  rename(dbs = dbsOnOff) %>%
  mutate(dbs = if_else(dbs == 2, 0, 1)) %>%
  mutate(run = as.integer(substr(run, 4, 4))) %>%
  mutate_at(vars(group:run), funs(as.factor)) %>%
  mutate(area.full = as.factor(ifelse(area == "1", "l.m1", ifelse(area == "2", "r
.m1", ifelse(area == "3", "l.cb", ifelse(area == "4", "r.cb", NA)))))) %>%
  mutate(area.full = factor(area.full, c("l.m1", "r.m1", "l.cb", "r.cb"))) %>%
  mutate(subj.alt = as.factor(ifelse(subj == "76", "s3", ifelse(subj == "635", "s
1", ifelse(subj == "985", "s2", NA))))) %>%
  group_by(group, subj, run) %>%
  mutate(bold.zs = (bold - mean(bold)) / sd(bold)) %>%
  ungroup() %>%
  mutate(area.rela = ifelse(subj.alt %in% c("s1", "s3"), ifelse(area.full == "l.m
1", "ipsi.m1", ifelse(area.full == "r.m1", "contra.m1", ifelse(area.full == "l.cb
", "ipsi.cb", ifelse(area.full == "r.cb", "contra.cb", NA)))), ifelse(area.full =
= "l.m1", "contra.m1", ifelse(area.full == "r.m1", "ipsi.m1", ifelse(area.full ==
"l.cb", "contra.cb", ifelse(area.full == "r.cb", "ipsi.cb", NA))))))
data <-
  data %>%
  select(group, subj, subj.alt, area, area.full, area.rela, dbs:bold, bold.zs)
rm(data.frq, data.vlt)
```

Hide

```
ggplot(
  data %>% filter(group == "voltage", subj.alt == "s1"),
  aes(
    x = factor(run),
    y = bold.zs
  )
) +
  geom_boxplot() +
  facet_grid(
    area.full ~ voltage
  )
```



Hide

```
std.err <-
  function(x) sd(x)/sqrt(length(x))
data.voltage <-
  data %>%
  filter(
    group == "voltage"
  )
CalculateAreaMeasures <- function(data) {
  data.test.tmp <-
    data %>%
    group_by(area.rela, contact, voltage, run, block) %>%
    summarise(
      bold.zs.avg = mean(bold.zs),
      dbs.val = unique(dbs)
    )

  data.test.tmp <-
    data.test.tmp %>%
    mutate(
      res =
        (lag(bold.zs.avg) - bold.zs.avg) /
        mean(c(lag(bold.zs.avg), bold.zs.avg), na.rm = TRUE) *
        100
    ) %>%
    filter(dbs.val == 0)

  data.test.tmp <-
    data.test.tmp %>%
    group_by(area.rela, contact, voltage, run) %>%
    mutate(pair = 1:n()) %>%
    select(area.rela, contact, voltage, run, pair, res)

  data.test.tmp <-
    data.test.tmp %>%
    group_by(area.rela, contact, voltage) %>%
    summarise(
      res.avg = mean(res, na.rm = TRUE),
      res.sem = std.err((res))
    ) %>%
    ungroup()
}
out <- list()
subj.contact.voltage.unqs <-
  data %>% select(subj.alt, contact, voltage) %>% unique()
for (i in 1:nrow(subj.contact.voltage.unqs)) {

  subj.alt.no <-
    subj.contact.voltage.unqs[i,]$subj.alt
  contact.no <-
    subj.contact.voltage.unqs[i,]$contact
  voltage.no <-
    subj.contact.voltage.unqs[i,]$voltage

  data.voltage.no <-
    data.voltage %>%
    filter(
      subj.alt == subj.alt.no &
        contact == contact.no &
        voltage == voltage.no
    )
  data.voltage.no.res <-
    CalculateAreaMeasures(data.voltage.no) %>%
    mutate(
      subj.alt = subj.alt.no
    )

