# melanomaAnalysis

Michael Miller

4/4/2022

### Read data and process variables

```
library(ggplot2)
df <- readRDS("Dataset.RData")
genes <- df[,c(9:ncol(df))]
df$ageStd <- (df$age-mean(df$age))/sd(df$age)
df$immuneClass <- ifelse(df$trans_class == "immune", 1, 0)
df$pathStageFactor <- as.factor(df$stage)
df$pathStageOrdered <- factor(df$stage,ordered=TRUE)
df$non_infiltration_count <- df$tile_count-df$infiltration_count
df$propInfiltrated <- df$infiltration_count/df$tile_count
phenoList <- c("ageStd","gender","pathStageFactor","immuneClass")
outcomeList <- c("infiltration_count","non_infiltration_count")
```

### Model 1 with general immune response genes

```
genesList1 <- c("KLRD1","RFX5","DDX58","EIF2AK2","CD86","C5")
df1 <- cbind(df[,outcomeList],df[,phenoList],genes[,genesList1])

m1 <- glm(cbind(infiltration_count,non_infiltration_count) ~ ., data = df1, family = binomial())
summary(m1)
```

```
##
## Call:
## glm(formula = cbind(infiltration_count, non_infiltration_count) ~
##     ., family = binomial(), data = df1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -19.785   -2.702    1.742    4.503   13.033
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       0.764772   0.023003  33.246  < 2e-16 ***
## ageStd           -0.021389   0.008340  -2.565   0.0103 *
## gendermale       -0.151022   0.015510  -9.737  < 2e-16 ***
## pathStageFactor2  0.341508   0.024077  14.184  < 2e-16 ***
## pathStageFactor3  0.188073   0.020492   9.178  < 2e-16 ***
## pathStageFactor4  1.765536   0.184947   9.546  < 2e-16 ***
## immuneClass      -0.308773   0.018460 -16.727  < 2e-16 ***
## KLRD1            -0.150194   0.010345 -14.518  < 2e-16 ***
## RFX5             -0.148238   0.008189 -18.103  < 2e-16 ***
## DDX58             0.140114   0.010704  13.090  < 2e-16 ***
## EIF2AK2          -0.105830   0.009201 -11.503  < 2e-16 ***
## CD86              0.110542   0.011352   9.738  < 2e-16 ***
## C5               -0.039438   0.007700  -5.121 3.03e-07 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 10648.2  on 239  degrees of freedom
## Residual deviance:  8365.6  on 227  degrees of freedom
## AIC: 9745.4
##
## Number of Fisher Scoring iterations: 4
```

### Model 1 with general immune response genes + inflammation genes

```
genesList2 <- c(genesList1, c("IL17RB","TLR4","CCL28","IL17C"))
df2 <- cbind(df[,outcomeList],df[,phenoList],genes[,genesList2])

m2 <- glm(cbind(infiltration_count,non_infiltration_count) ~ ., data = df2, family = binomial())
summary(m2)
```

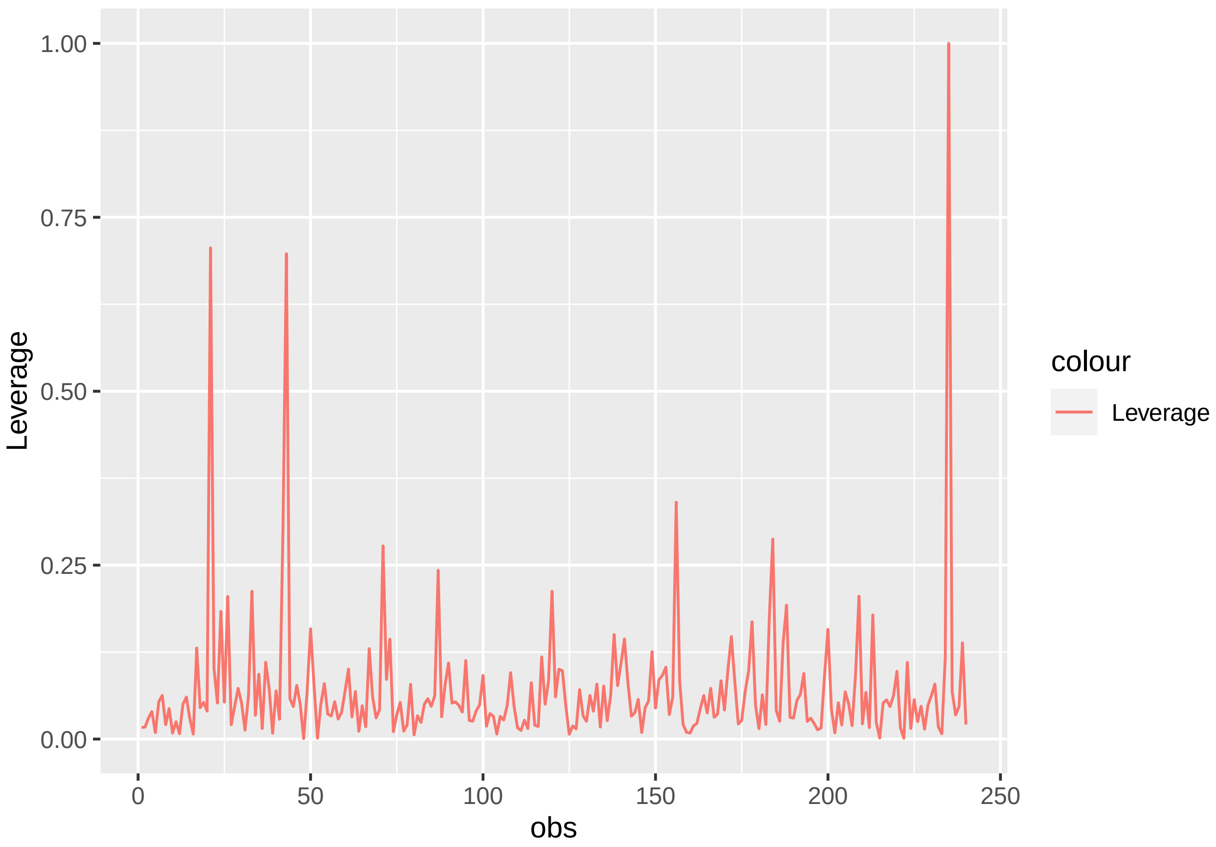
```
##
## Call:
## glm(formula = cbind(infiltration_count, non_infiltration_count) ~
##     ., family = binomial(), data = df2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -21.819   -2.643    1.572    4.250   14.241
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       0.758015   0.023251  32.601  < 2e-16 ***
## ageStd           -0.029258   0.008421  -3.475 0.000512 ***
## gendermale       -0.152022   0.015560  -9.770  < 2e-16 ***
## pathStageFactor2  0.378776   0.024528  15.443  < 2e-16 ***
## pathStageFactor3  0.140845   0.021073   6.684 2.33e-11 ***
## pathStageFactor4  1.880247   0.185861  10.116  < 2e-16 ***
## immuneClass      -0.267906   0.018782 -14.264  < 2e-16 ***
## KLRD1            -0.095883   0.011993  -7.995 1.30e-15 ***
## RFX5             -0.144239   0.008890 -16.225  < 2e-16 ***
## DDX58             0.122571   0.010855  11.291  < 2e-16 ***
## EIF2AK2          -0.048898   0.009562  -5.114 3.15e-07 ***
## CD86              0.080394   0.014731   5.458 4.83e-08 ***
## C5               -0.035778   0.007796  -4.589 4.45e-06 ***
## IL17RB           -0.050788   0.006845  -7.420 1.17e-13 ***
## TLR4              -0.120130   0.008921 -13.466   < 2e-16 ***
## CCL28             -0.147772   0.008064 -18.325   < 2e-16 ***
## IL17C              0.059851   0.007851   7.624 2.47e-14 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 10648.2  on 239  degrees of freedom
## Residual deviance:  7671.3  on 223  degrees of freedom
## AIC: 9059.2
##
## Number of Fisher Scoring iterations: 4
```

