FMPH221 Project Report Code

Gabriel Riegner & Kevin Nguyen

Loading Libraries

```
library(alr4)
```

```
## Loading required package: car

## Loading required package: carData

## Loading required package: effects

## Warning: package 'effects' was built under R version 4.1.1

## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

```
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 4.1.1

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.1.1

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(leaps)
```

Functions

```
normalize = function(x) { # used to scale data to 0 to 1 [later important for log transformation]
  (x - min(x)) / (max(x) - min(x))
}
```

Data and key

```
neuromaps_df = read.csv("/Users/kevinnguyen/Desktop/fmph221/data/neuromaps-mni152_y-siips_res-basc444.c
neuromaps_df = apply(neuromaps_df, 2, normalize) |> as.data.frame()
neuromaps_key = read.csv("/Users/kevinnguyen/Desktop/fmph221/data/neuromaps-key.csv")
```

Log transforming the skewed variables

Chosen by a visual inspection of each histogram

```
logcols = c(
    "aghourian2017.feobv",
    "bedard2019.feobv",
    "gallezot2010.p943",
    "jaworska2020.fallypride",
    "sandiego2015.flb457",
    "sasaki2012.fepe2i",
    "tuominen.feobv"
)
neuromaps_df[logcols] = log(neuromaps_df[logcols] + 1)
```

Principal Component Analyses to reduce collinearity and dimensionality

Many of the predictors are measuring the same things

```
neuromaps_pca = data.frame(siips = neuromaps_df$siips)

for (group in unique(neuromaps_key$description)) {
    colnames = neuromaps_key[neuromaps_key$description == group, "colname"]
    if (length(colnames) > 1) { # first PC
        pcs <- prcomp(neuromaps_df[, colnames])
        neuromaps_pca[, paste0(group, ".pc1")] = pcs$x[, 1]
    } else { # leave as is
        neuromaps_pca[group] = neuromaps_df[, colnames]
    }
}
```

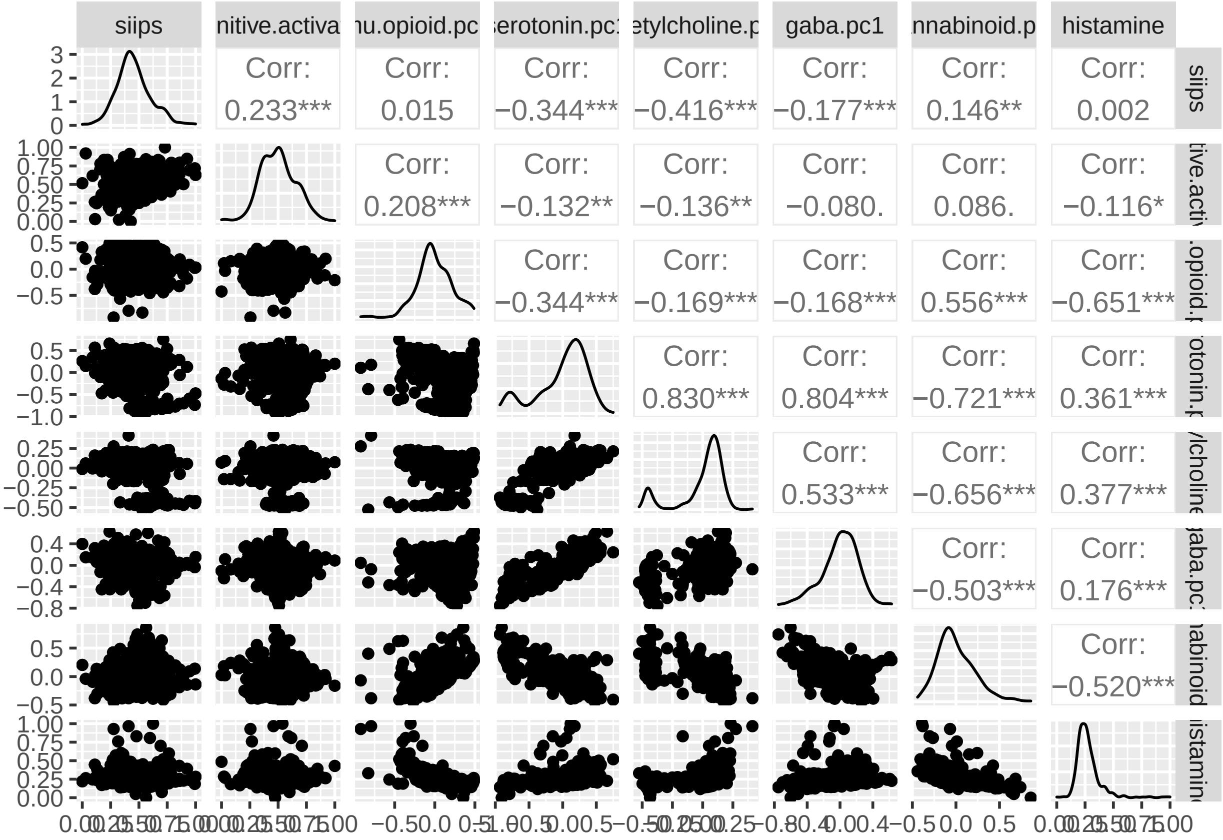
Feature selection

using in-sample criteria to maximize R^2

