Fit Error in Variables Model to Stage Specific RNAseq Data and ROC Phi

Michael A. Gilchrist

21 Jul 2020

ESS Commands

See (~/Software/R/r.notes.Rmd) for more details

Purpose

- Fit stage specific measurements of gene expression to phi values estimated by ROC.

Results So Far

- Have restricted genes in the 20% to 80%-ile of the phi values estimated by ROC. Based on Alex Cope suggested restricting data to genes where the counts are 'not low', e.g. > 100 in every stage if we were working with FPKM (as I understand it).
- Initially using simple polynomial models.
  - Initial fits were poor
    - negative coefficients
    - highly structured residuals that declined linearly with the predictor. (See figure for linear `lm()` fit)
  - Residuals suggest a second order polynomial will work well.
- Could see if log(phi) is better predicted using the geometric mean (though a 0 would mess things up).
- May eventually fit error in variables regression approach since there is substantial error in both predictor (RNAseq counts) and response variable (ROC's ϕ).

Load Libraries

```
##library(Biostrings) ## process first to avoid conflicts
## May want to set library(verbose=TRUE)
library(tidyr)
library(tibble)
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
library(stringr)
library(forcats)
library(ggplot2)
library(knitr)
library(ggpubr)
library(ggpmisc)
library(gridExtra)
##
## Attaching package: 'gridExtra'
## The following object is masked from 'package:dplyr':
##
##     combine
library(purrr)
library(eivtools)
## Loading required package: R2jags
## Loading required package: rjags
## Loading required package: coda
## Linked to JAGS 4.3.0
## Loaded modules: basemod,bugs
##
## Attaching package: 'R2jags'
## The following object is masked from 'package:coda':
##
##     traceplot
creationInfo <- paste0("\tDate: ", date(), "\n\tLocation: ", sub(".*/AcrossTissue", "AcrossTissue", getw

#exportData=TRUE ## Flag for running save(), write.csv() and other output commands
```

Load and Filter Data

```
## Define file names
emtabDataFile <- "Input/processed.E-MTAB.data.Rdata"
isoformSummaryStatsFile <- "Input/ROC.isoform.summary.stats-WB.fasta.sphi.equal.2.8.Rdata"
##embryoStageCountMomentsFile <- "Input/summary.stats.of.embryo.stage.Rdata"


##labeledPhiFile <- "Input/ROC_labeled.phi.summaries.with.sphi.equal.2.8.csv"

## Load E-MTAB Data
```

```
load(file=emtabDataFile) ## lifeStageCount and lifeStage

lifeStageCount <- rename(lifeStageCount,
                         WormBase.ID = WBID)

## $\phi$ Data
load(isoformSummaryStatsFile)
comment(isoformSummaryStats)
```

```
##  [1] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includeWorml
##  [2] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemean_
##  [3] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includesd_ph
##  [4] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemean_
##  [5] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemean_
##  [6] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemean_
##  [7] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemean_
##  [8] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemean_
##  [9] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemean_
## [10] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemedia
## [11] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemedia
## [12] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemedia
## [13] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemedia
## [14] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includemedia
## [15] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includen_iso
## [16] "Various summary stats of ROC phi output across a WormBase gene's isoforms. Columns includen_iso
```

Plot ROC Isoform Data

- Should probably reside some where else
- May already exist in Plot.ROC...Rmd files

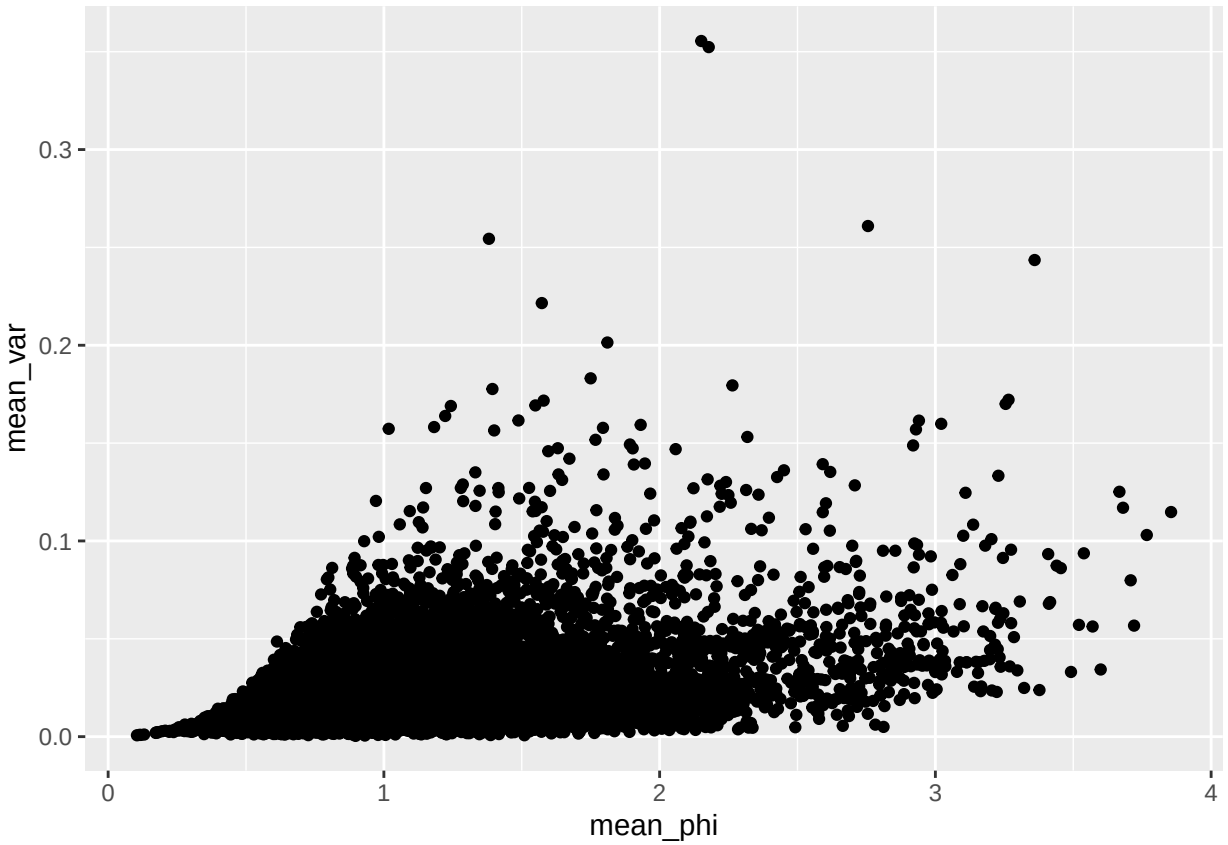
```
selectStats  <- select(isoformSummaryStats, c("WormBase.ID", "mean_phi", "mean_var"))


## only retain genes with matches in both datasets
## NOTE:
meanLog10MeanPhi <- mean(log10(selectStats$mean_phi))
sdLog10MeanPhi <- sd(log10(selectStats$mean_phi))

qplot(data = selectStats, x=mean_phi, log = "x") # +
##    annotate(geom = geom_vline(aes(xintercept = meanLog10MeanPhi) ) ) #+
##    annotate(geom = geom_segment(x = meanLog10MeanPhi, xend = meanLog10MeanPhi + sdLog10MeanPhi, y = 

qplot(data = selectStats, x=mean_var, log = "y")
```