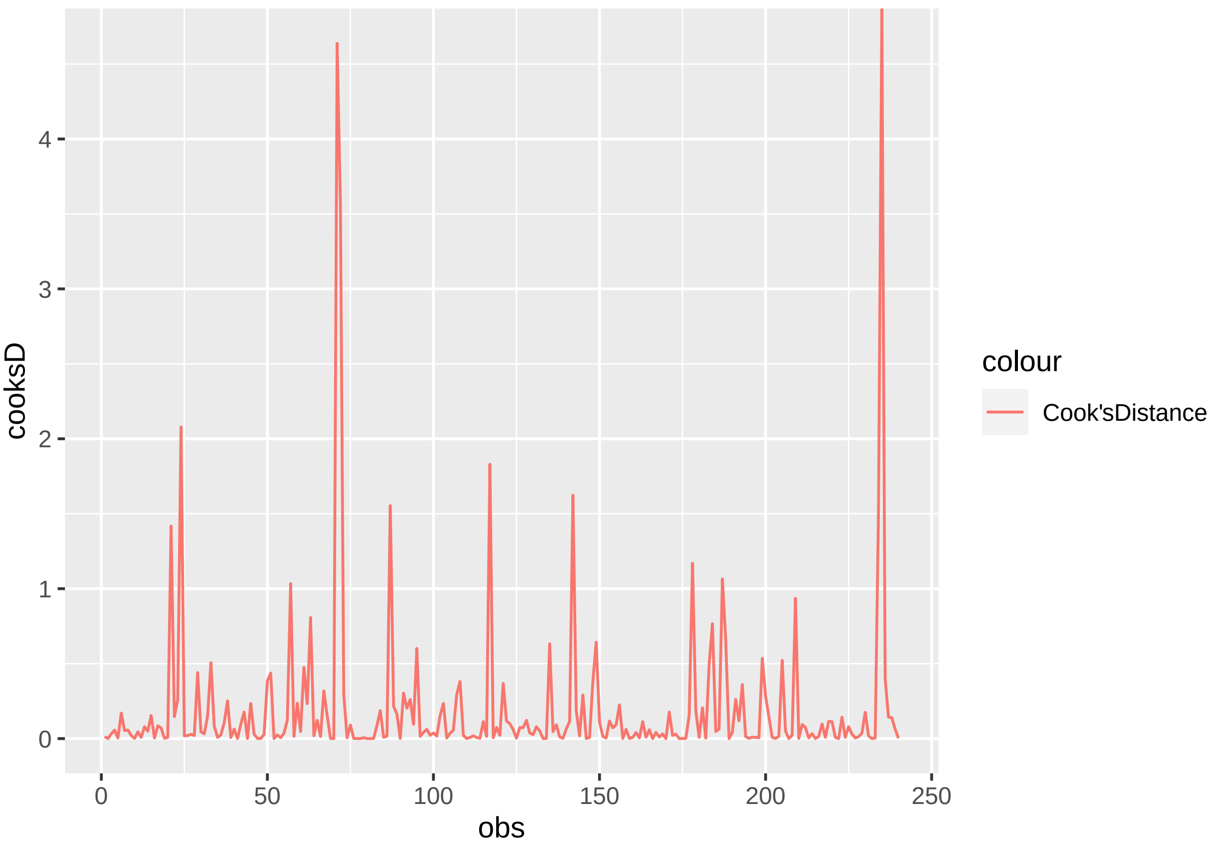
### GLM Diagnostics Function

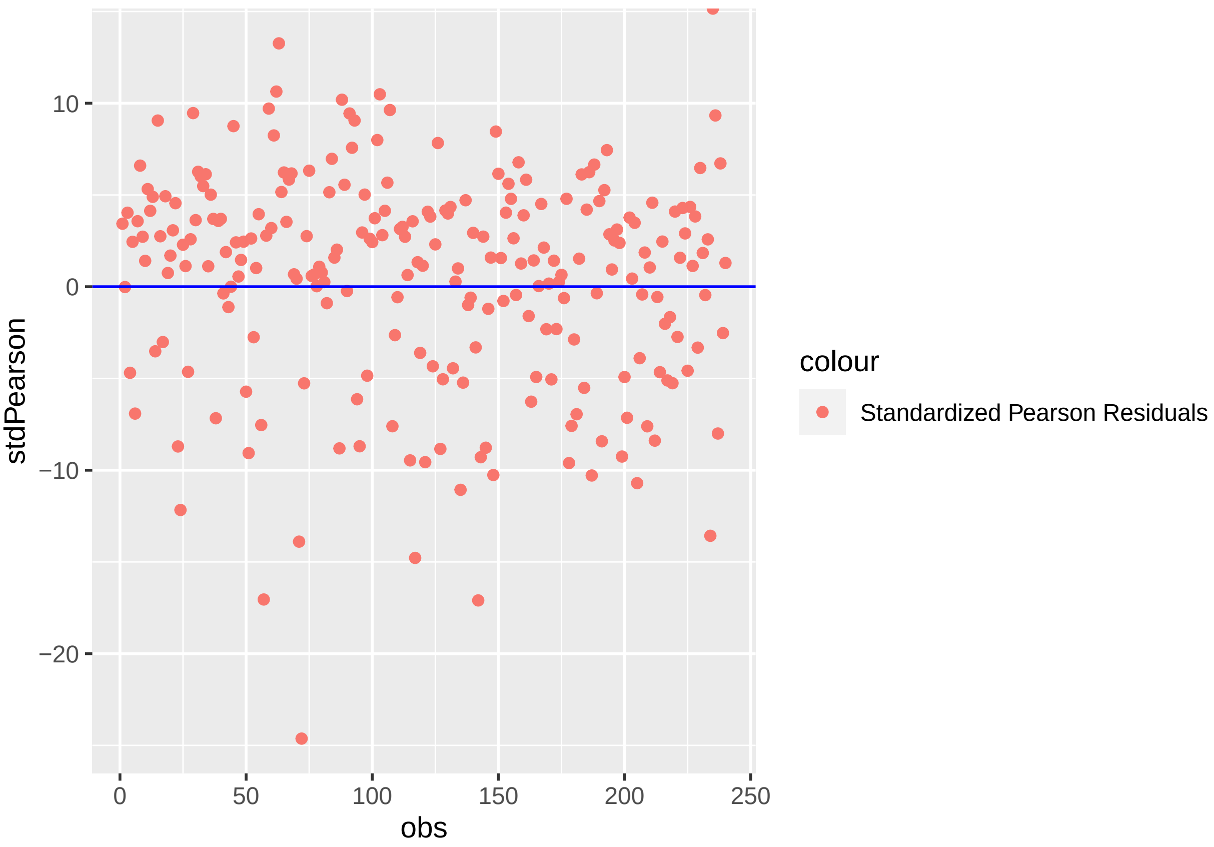
```
diagnosticsGLM <- function(mod, dat) {
  leverage <- influence(mod)$hat
  rp <- resid(mod, type = "pearson")
  rps <- rp/sqrt(1-leverage)
  cooksD <- leverage/(ncol(dat)*(1-leverage))*rps^2
  leverage.df <- data.frame(obs = 1:nrow(dat),
                            Leverage=leverage,stdPearson=rps,cooksD=cooksD)

  p1 <- ggplot(data=leverage.df, aes(x=obs))+geom_line(aes(y=Leverage, colour="Leverage"))
  plot(p1)

  p2 <- ggplot(data=leverage.df, aes(x=obs))+geom_line(aes(y=cooksD, colour="Cook'sDistance"))
  plot(p2)

  p3 <- ggplot(data=leverage.df, aes(x=obs))+geom_point(aes(y=stdPearson, colour="Standardized Pearson
  plot(p3)
  res <- c(p1,p2,p3)
  names(res) <- c("p1","p2","p3")
  return(res)
}
```

### Model 2 Diagnostics

```
diagRes <- diagnosticsGLM(m2,df2)
```


Leverage
1.00
0.75
0.50
0.25
0.00
0
50
100
150
200
250
obs
colour
Leverage


cooksD
0
1
2
3
4
obs
0
50
100
150
200
250
colour
Cook'sDistance



### Testing model with pathologic stage as ordinal response

```
library(VGAM)
```

```
## Warning: package 'VGAM' was built under R version 4.1.3
```

```
## Loading required package: stats4
```

```
## Loading required package: splines
```

```
phenoList <- c("ageStd","gender","propInfiltrated","immuneClass")
outcomeList <- c("pathStageOrdered")
genesList2 <- c(genesList1, c("IL17RB","TLR4","CCL28","IL17C"))
df3 <- cbind(df[,outcomeList],df[,phenoList],genes[,genesList2])

m3 = vglm(pathStageOrdered~.,data=df3, family=cumulative(parallel=T))
summary(m3)
```

```
##
## Call:
## vglm(formula = pathStageOrdered ~ ., family = cumulative(parallel = T),
##     data = df3)
```