  out[[i]] <- data.voltage.no.res
}
result <-
  bind_rows(out) %>%
  select(subj.alt, area.rela:res.sem) %>%
  group_by(subj.alt, contact, voltage) %>%
  mutate(
    res.avg.wins = ifelse(area.rela != "ipsi.m1" & res.avg < 0, abs(mean(res.avg)
), res.avg),
    ipsi.m1.weighted = (0.9 * (res.avg.wins[area.rela == "ipsi.m1"])),
    other.weighted = (0.1 * (mean(c(res.avg.wins[area.rela == "ipsi.cb"], res.avg
.wins[area.rela == "contra.cb"], res.avg.wins[area.rela == "contra.m1"])))),
    res.overall =
      ipsi.m1.weighted / other.weighted
  ) %>%
  arrange(subj.alt, desc(res.overall)); result %>%
  ungroup()
```

subj.alt <fctr>	area.rela <chr>	contact <fctr>	voltage <fctr>	res.avg <dbl>	res.sem <dbl>	res.avg.wins <dbl>	ips
s1	contra.cb	2	3	-1.492424171	2.68586407	0.372365767	
s1	contra.m1	2	3	0.124078968	0.25987457	0.124078968	
s1	ipsi.cb	2	3	0.669021111	0.70813147	0.669021111	
s1	ipsi.m1	2	3	2.188787158	1.58336077	2.188787158	
s1	contra.cb	1	1	0.460450739	1.07330687	0.460450739	
s1	contra.m1	1	1	1.048837796	1.21138233	1.048837796	

subj.alt	area.rela	contact	voltage	res.avg	res.sem	res.avg.wins	ips
<fctr>	<chr>	<fctr>	<fctr>	<dbl>	<dbl>	<dbl>	
s1	ipsi.cb	1	1	0.087214912	0.14367365	0.087214912	
s1	ipsi.m1	1	1	2.936753351	3.65902335	2.936753351	
s1	contra.cb	3	5	0.023934234	0.04168758	0.023934234	
s1	contra.m1	3	5	0.978259652	0.34377631	0.978259652	

1-10 of 84 rows | 1-8 of 10 columns Previous **1** 2 3 4 5 6 ... 9 Next

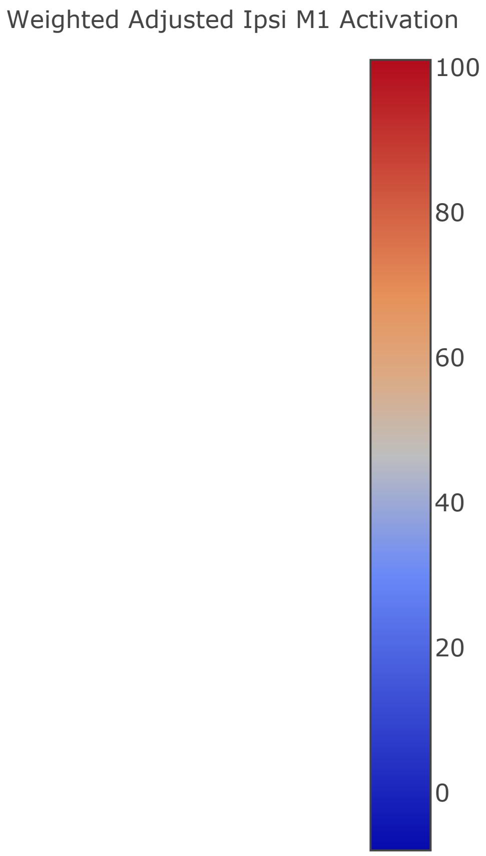
Hide

```
# https://plot.ly/r/3d-scatter-plots/
# http://www.statmethods.net/graphs/scatterplot.html
plot.data <-
  result %>%
  group_by(subj.alt, contact, voltage) %>%
  summarise(res = unique(res.overall)) %>%
  ungroup() %>%
  filter(!(subj.alt == "s3" & contact == "3" & voltage == "5"))
library(plotly)
```

```
Attaching package: ‘plotly’

The following object is masked from ‘package:ggplot2’:

    last_plot

The following object is masked from ‘package:stats’:

    filter

The following object is masked from ‘package:graphics’:

    layout
```

Hide

```
p <- plot_ly(
  plot.data,
  x = ~contact, y = ~voltage, z = ~res,
  marker = list(color = ~res, colorscale = c('#FFE1A1', '#683531'), showscale = T
RUE)) %>%
  add_markers() %>%
  layout(scene = list(xaxis = list(title = 'Contact'),
                      yaxis = list(title = 'Voltage'),
                      zaxis = list(title = 'Effectiveness')),
         annotations = list(
           x = 1.13,
           y = 1.05,
           text = 'Weighted Adjusted Ipsi M1 Activation',
           xref = 'paper',
           yref = 'paper',
           showarrow = FALSE
         ))
p
```



Hide