Reshape E-MTAB Data

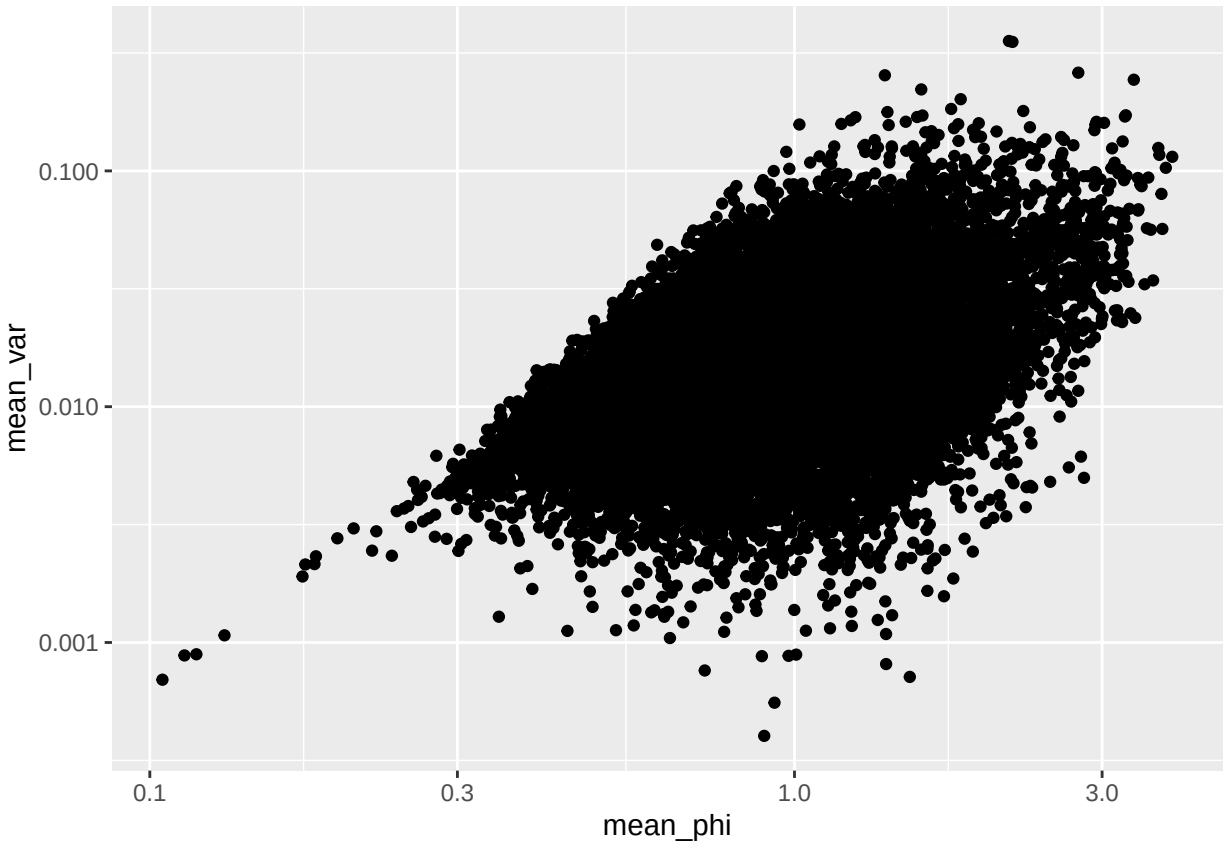
```
tallData <- lifeStageCount %>%
    filter(grepl("hermaphrodite", sex)) ##%>%
    ## Ensure average for each set of data is 1
```

```
    ## Should update scaledCount later when genes are filtered based on Phi values
##    mutate(scaledCount=(count/sum(count)*), totalCount=sum(count)) %>%
##    group_by(stage) %>%
##    semi_join(filteredWBID) %>% select(-c(tissue, sex, count, totalCount))

wideData <- pivot_wider(tallData, names_from=stage, values_from=count )

## Update column names
names(wideData) <-
    names(wideData) %>%
    str_replace_all(' Ce', '') %>%
    str_replace_all(' larva', '') %>%
    str_replace_all(' stage', '') %>%
    str_replace_all(' ', '.')

stages <- lifeStages %>%
    str_replace_all(' Ce', '') %>%
    str_replace_all(' larva', '') %>%
    str_replace_all(' stage', '') %>%
    str_replace_all(' ', '.')
```