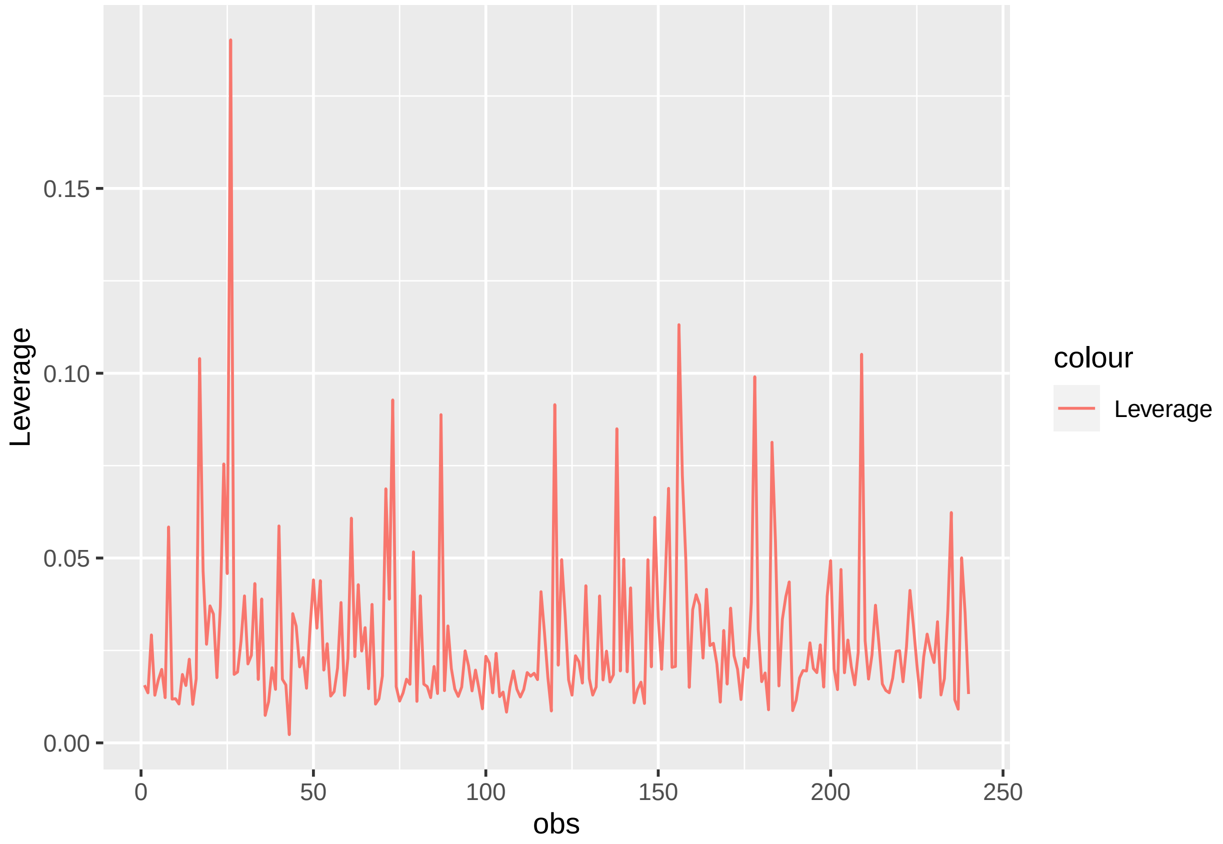
```
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept):1   -1.334348   0.511490  -2.609  0.00909 **
## (Intercept):2    0.265741   0.504912   0.526  0.59867
## (Intercept):3    6.304654   1.250113   5.043 4.58e-07 ***
## ageStd          -0.331420   0.098108  -3.378  0.00073 ***
## gendermale       0.195073   0.193950   1.006  0.31452
## propInfiltrated -0.606979   0.619006  -0.981  0.32680
## immuneClass      0.514165   0.235910   2.179  0.02929 *
## KLRD1            0.005921   0.150459   0.039  0.96861
## RFX5            -0.042155   0.113371  -0.372  0.71002
## DDX58            0.290809   0.126608   2.297  0.02162 *
## EIF2AK2         -0.035532   0.120604  -0.295  0.76829
## CD86            -0.231054   0.195329  -1.183  0.23685
## C5              -0.110356   0.099928  -1.104  0.26944
## IL17RB          -0.234213   0.116752  -2.006  0.04485 *
## TLR4             0.030805   0.121748   0.253  0.80025
## CCL28            0.304295   0.100175   3.038  0.00238 **
## IL17C           -0.555907   0.143711  -3.868  0.00011 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Names of linear predictors: logitlink(P[Y<=1]), logitlink(P[Y<=2]),
## logitlink(P[Y<=3])
##
## Residual deviance: 481.5022 on 703 degrees of freedom
##
## Log-likelihood: -240.7511 on 703 degrees of freedom
##
## Number of Fisher scoring iterations: 14
##
## Warning: Hauck-Donner effect detected in the following estimate(s):
## ’(Intercept):3’
##
##
## Exponentiated coefficients:
##          ageStd      gendermale propInfiltrated     immuneClass           KLRD1
##       0.7179038       1.2153998       0.5449950       1.6722423       1.0059383
##            RFX5           DDX58         EIF2AK2            CD86              C5
##       0.9587211       1.3375097       0.9650918       0.7936966       0.8955157
##          IL17RB            TLR4           CCL28           IL17C
##       0.7911933       1.0312844       1.3556692       0.5735516
```