```
# # Create a shareable link to your chart
# # Set up API credentials: https://plot.ly/r/getting-started
# chart_link = plotly_POST(p, filename="scatter3d/colorscale")
# chart_link
```

Hide

```
# http://www.statmethods.net/stats/regression.html
fit <-
  lm(bold.zs ~ group + area.rela + dbs + contact + voltage + freq, data = data)
summary(fit)
```

```


Call:
lm(formula = bold.zs ~ group + area.rela + dbs + contact + voltage +
    freq, data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-3.5012 -0.4139 -0.0182  0.3796  2.4053

Coefficients: (1 not defined because of singularities)
                    Estimate Std. Error  t value Pr(>|t|)
(Intercept)        -1.111089   0.010644 -104.387  < 2e-16 ***
groupvoltage        0.290294   0.006958   41.722  < 2e-16 ***
area.relacontra.m1  1.219311   0.005439  224.176  < 2e-16 ***
area.relaipsi.cb   -0.160875   0.005439  -29.578  < 2e-16 ***
area.relaipsi.m1    1.530060   0.005439  281.309  < 2e-16 ***
dbs1                0.001181   0.003847    0.307  0.75884
contact2           -0.015744   0.005565   -2.829  0.00467 **
contact3           -0.228379   0.005470  -41.754  < 2e-16 ***
voltage3            0.366700   0.008777   41.778  < 2e-16 ***
voltage5            0.180250   0.008777   20.536  < 2e-16 ***
voltage7            0.391127   0.009751   40.111  < 2e-16 ***
freq130                   NA         NA       NA       NA
freq180             0.323256   0.008159   39.621  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6568 on 116628 degrees of freedom
Multiple R-squared:  0.5686,    Adjusted R-squared:  0.5685
F-statistic: 1.397e+04 on 11 and 116628 DF,  p-value: < 2.2e-16
```

Hide

```
coefficients(fit)
```

```
      (Intercept)       groupvoltage area.relacontra.m1   area.relaipsi.cb 
      -1.11108854         0.29029410         1.21931118        -0.16087506 
 area.relaipsi.m1               dbs1           contact2           contact3 
       1.53005981         0.00118103        -0.01574431        -0.22837915 
         voltage3           voltage5           voltage7            freq130 
       0.36669974         0.18024999         0.39112665                 NA 
          freq180 
       0.32325625 
```

Hide