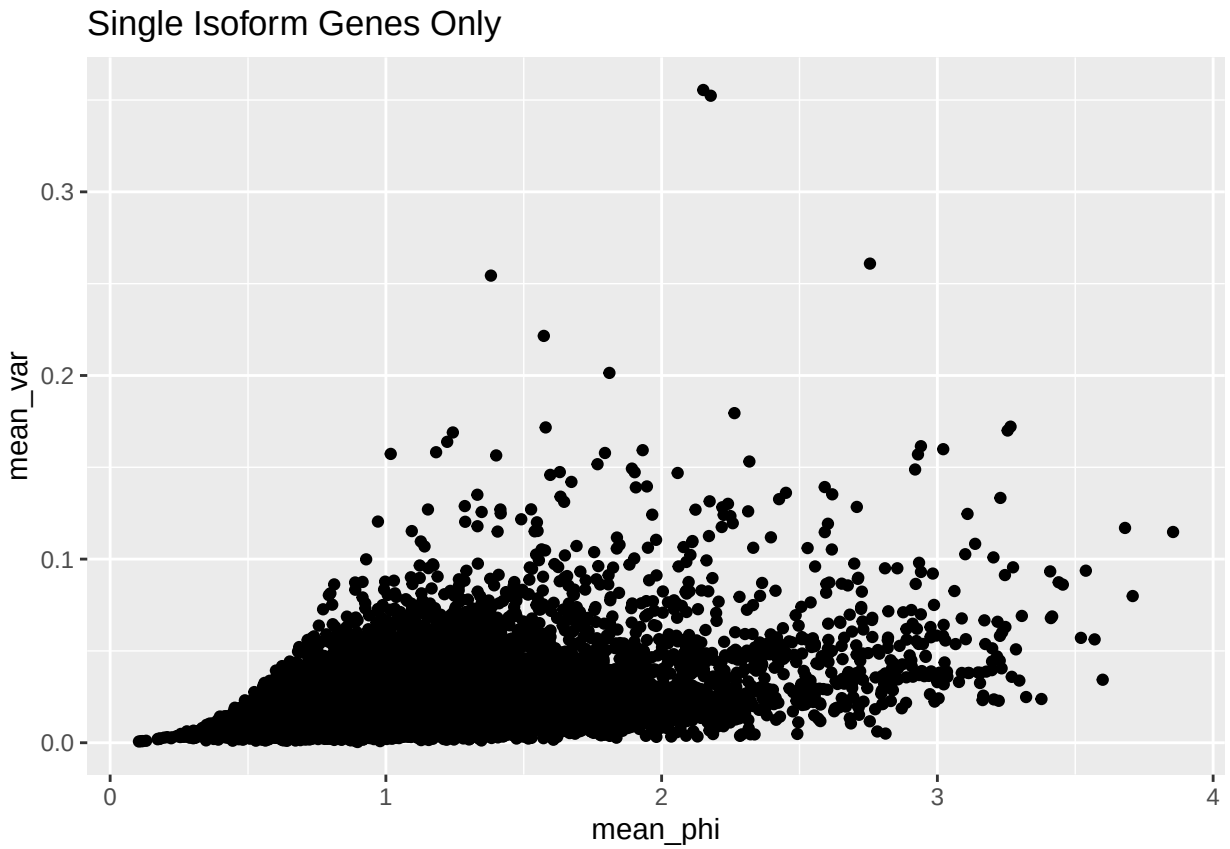
```
## Find genes in common between ROC and E-MTAB data
## Old: dataForModel <- inner_join(wideData, wtPhiData, by = "geneName")
geneData <- inner_join(wideData, isoformSummaryStats, by = "WormBase.ID")
singleIsoformGeneData <- filter(geneData, n_isoforms==1)

comment(geneData) = "tibble with data from E-MTAB file (single embryo stage) and the summary stats of pl

qplot(x = mean_phi, y = mean_var, data = geneData)
```