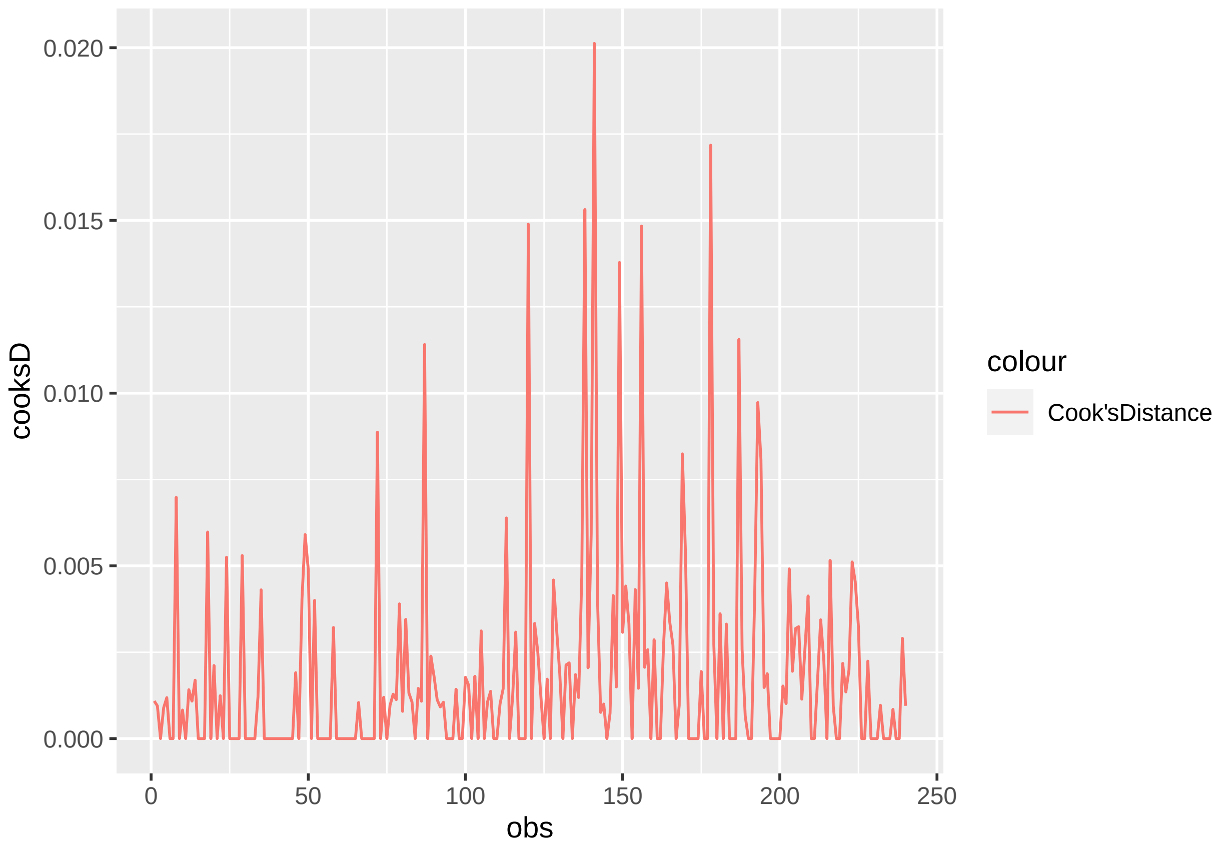
### VGLM Diagnostics

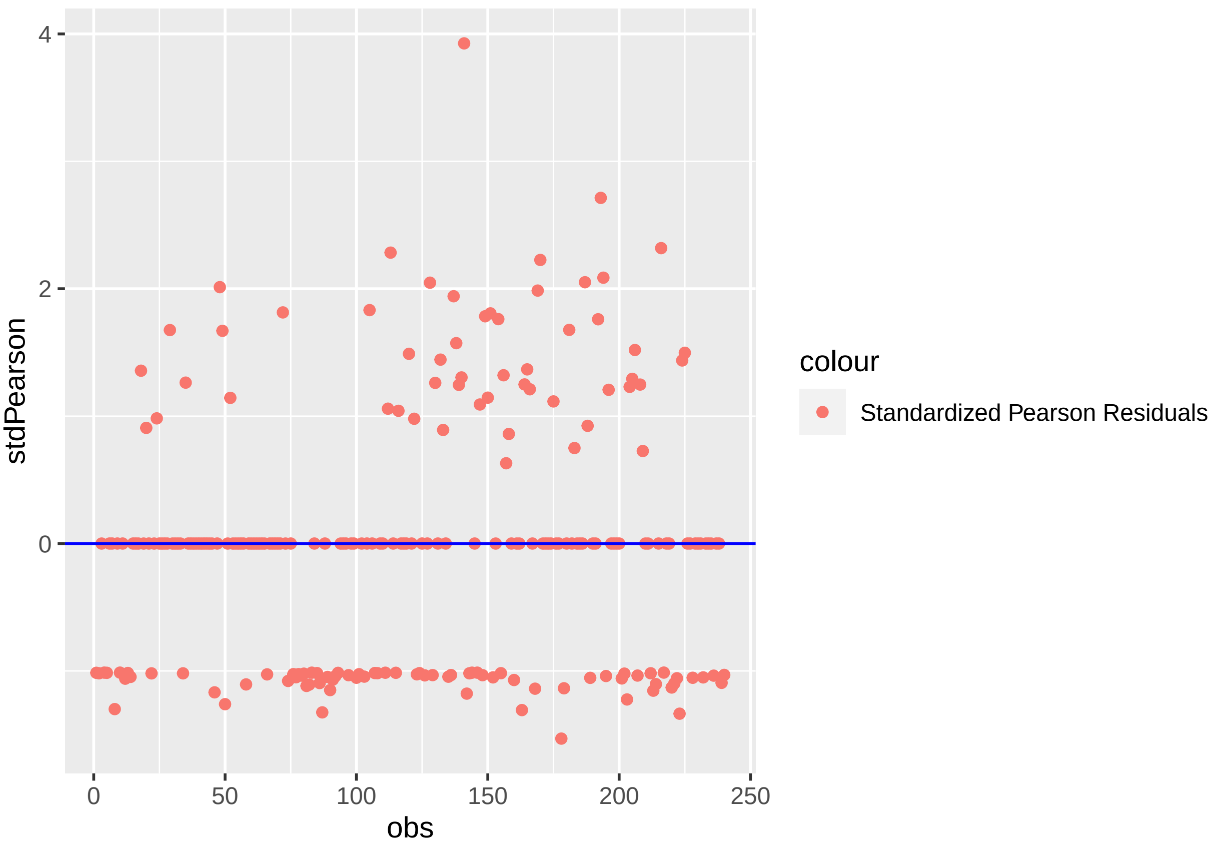
```
diagnosticsVGLM <- function(mod, dat) {
  hatvals <- as.matrix(hatvalues(mod))
  residVals <- residuals(mod,type="pearson")
  for (i in 1:3) {
    leverage <- hatvals[,i]
    rp <- residVals[,i]
    rps <- rp/sqrt(1-leverage)
    cooksD <- leverage/(ncol(dat)*(1-leverage))*rps^2
    leverage.df <- data.frame(obs = 1:nrow(dat),
                              Leverage=leverage,stdPearson=rps,cooksD=cooksD)