```
xs = as.matrix(neuromaps_pca[, -1])
all_subsets = leaps(xs, neuromaps_pca$siips, method = "adjr2")
col_idx = all_subsets$which[which.max(all_subsets$adjr2), ]
neuromaps_subset = data.frame(siips = neuromaps_pca$siips, xs[, col_idx])
```

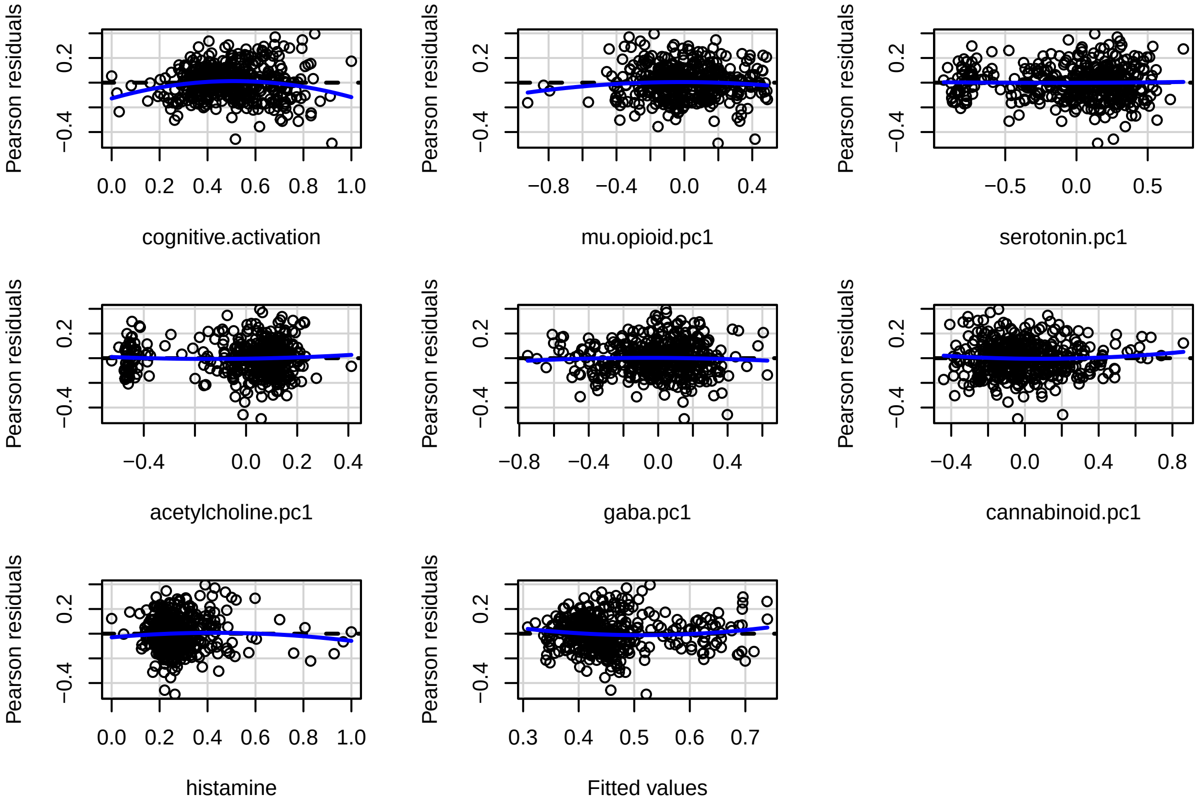
Pairs Plot