```
confint(fit, level=0.95)
```

```
                        2.5 %       97.5 %
(Intercept)        -1.131950511 -1.090226576
groupvoltage        0.276656848  0.303931362
area.relacontra.m1  1.208650680  1.229971681
area.relaipsi.cb   -0.171535564 -0.150214563
area.relaipsi.m1    1.519399309  1.540720310
dbs1               -0.006358944  0.008721004
contact2           -0.026652510 -0.004836110
contact3           -0.239099526 -0.217658769
voltage3            0.349496468  0.383903016
voltage5            0.163046718  0.197453267
voltage7            0.372014464  0.410238840
freq130                      NA           NA
freq180             0.307265499  0.339247000
```

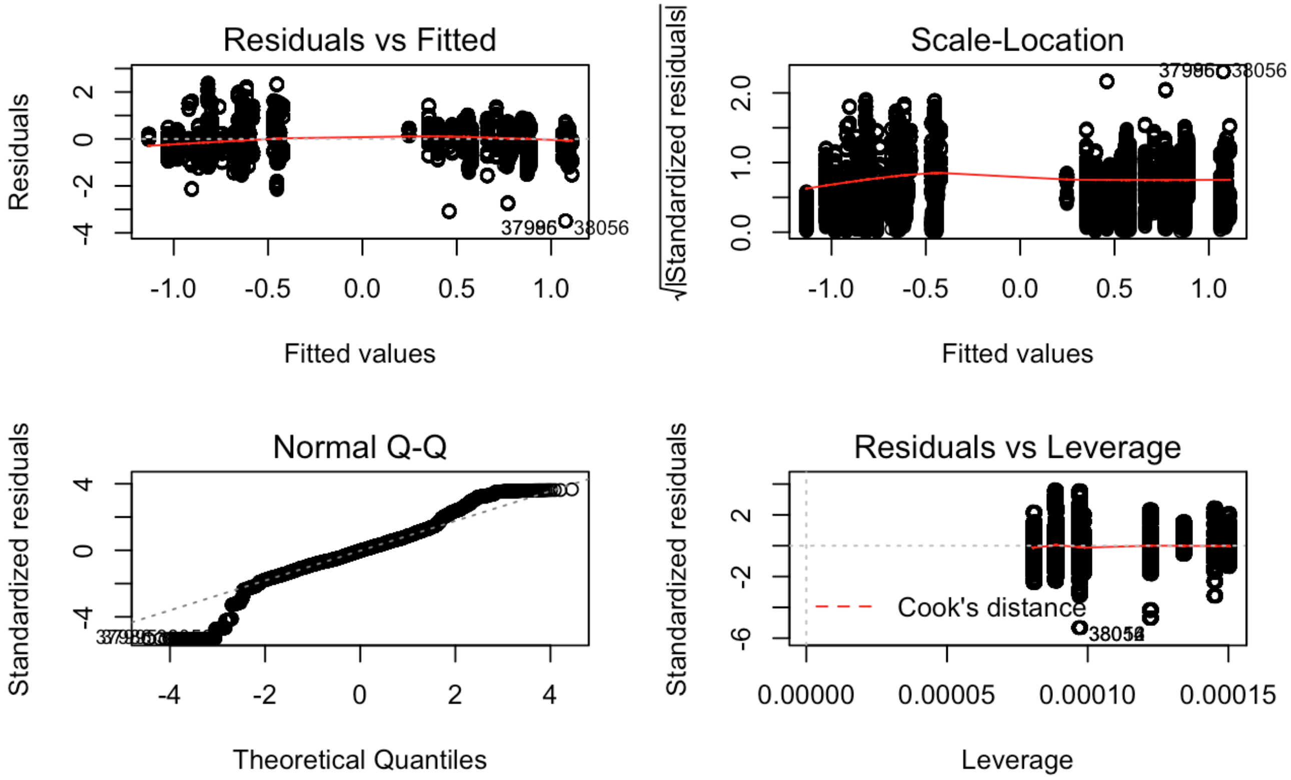
Hide

```
# fitted(fit)
# residuals(fit)
anova(fit)
```

```
Analysis of Variance Table

Response: bold.zs
              Df Sum Sq Mean Sq    F value Pr(>F)    
group          1      0     0.0     0.0000 1.0000    
area.rela      3  63528 21176.0 49095.0171 <2e-16 ***
dbs            1      0     0.0     0.0943 0.7588    
contact        2    874   436.8  1012.5845 <2e-16 ***
voltage        3   1220   406.8   943.0367 <2e-16 ***
freq           1    677   677.1  1569.8616 <2e-16 ***
Residuals 116628  50305     0.4                      
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Hide

```
# vcov(fit)
# influence(fit)
layout(matrix(c(1,2,3,4),2,2)) # optional 4 graphs/page
plot(fit)
```



Hide

```
# http://www.statmethods.net/advstats/cart.html
# http://blog.revolutionanalytics.com/2013/06/plotting-classification-and-regress
ion-trees-with-plotrpart.html
# http://www.milbo.org/rpart-plot/prp.pdf
# grow tree
fit <-
  rpart(
    bold.zs ~ group + area.rela + dbs + contact + voltage + freq,
    method = "anova",
    control = rpart.control(cp = 0.003),
    data = data
  )
# fit <- rpart(Mileage~Price + Country + Reliability + Type,
#    method="anova", data=cu.summary)
printcp(fit) # display the results
```

```
Regression tree:
rpart(formula = bold.zs ~ group + area.rela + dbs + contact +
    voltage + freq, data = data, method = "anova", control = rpart.control(cp = 0
.003))

Variables actually used in tree construction:
[1] area.rela contact   freq      voltage

Root node error: 116604/116640 = 0.99969

n= 116640

         CP nsplit rel error  xerror      xstd
1 0.5295092      0   1.00000 1.00003 0.0028884
2 0.0206381      1   0.47049 0.47050 0.0026948
3 0.0149736      2   0.44985 0.44987 0.0027459
4 0.0120743      3   0.43488 0.43492 0.0025123
5 0.0096233      4   0.42280 0.42285 0.0025424
6 0.0043959      6   0.40356 0.40362 0.0024508
7 0.0036837      7   0.39916 0.39923 0.0024328
8 0.0035013      8   0.39548 0.39556 0.0024223
9 0.0030000     10   0.38848 0.38857 0.0024361
```