    p1 <- ggplot(data=leverage.df, aes(x=obs))+geom_line(aes(y=Leverage, colour="Leverage"))
    plot(p1)

    p2 <- ggplot(data=leverage.df, aes(x=obs))+geom_line(aes(y=cooksD, colour="Cook'sDistance"))
    plot(p2)

    p3 <- ggplot(data=leverage.df, aes(x=obs))+geom_point(aes(y=stdPearson, colour="Standardized Pearso
    plot(p3)
  }
}
```

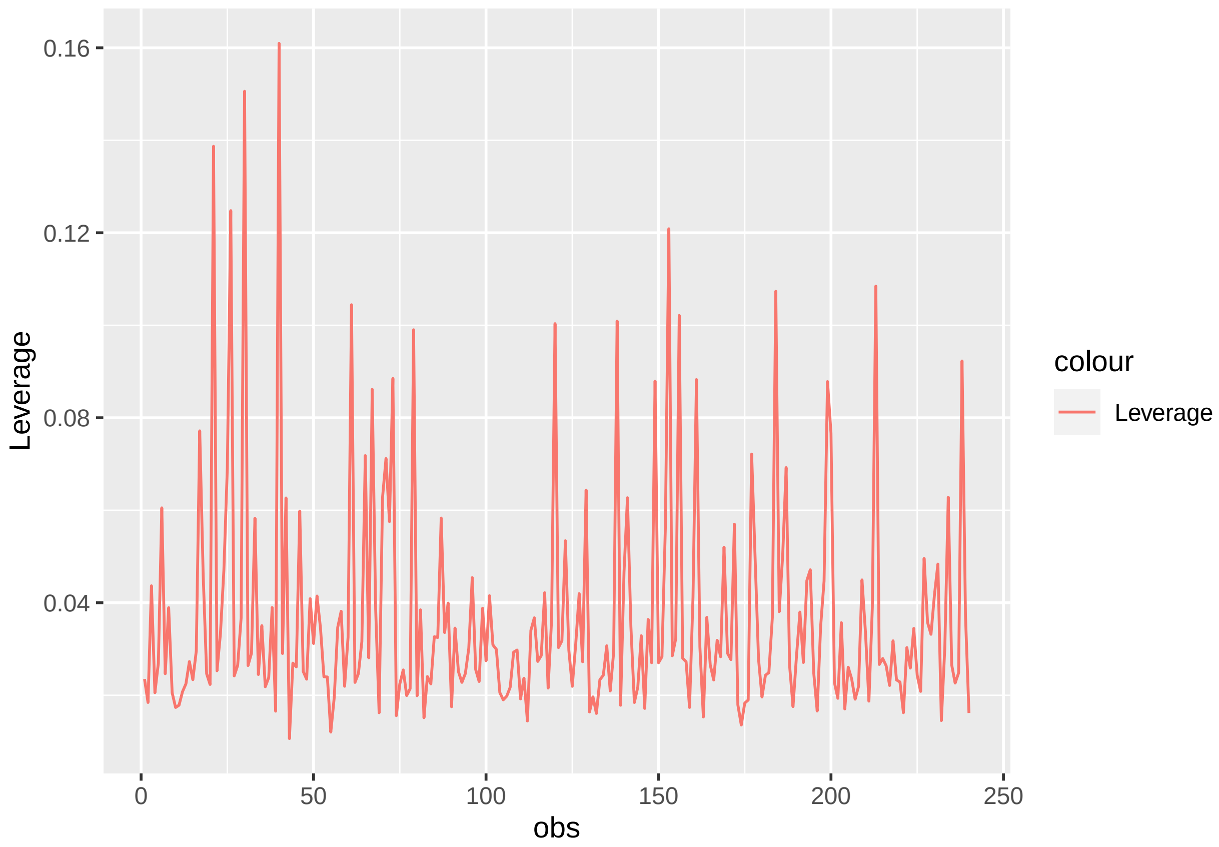
### Proportional odds model diagnostics

```
diagnosticsVGLM(m3, df3)
```