```
ggpairs(neuromaps_subset)
```



Naive model

```
ols1 = lm(siips ~ ., data = neuromaps_subset)
residualPlots(ols1) #  nonlinearity
```



```
##                      Test stat Pr(>|Test stat|)
## cognitive.activation   -3.0341         0.002557 **
## mu.opioid.pc1          -1.6227         0.105369
## serotonin.pc1           0.2382         0.811865
## acetylcholine.pc1       0.7013         0.483506
## gaba.pc1               -0.6238         0.533089
## cannabinoid.pc1         1.0840         0.278982
## histamine              -1.1978         0.231653
## Tukey test              1.9282         0.053836 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
vif(ols1) # collinearity
```

```
## cognitive.activation        mu.opioid.pc1        serotonin.pc1
##             1.088422             2.838364            10.478633
##    acetylcholine.pc1             gaba.pc1      cannabinoid.pc1
##             5.358733             3.783645             3.015873
##            histamine
##             2.103284
```

```
ncvTest(ols1) # heteroscedasticity
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 8.132561, Df = 1, p = 0.0043477
```

Addressing nonlinearity: quadratic terms

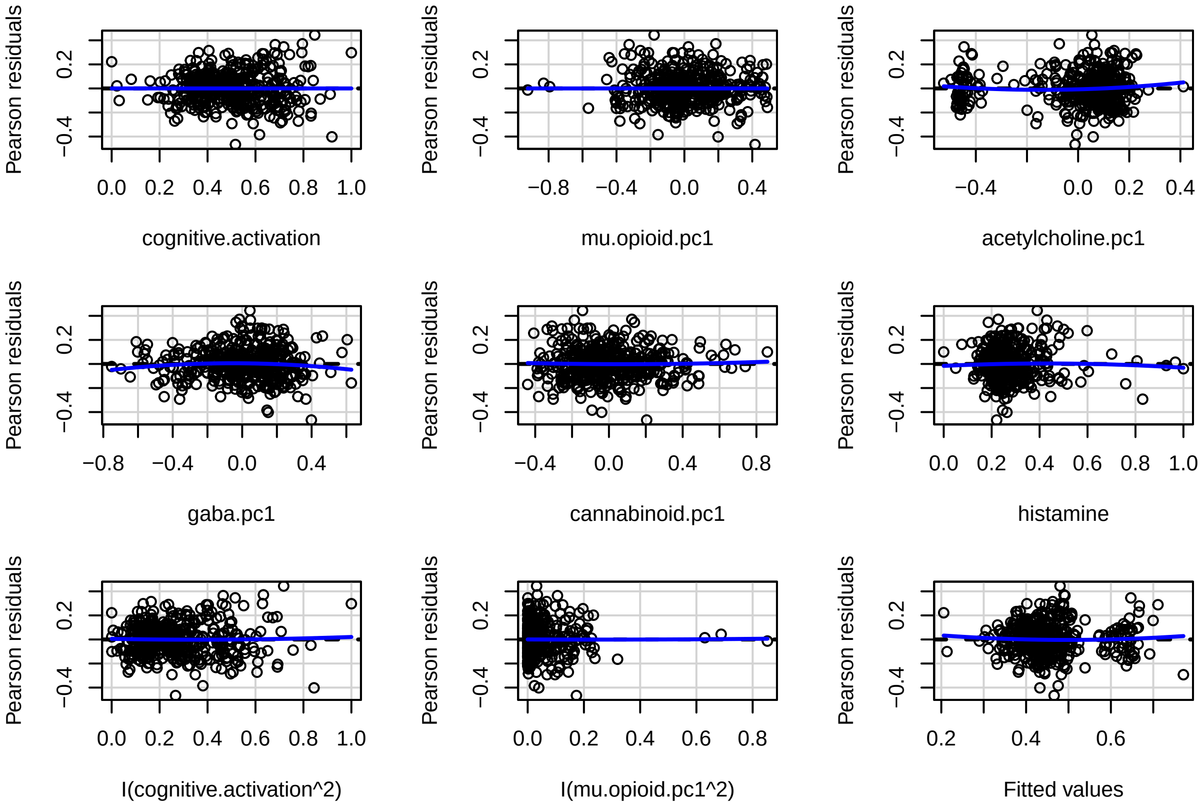
Addressing collinearity: removing serotonin