Hide

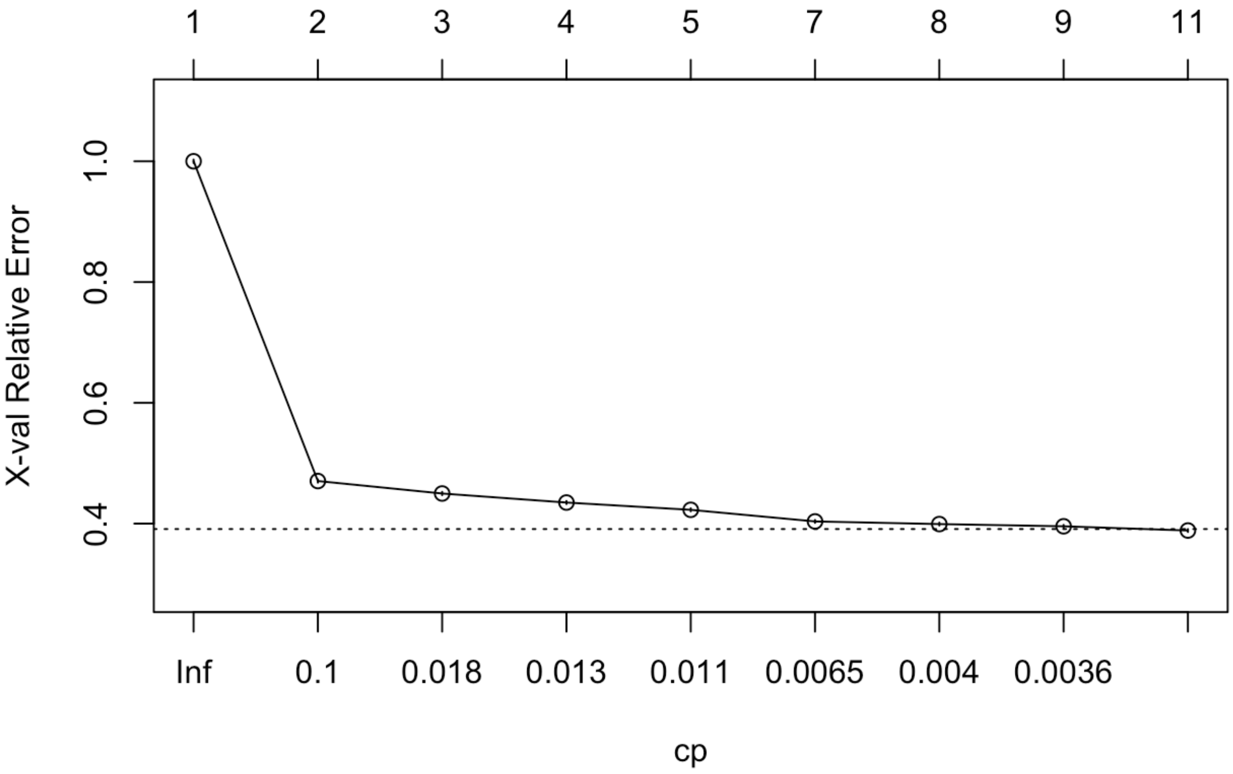
```
plotcp(fit) # visualize cross-validation results
summary(fit) # detailed summary of splits
```

```
Call:
rpart(formula = bold.zs ~ group + area.rela + dbs + contact +
    voltage + freq, data = data, method = "anova", control = rpart.control(cp = 0
.003))
  n= 116640

           CP nsplit rel error    xerror        xstd
1 0.529509184      0 1.0000000 1.0000268 0.002888431
2 0.020638139      1 0.4704908 0.4705037 0.002694793
3 0.014973560      2 0.4498527 0.4498723 0.002745938
4 0.012074315      3 0.4348791 0.4349208 0.002512326
5 0.009623287      4 0.4228048 0.4228533 0.002542409
6 0.004395947      6 0.4035582 0.4036229 0.002450821
7 0.003683676      7 0.3991623 0.3992347 0.002432814
8 0.003501347      8 0.3954786 0.3955552 0.002422309
9 0.003000000     10 0.3884759 0.3885661 0.002436089

Variable importance
area.rela   voltage   contact      freq     group
       88         5         5         2         1

Node number 1: 116640 observations,    complexity param=0.5295092
  mean=3.988198e-18, MSE=0.9996914
  left son=2 (58320 obs) right son=3 (58320 obs)
  Primary splits:
      area.rela splits as  LRLR, improve=5.295092e-01, (0 missing)
      voltage   splits as  LRLR, improve=6.329797e-03, (0 missing)
      contact   splits as  RRL,  improve=6.291888e-03, (0 missing)
      freq      splits as  LRR,  improve=3.266464e-03, (0 missing)
      dbs       splits as  LR,   improve=3.486433e-07, (0 missing)

Node number 2: 58320 observations,    complexity param=0.02063814
  mean=-0.7275615, MSE=0.6003057
  left son=4 (22680 obs) right son=5 (35640 obs)
  Primary splits:
      contact   splits as  RRL,  improve=0.0687375500, (0 missing)
      voltage   splits as  LRLR, improve=0.0184741400, (0 missing)
      freq      splits as  LRR,  improve=0.0122965000, (0 missing)
      area.rela splits as  R-L-, improve=0.0107781700, (0 missing)
      group     splits as  LR,   improve=0.0001607752, (0 missing)
  Surrogate splits:
      voltage splits as  RRRL, agree=0.648, adj=0.095, (0 split)

Node number 3: 58320 observations,    complexity param=0.01207432
  mean=0.7275615, MSE=0.3403855
  left son=6 (29160 obs) right son=7 (29160 obs)
  Primary splits:
      area.rela splits as  -L-R, improve=0.070923040, (0 missing)
      contact   splits as  LRR,  improve=0.031089810, (0 missing)
      voltage   splits as  LRLR, improve=0.008498923, (0 missing)
      freq      splits as  LLR,  improve=0.005027549, (0 missing)
      group     splits as  RL,   improve=0.000283544, (0 missing)

Node number 4: 22680 observations,    complexity param=0.01497356
  mean=-0.9822038, MSE=0.4796253
  left son=8 (16200 obs) right son=9 (6480 obs)
  Primary splits:
      voltage   splits as  -LLR, improve=1.605068e-01, (0 missing)
      area.rela splits as  R-L-, improve=2.969807e-03, (0 missing)
      freq      splits as  LLR,  improve=4.780479e-04, (0 missing)
      group     splits as  RL,   improve=3.864680e-04, (0 missing)
      dbs       splits as  LR,   improve=4.739182e-08, (0 missing)

Node number 5: 35640 observations,    complexity param=0.009623287
  mean=-0.5655164, MSE=0.6095801
  left son=10 (4860 obs) right son=11 (30780 obs)
  Primary splits:
      freq      splits as  LRR,  improve=0.042530210, (0 missing)
      voltage   splits as  LRRR, improve=0.030376280, (0 missing)