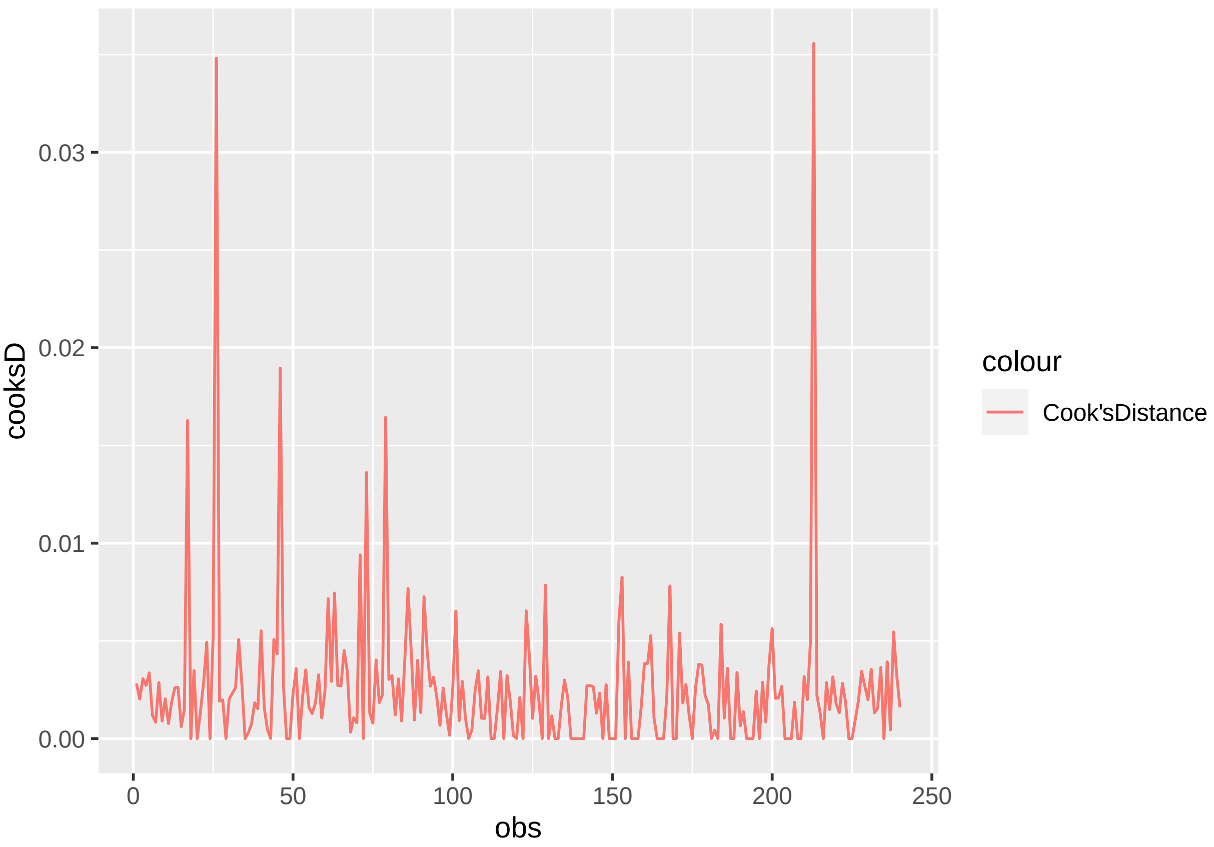


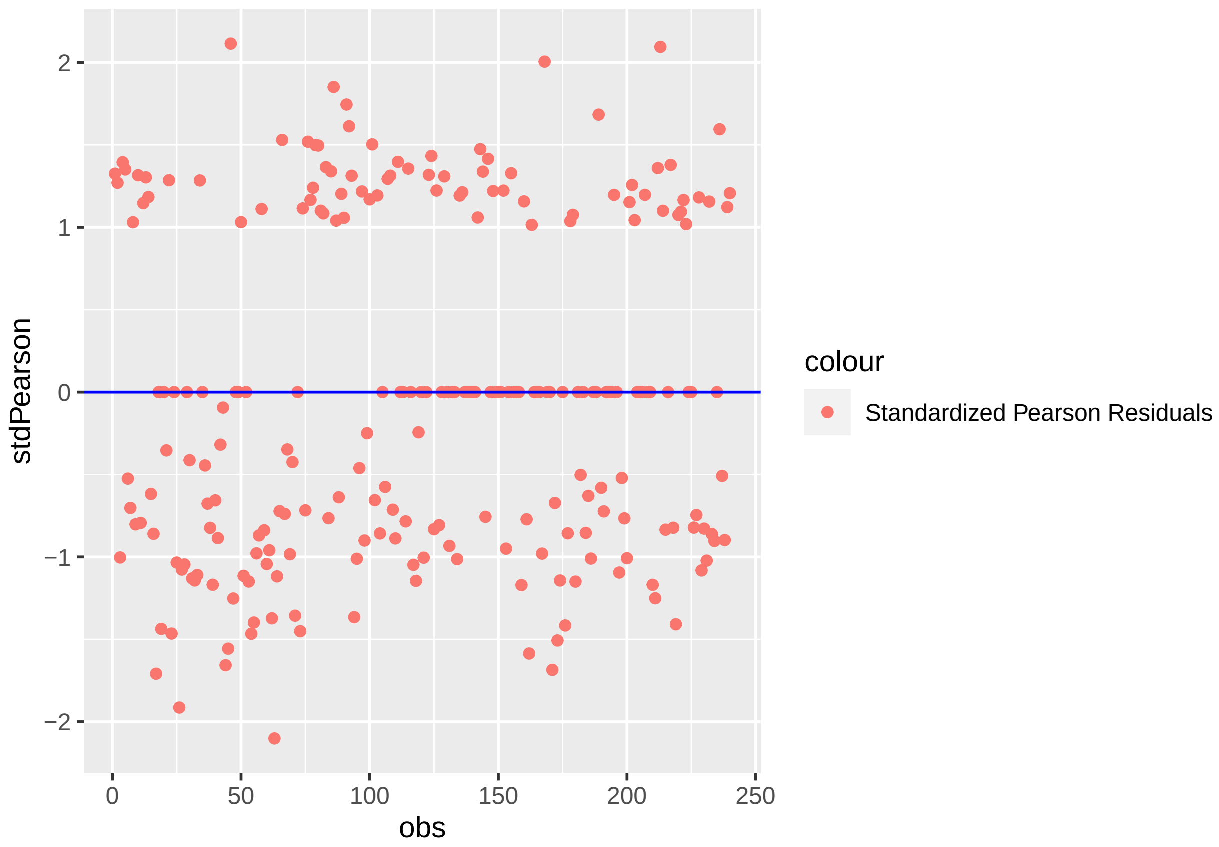

0.020
0.015
0.010
0.005
0.000
cooksD
0
50
100
150
200
250
obs