```
ols2 = update(ols1, ~ . - serotonin.pc1 + I(cognitive.activation^2) + I(mu.opioid.pc1^2))

anova(ols1, ols2) # confirm ols2 is a better model
```

```
## Analysis of Variance Table
##
## Model 1: siips ~ cognitive.activation + mu.opioid.pc1 + serotonin.pc1 +
##     acetylcholine.pc1 + gaba.pc1 + cannabinoid.pc1 + histamine
## Model 2: siips ~ cognitive.activation + mu.opioid.pc1 + acetylcholine.pc1 +
##     gaba.pc1 + cannabinoid.pc1 + histamine + I(cognitive.activation^2) +
##     I(mu.opioid.pc1^2)
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1    436 7.6417
## 2    435 7.4994  1   0.14233 8.2556 0.004261 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
residualPlots(ols2) # fixes nonlinearities
```



```
##                           Test stat Pr(>|Test stat|)
## cognitive.activation        -1.6190           0.1062
## mu.opioid.pc1                0.5513           0.5817
## acetylcholine.pc1            1.3049           0.1926
## gaba.pc1                    -1.5065           0.1327
## cannabinoid.pc1              0.4015           0.6883
## histamine                   -0.6729           0.5013
## I(cognitive.activation^2)    0.6000           0.5488
## I(mu.opioid.pc1^2)           0.0850           0.9323
## Tukey test                   0.8646           0.3873
```

```
ncvTest(ols2) # heteroscedasticity still an issue
```

```
## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 8.027564, Df = 1, p = 0.0046071
```

Check to see if errors are autocorrelated

```
resid = residuals(ols2)
cor(resid[-1], resid[-length(resid)])
```

```
## [1] 0.04566048
```

```
durbinWatsonTest(ols2) # GLS model doesn't seem nessessary
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.04565254      1.908337   0.354
##  Alternative hypothesis: rho != 0
```

Bootstrap to re-estimate standard errors and t-values, under heteroscedasity

```
n_bootstraps = 1000
n_samples = nrow(neuromaps_subset)
ols_coefs = ols2$coefficients
boot_coefs = matrix(0, n_bootstraps, length(ols_coefs))

for (i in 1:n_bootstraps) {
    idx = sample(1:n_samples, replace = TRUE)
    m = update(ols2, ~., data = neuromaps_subset[idx, ])
    boot_coefs[i, ] <- m$coefficients
}

stderr = apply(boot_coefs, 2, sd)
ci = apply(boot_coefs, 2, quantile, probs = c(0.025, 0.975))
tvals = ols_coefs / stderr
pvals = 2 * pt(-abs(tvals), df = nrow(neuromaps_subset) - length(ols_coefs))
```

Compiling bootstrap results

```
boot_summary = data.frame(
    "Estimate" = ols_coefs,
    "Std Error" = stderr,
    "t value" = tvals,
    "p value" = pvals
)
```

Compare summary tables