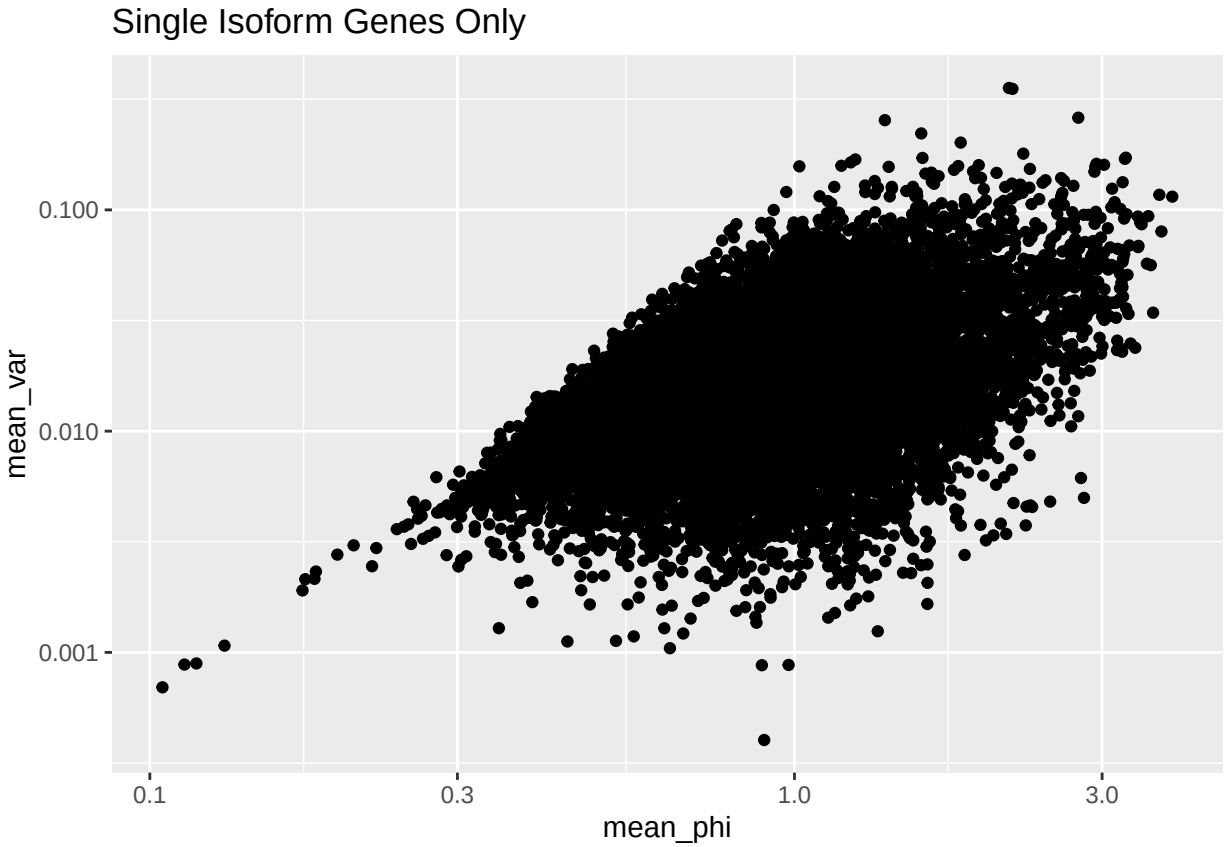

```
qplot(x = mean_phi, y = mean_var, data = geneData, log = "xy")
```




```
qplot(x = mean_phi, y = mean_var, data = singleIsoformGeneData,
      main="Single Isoform Genes Only")
```




```
qplot(x = mean_phi, y = mean_var, data = singleIsoformGeneData, log = "xy",
      main="Single Isoform Genes Only")
```



Filter Genes with Extreme Phi Values

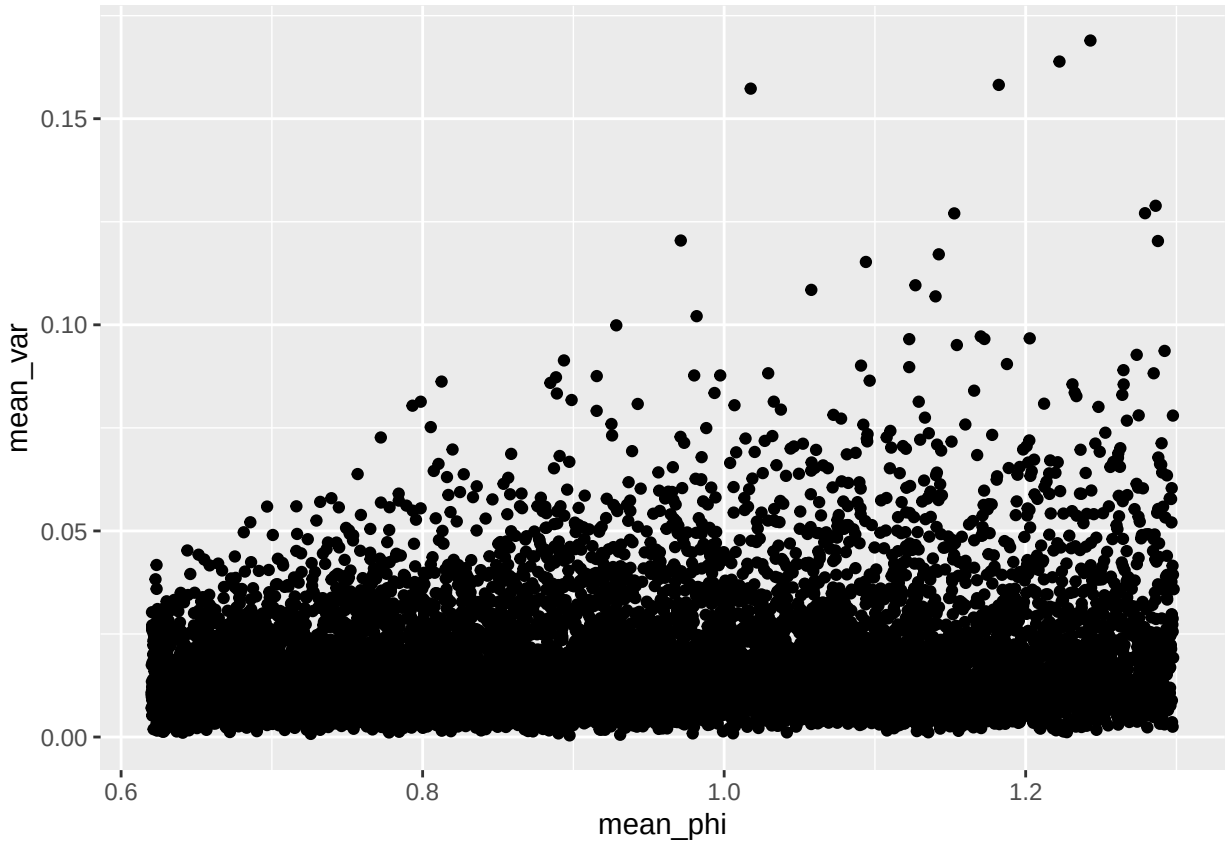
```
myPhiPercentileRange <- c(0.2, 0.8)
filterPhiRange <- quantile(x = geneData[["mean_phi"]], probs= myPhiPercentileRange)


## Filter data based on phi values
tmpGeneData <- filter(geneData, mean_phi > filterPhiRange[1] & mean_phi < filterPhiRange[2])

## Scale counts for each E-MTAB stage such that the mean is 1
filteredGeneData <-
    tmpGeneData %>%
    mutate(across(stages,  ~(.x/sum(.x)*length(.x)) ) )
```

```
## Note: Using an external vector in selections is ambiguous.
## i Use `all_of(stages)` instead of `stages` to silence this message.
## i See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