      area.rela splits as  R-L-, improve=0.018995820, (0 missing)
      group     splits as  LR,   improve=0.007825542, (0 missing)
      contact   splits as  RL-,  improve=0.002721304, (0 missing)

Node number 6: 29160 observations
  mean=0.5721872, MSE=0.2480955

Node number 7: 29160 observations,    complexity param=0.004395947
  mean=0.8829358, MSE=0.3843932
  left son=14 (8100 obs) right son=15 (21060 obs)
  Primary splits:
      contact splits as  LRR,  improve=4.573015e-02, (0 missing)
      voltage splits as  RRLL, improve=1.502599e-02, (0 missing)
      freq    splits as  LLR,  improve=1.478618e-03, (0 missing)
      group   splits as  RL,   improve=2.985965e-04, (0 missing)
      dbs     splits as  LR,   improve=6.113844e-06, (0 missing)
  Surrogate splits:
      voltage splits as  LRRR, agree=0.796, adj=0.267, (0 split)

Node number 8: 16200 observations
  mean=-1.157684, MSE=0.1499303

Node number 9: 6480 observations
  mean=-0.5435037, MSE=1.034422

Node number 10: 4860 observations
  mean=-0.9707266, MSE=0.4960903
```

```
Node number 11: 30780 observations,    complexity param=0.009623287
  mean=-0.5015359, MSE=0.5974805
  left son=22 (17280 obs) right son=23 (13500 obs)
  Primary splits:
      voltage   splits as  LRLL, improve=0.071789450, (0 missing)
      area.rela splits as  R-L-, improve=0.020276010, (0 missing)
      contact   splits as  RL-,  improve=0.004737381, (0 missing)
      group     splits as  RL,   improve=0.004197774, (0 missing)
      freq      splits as  -LR,  improve=0.004197774, (0 missing)
  Surrogate splits:
      group splits as  RL,  agree=0.719, adj=0.36, (0 split)
      freq  splits as  -LR, agree=0.719, adj=0.36, (0 split)

Node number 14: 8100 observations
  mean=0.6691515, MSE=0.4748433

Node number 15: 21060 observations
  mean=0.9651606, MSE=0.3252655

Node number 22: 17280 observations,    complexity param=0.003683676
  mean=-0.6845932, MSE=0.441233
  left son=44 (8640 obs) right son=45 (8640 obs)
  Primary splits:
      area.rela splits as  R-L-, improve=5.633562e-02, (0 missing)
      voltage   splits as  L-RR, improve=4.620928e-02, (0 missing)
      contact   splits as  RL-,  improve=3.511804e-02, (0 missing)
      dbs       splits as  LR,   improve=4.892559e-07, (0 missing)

Node number 23: 13500 observations
  mean=-0.2672224, MSE=0.6996816

Node number 44: 8640 observations
  mean=-0.8422547, MSE=0.3597722

Node number 45: 8640 observations,    complexity param=0.003501347
  mean=-0.5269318, MSE=0.4729796
  left son=90 (2160 obs) right son=91 (6480 obs)
  Primary splits:
      voltage splits as  L-RR, improve=6.577038e-02, (0 missing)
      contact splits as  RL-,  improve=2.276113e-02, (0 missing)
      dbs     splits as  LR,   improve=8.546691e-07, (0 missing)

Node number 90: 2160 observations
  mean=-0.8324218, MSE=0.1822746

Node number 91: 6480 observations,    complexity param=0.003501347
  mean=-0.4251017, MSE=0.5284039
  left son=182 (4320 obs) right son=183 (2160 obs)
  Primary splits:
      contact splits as  RL-,  improve=1.599764e-01, (0 missing)
      voltage splits as  --RL, improve=5.910179e-03, (0 missing)
      dbs     splits as  LR,   improve=7.994244e-07, (0 missing)

Node number 182: 4320 observations
  mean=-0.6306889, MSE=0.2629851

Node number 183: 2160 observations
  mean=-0.01392737, MSE=0.805645
```