```
summary(ols2)$coefficients |> round(3)
```

```
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   0.119      0.056   2.136    0.033
## cognitive.activation          0.795      0.196   4.057    0.000
## mu.opioid.pc1                 0.093      0.046   2.031    0.043
## acetylcholine.pc1            -0.469      0.051  -9.110    0.000
## gaba.pc1                      0.045      0.035   1.276    0.203
## cannabinoid.pc1              -0.090      0.058  -1.540    0.124
## histamine                     0.420      0.089   4.709    0.000
## I(cognitive.activation^2)    -0.616      0.186  -3.311    0.001
## I(mu.opioid.pc1^2)           -0.205      0.099  -2.084    0.038
```

```
boot_summary |> round(3)
```

```
##                            Estimate Std.Error t.value p.value
## (Intercept)                   0.119     0.065   1.829   0.068
## cognitive.activation          0.795     0.253   3.138   0.002
## mu.opioid.pc1                 0.093     0.046   2.007   0.045
## acetylcholine.pc1            -0.469     0.055  -8.502   0.000
## gaba.pc1                      0.045     0.037   1.221   0.223
## cannabinoid.pc1              -0.090     0.052  -1.731   0.084
## histamine                     0.420     0.093   4.511   0.000
## I(cognitive.activation^2)    -0.616     0.261  -2.358   0.019
## I(mu.opioid.pc1^2)           -0.205     0.083  -2.488   0.013
```

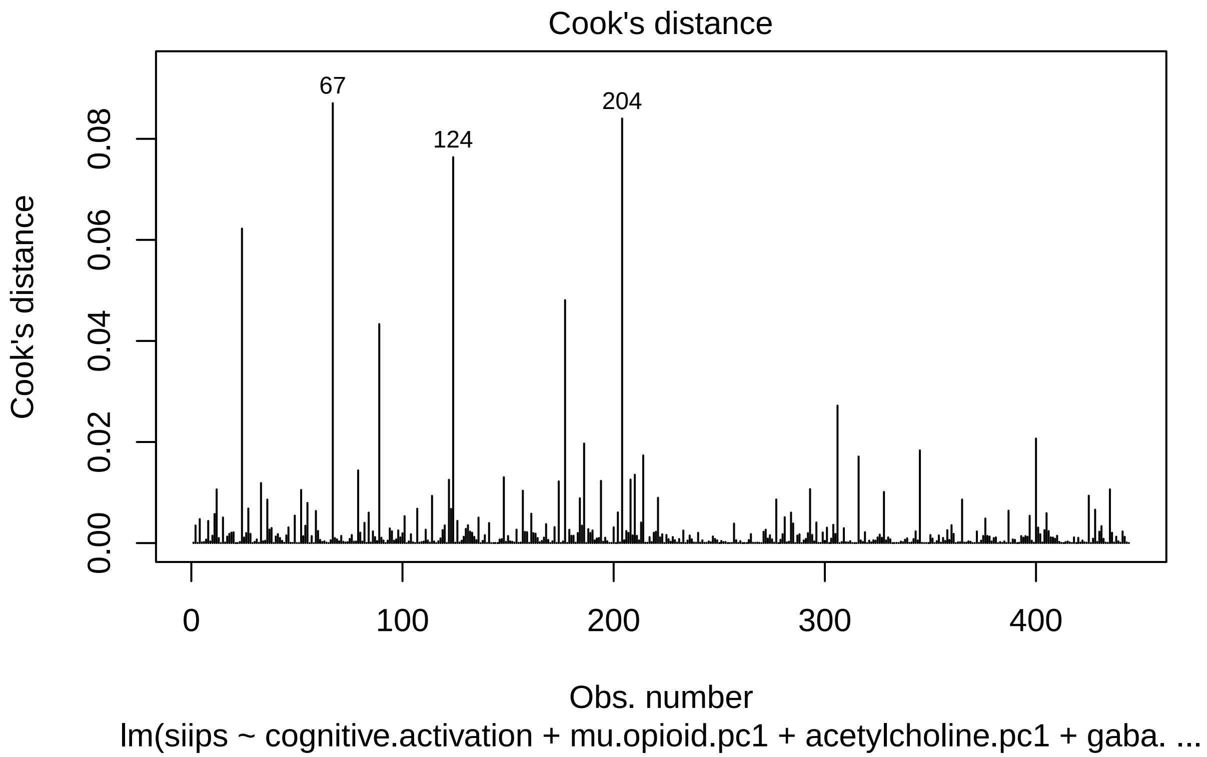
Compare confidence intervals