```
summarise(filteredGeneData, across(stages, mean))
```

```
## # A tibble: 1 x 8
##   embryo    L1    L2    L3    L4 adult dauer post.dauer
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>      <dbl>
## 1      1     1     1     1     1  1.00     1          1
```

```
qplot(x = mean_phi, y = mean_var, data = filteredGeneData)
```

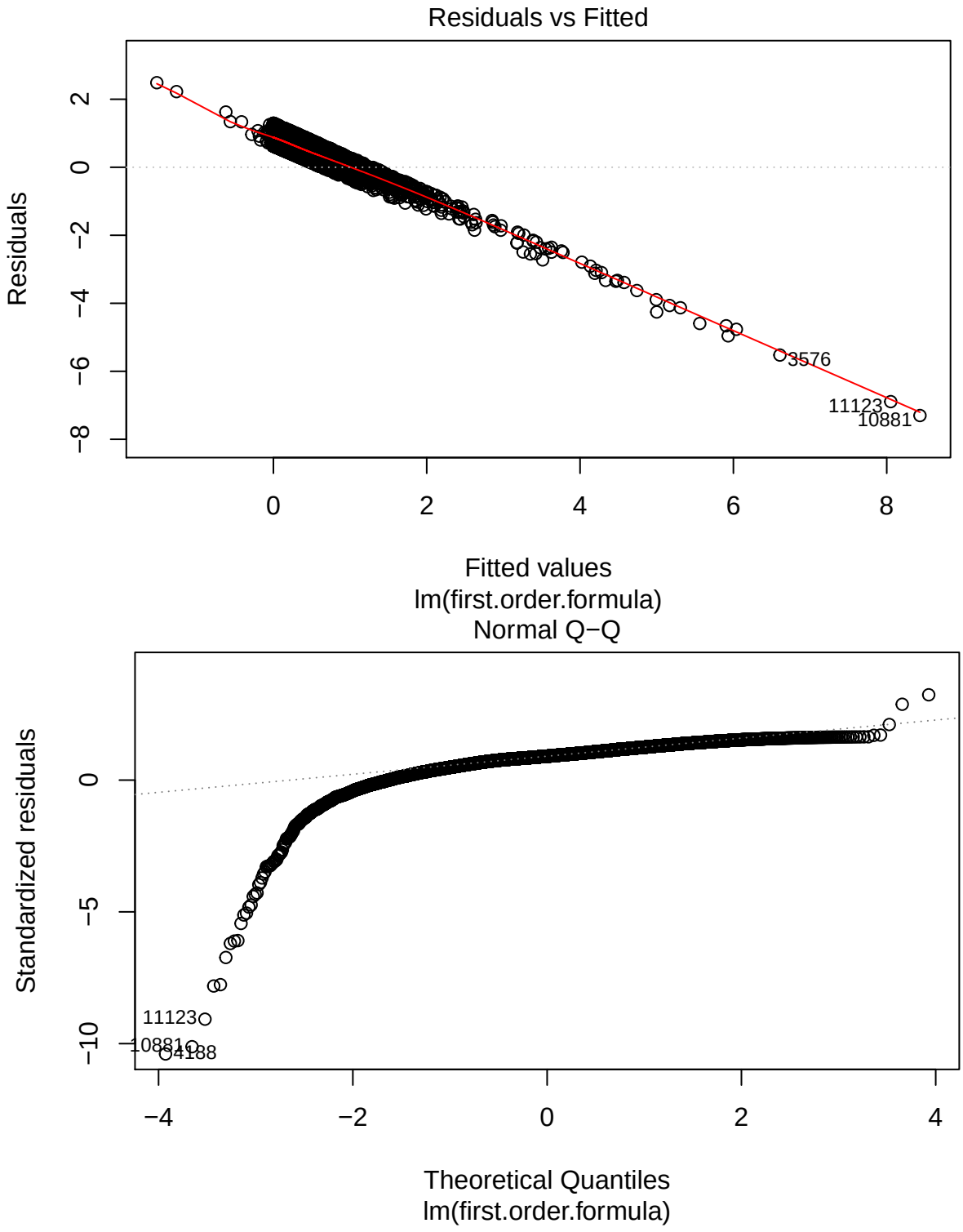


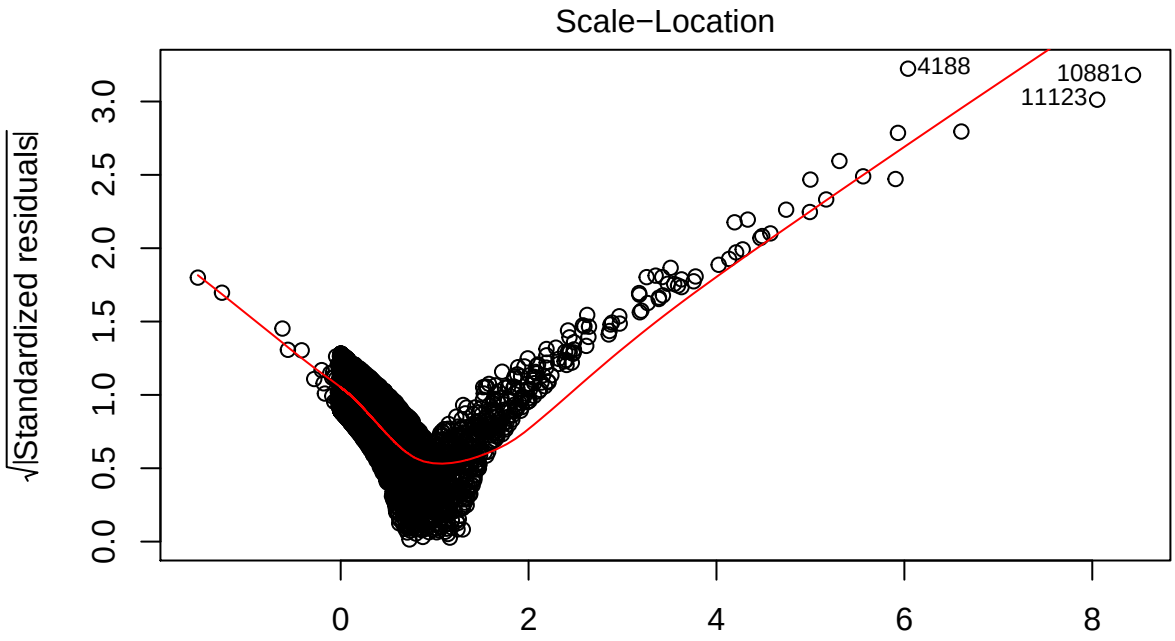
Fit model

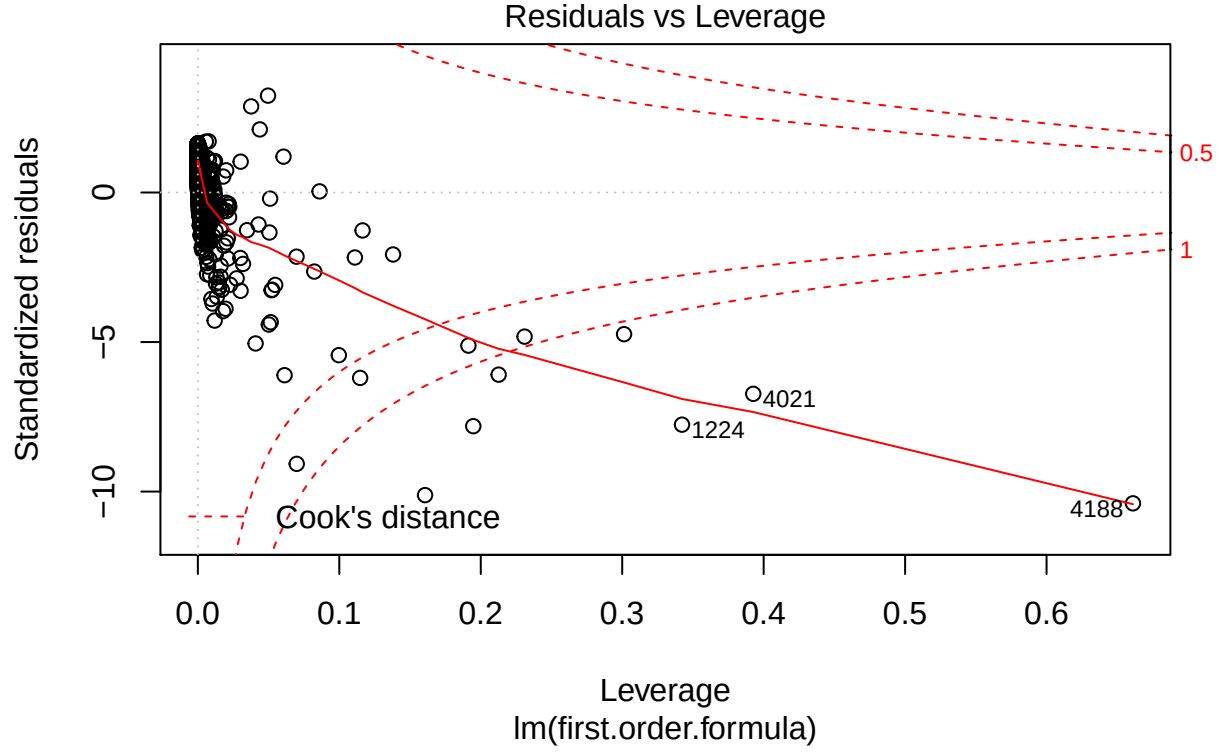
Linear Model

- Note the crazy residuals.