Hide

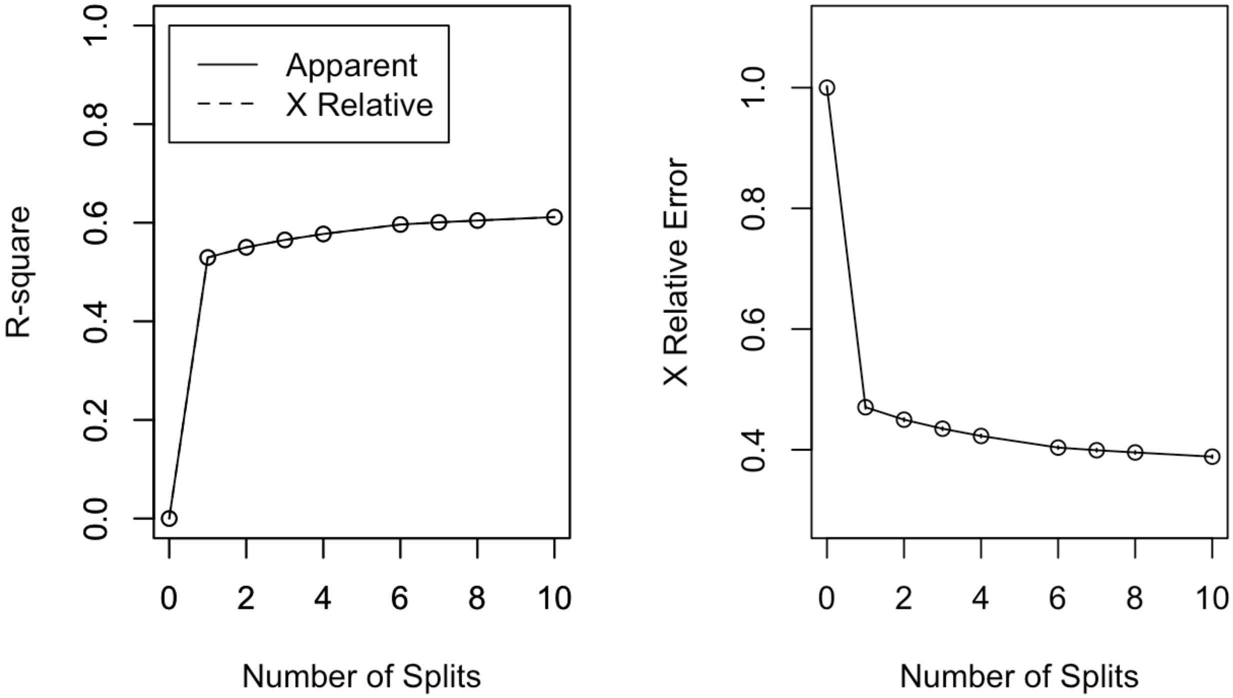
```
# create additional plots
par(mfrow=c(1,2)) # two plots on one page
```