```
confint(ols2) |>
    cbind(t(ci)) |>
    round(3)
```

```
##                             2.5 % 97.5 %   2.5%  97.5%
## (Intercept)                 0.010  0.228 -0.025  0.232
## cognitive.activation        0.410  1.180  0.361  1.371
## mu.opioid.pc1               0.003  0.183  0.009  0.189
## acetylcholine.pc1          -0.570 -0.367 -0.572 -0.363
## gaba.pc1                   -0.024  0.115 -0.028  0.116
## cannabinoid.pc1            -0.204  0.025 -0.196  0.004
## histamine                   0.244  0.595  0.266  0.626
## I(cognitive.activation^2) -0.982 -0.250 -1.195 -0.163
## I(mu.opioid.pc1^2)        -0.399 -0.012 -0.385 -0.047
```

Outliers

```
plot(ols2, 4)
```

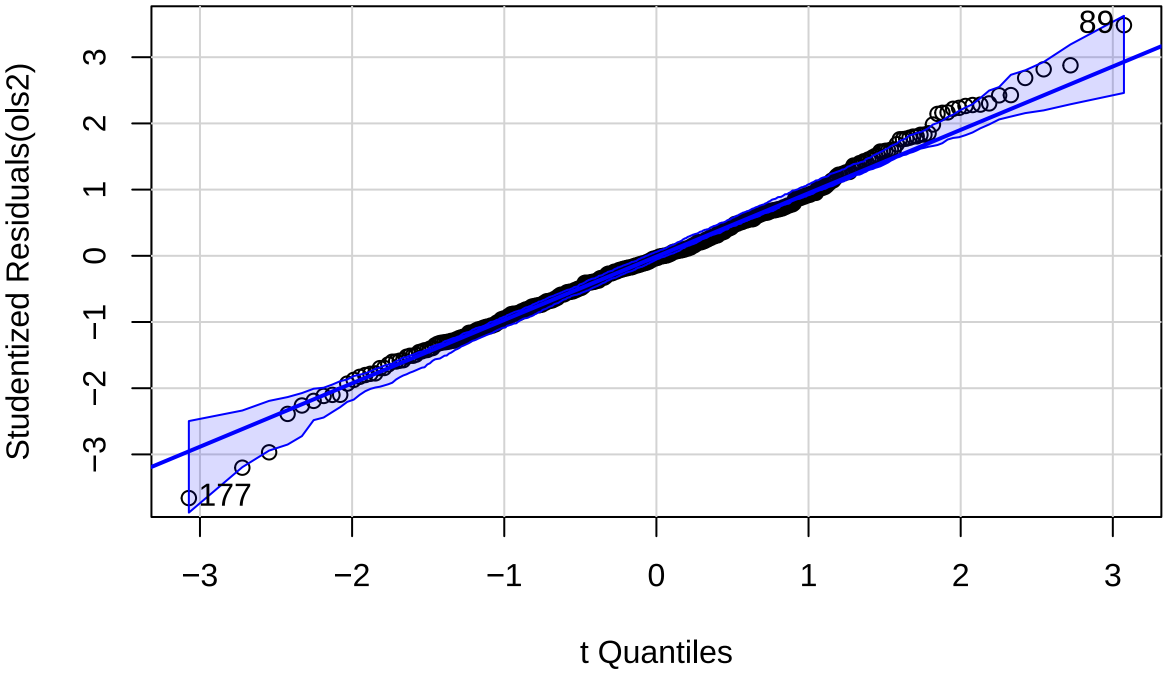


```
outlierTest(ols2)
```

```
## No Studentized residuals with Bonferroni p < 0.05
## Largest |rstudent|:
##      rstudent unadjusted p-value Bonferroni p
## 177 -3.659496         0.00028381      0.12601
```

Normality of errors

```
qqPlot(ols2)
```



```
## [1]  89 177
```