colour
Cook'sDistance

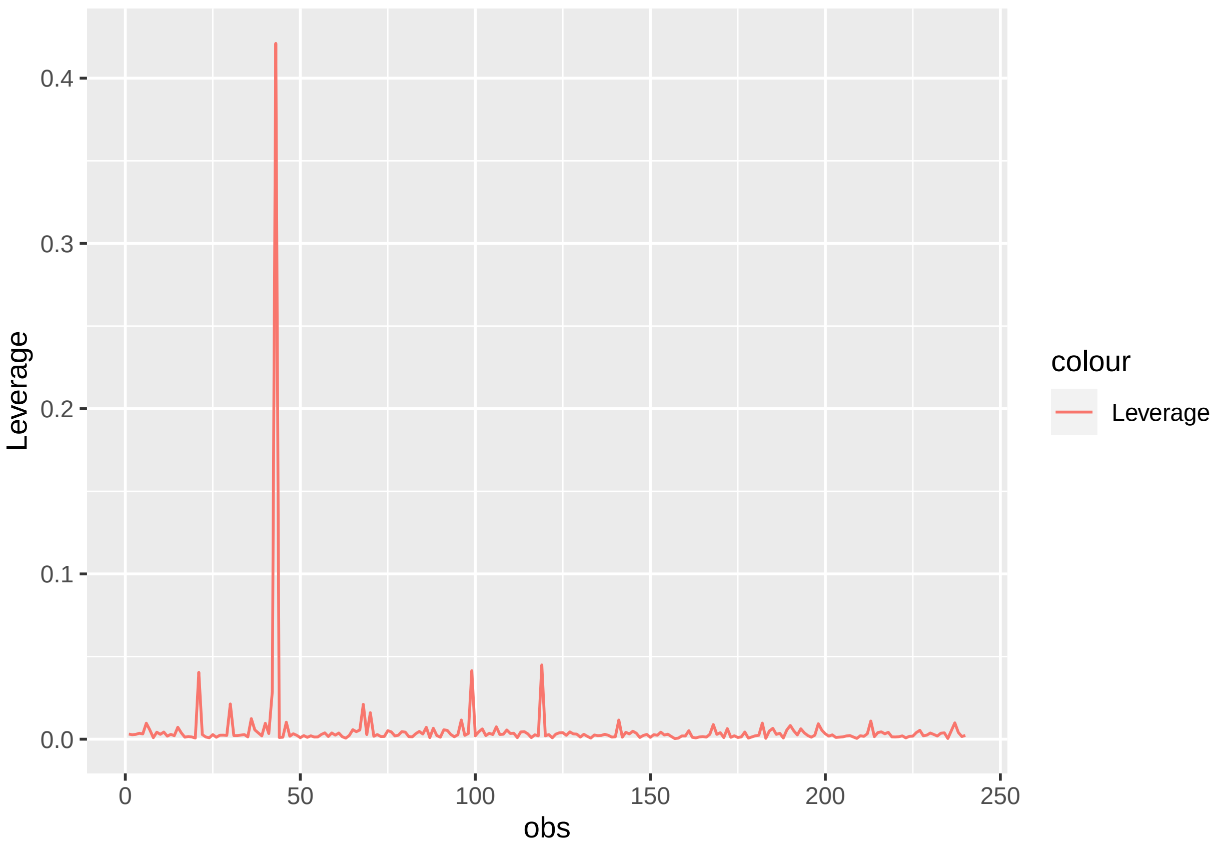

4
2
0
stdPearson
0
50
100
150
200
250
obs
colour
● Standardized Pearson Residuals