Hide

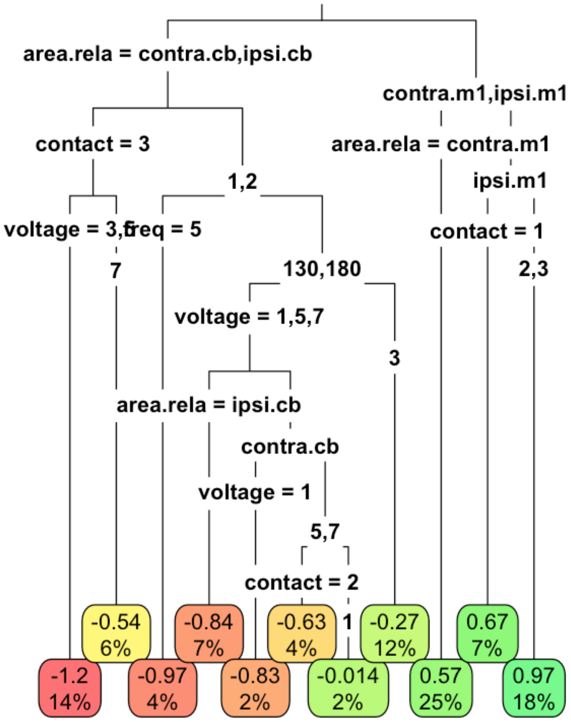
```
rsq.rpart(fit) # visualize cross-validation results
```

```
Regression tree:
rpart(formula = bold.zs ~ group + area.rela + dbs + contact +
    voltage + freq, data = data, method = "anova", control = rpart.control(cp = 0
.003))

Variables actually used in tree construction:
[1] area.rela contact   freq      voltage

Root node error: 116604/116640 = 0.99969

n= 116640

         CP nsplit rel error  xerror      xstd
1 0.5295092      0   1.00000 1.00003 0.0028884
2 0.0206381      1   0.47049 0.47050 0.0026948
3 0.0149736      2   0.44985 0.44987 0.0027459
4 0.0120743      3   0.43488 0.43492 0.0025123
5 0.0096233      4   0.42280 0.42285 0.0025424
6 0.0043959      6   0.40356 0.40362 0.0024508
7 0.0036837      7   0.39916 0.39923 0.0024328
8 0.0035013      8   0.39548 0.39556 0.0024223
9 0.0030000     10   0.38848 0.38857 0.0024361
```



Hide

```
# # plot tree
# plot(fit, uniform=TRUE,
#      main="Regression Tree for DBS Effect on Bold")
# text(fit, use.n=TRUE, all=TRUE, cex=.8)
# # create attractive postcript plot of tree
# post(fit, file = "c:/tree2.ps",
#      title = "Regression Tree for Mileage ")
rpart.plot(fit, type = 3, box.palette="RdYlGn", tweak = 1.2)
```


area.rela = contra.cb,ipsi.cb
contra.m1,ipsi.m1
contact = 3
area.rela = contra.m1
1,2
ipsi.m1
voltage = 3,5
freq = 5
contact = 1
7
130,180
2,3
voltage = 1,5,7
3
area.rela = ipsi.cb
contra.cb
voltage = 1
5,7
contact = 2
1
-0.54 6%
-0.84 7%
-0.63 4%
-0.27 12%
0.67 7%
-1.2 14%
-0.97 4%
-0.83 2%
-0.014 2%
0.57 25%
0.97 18%