```
## Taken from https://ryouready.wordpress.com/2009/02/06/r-calculating-all-possible-linear-regression-m
## as.formula(paste(c("y ~ 1", regressors[vec]), collapse=" + "))
first.order.formula<- as.formula(paste(c("mean_phi ~ -1", stages), collapse = " + "))

firstOrderPhiFit  <- lm(first.order.formula, filteredGeneData)


## Residuals are linear with the response variable!!
plot(firstOrderPhiFit)
```


Residuals vs Fitted
Residuals
Fitted values
lm(first.order.formula)
3576
11123
10881
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(first.order.formula)
11123
10881
4188






```
summary(firstOrderPhiFit)
```

```
##
## Call:
## lm(formula = first.order.formula, data = filteredGeneData)
##
## Residuals:
```

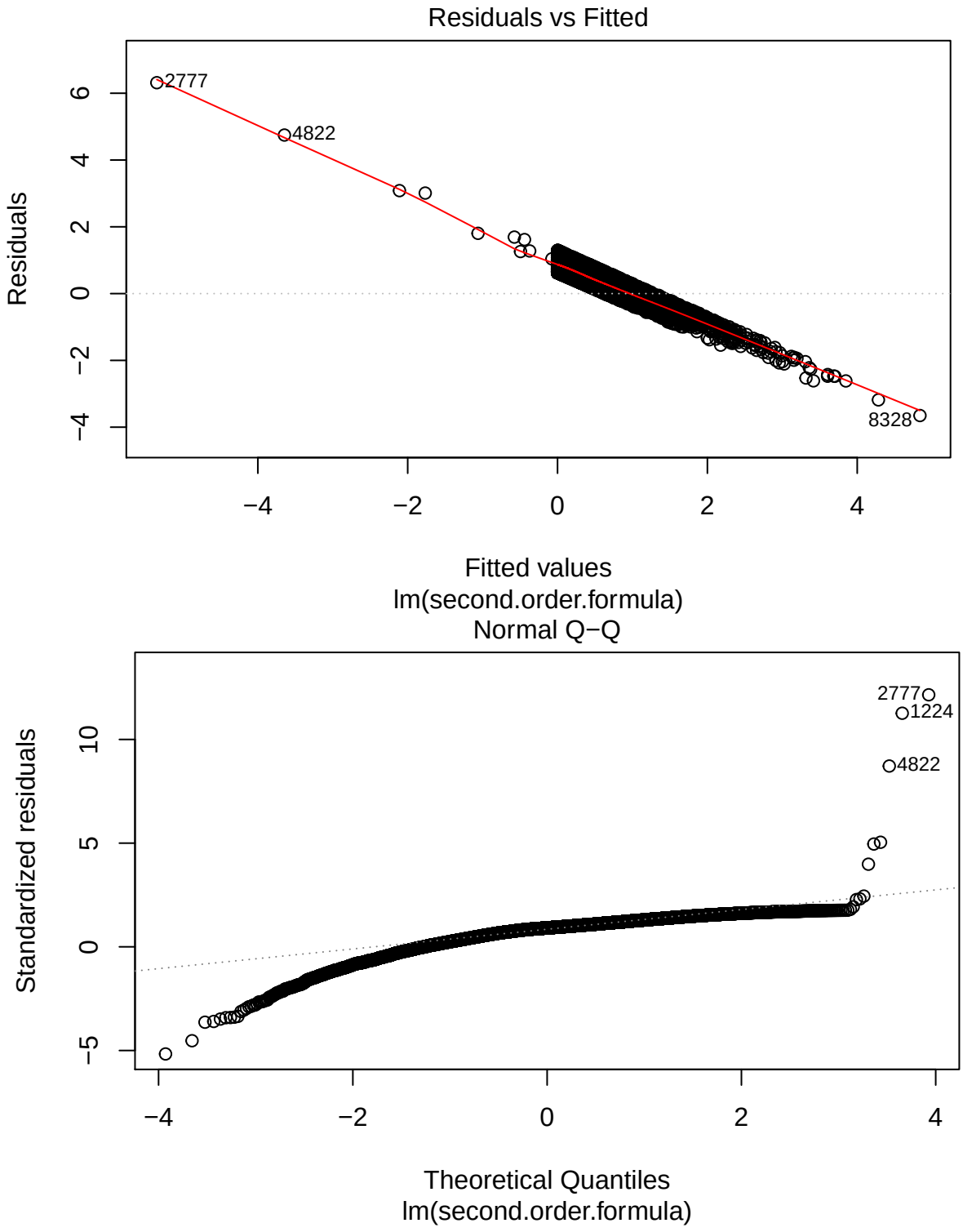
```
##     Min      1Q  Median      3Q     Max
## -7.3010  0.5326  0.7158  0.8969  2.4863
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## embryo      0.068461   0.004070  16.822  < 2e-16 ***
## L1          0.010119   0.002174   4.655 3.28e-06 ***
## L2          0.084720   0.007351  11.525  < 2e-16 ***
## L3          0.015925   0.005935   2.683   0.0073 **
## L4          0.009090   0.003711   2.450   0.0143 *
## adult       