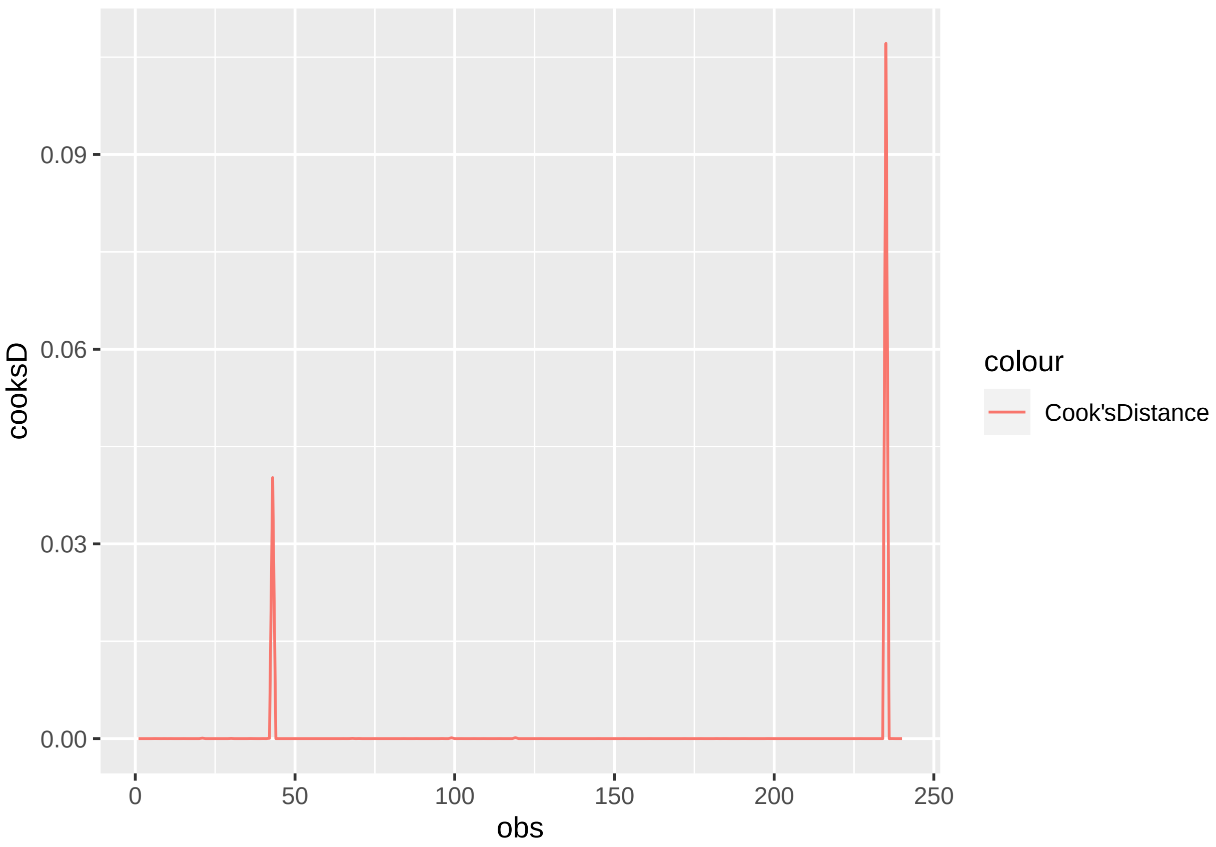


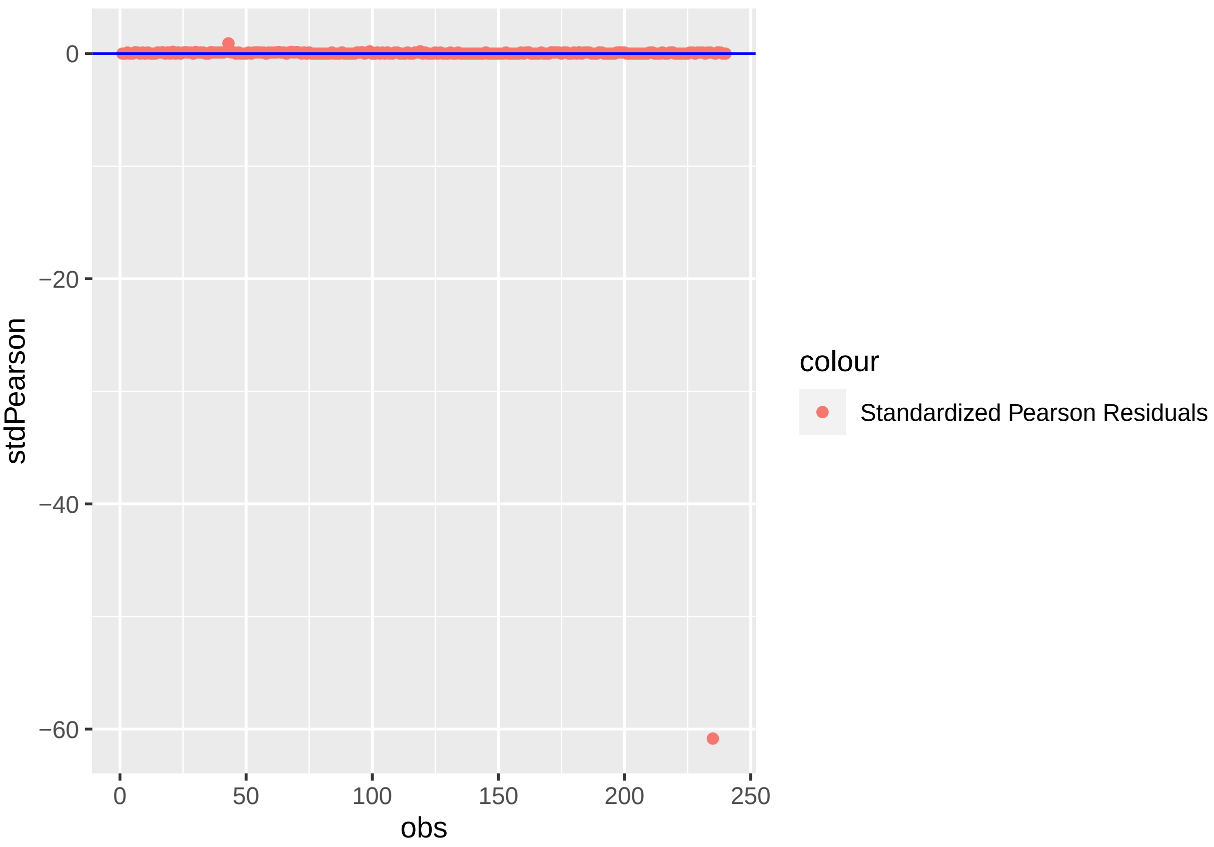
Leverage
0.16
0.12
0.08
0.04
0
50
100
150
200
250
obs
colour
Leverage


cooksD
0.03
0.02
0.01
0.00
0
50
100
150
200
250
obs
colour
Cook'sDistance


stdPearson
2
1
0
−1
−2
0
50
100
150
200
250
obs
colour
● Standardized Pearson Residuals


0.4
0.3
0.2
0.1
0.0
Leverage
0
50
100
150
200
250
obs
colour
Leverage


0.09
0.06
0.03
0.00
cooksD
0
50
100
150
200
250
obs
colour
Cook'sDistance


0
−20
−40
−60
stdPearson
0
50
100
150
200
250
obs
colour
● Standardized Pearson Residuals