0.039206   0.003172  12.358  < 2e-16 ***
## dauer      -0.041717   0.004611  -9.047  < 2e-16 ***
## post.dauer  0.069863   0.006186  11.294  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7876 on 11678 degrees of freedom
## Multiple R-squared:  0.2872, Adjusted R-squared:  0.2867
## F-statistic: 588.2 on 8 and 11678 DF,  p-value: < 2.2e-16
```

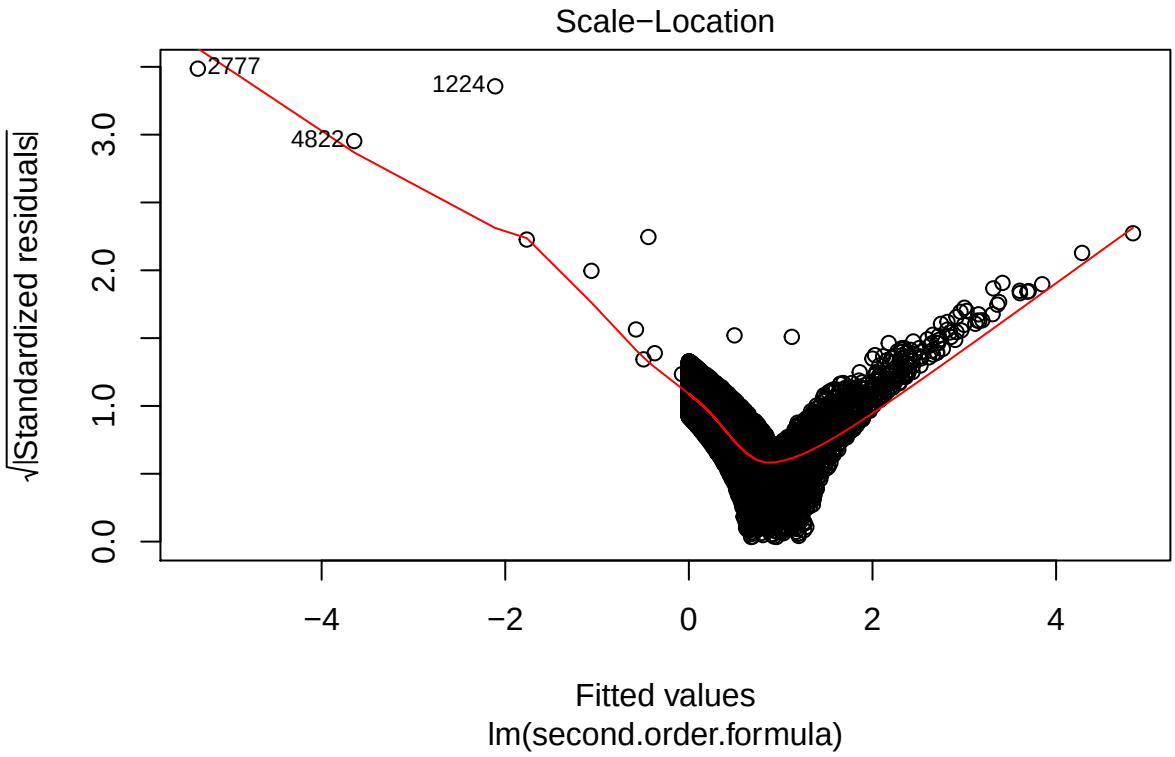
```
## Need to do a contrained fit where
## sum(coefficients) =1 and coefficients > 0
```

Quadratic Model Fit

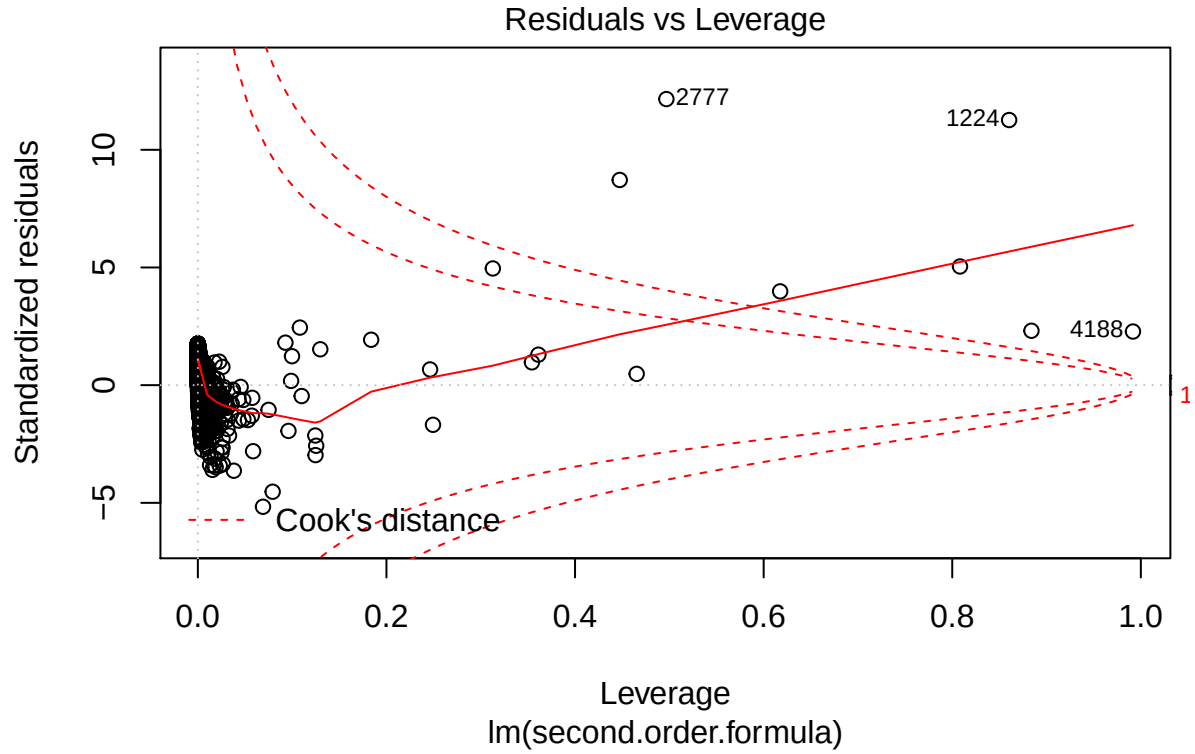
- Still have crazy residuals. As a result, I'm confused; why didn't this drastically improve the residuals?

```
tmp <- paste(paste0("I(", stages, "^2)"))

second.order.formula<- as.formula(paste(c("mean_phi ~ -1", stages, tmp), collapse = " + "))


secondOrderPhiFit  <- lm(second.order.formula, filteredGeneData)
plot(secondOrderPhiFit)
```


Residuals vs Fitted
Residuals
Fitted values
lm(second.order.formula)
2777
4822
8328
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(second.order.formula)
2777
1224
4822




```
## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced

## Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced
```

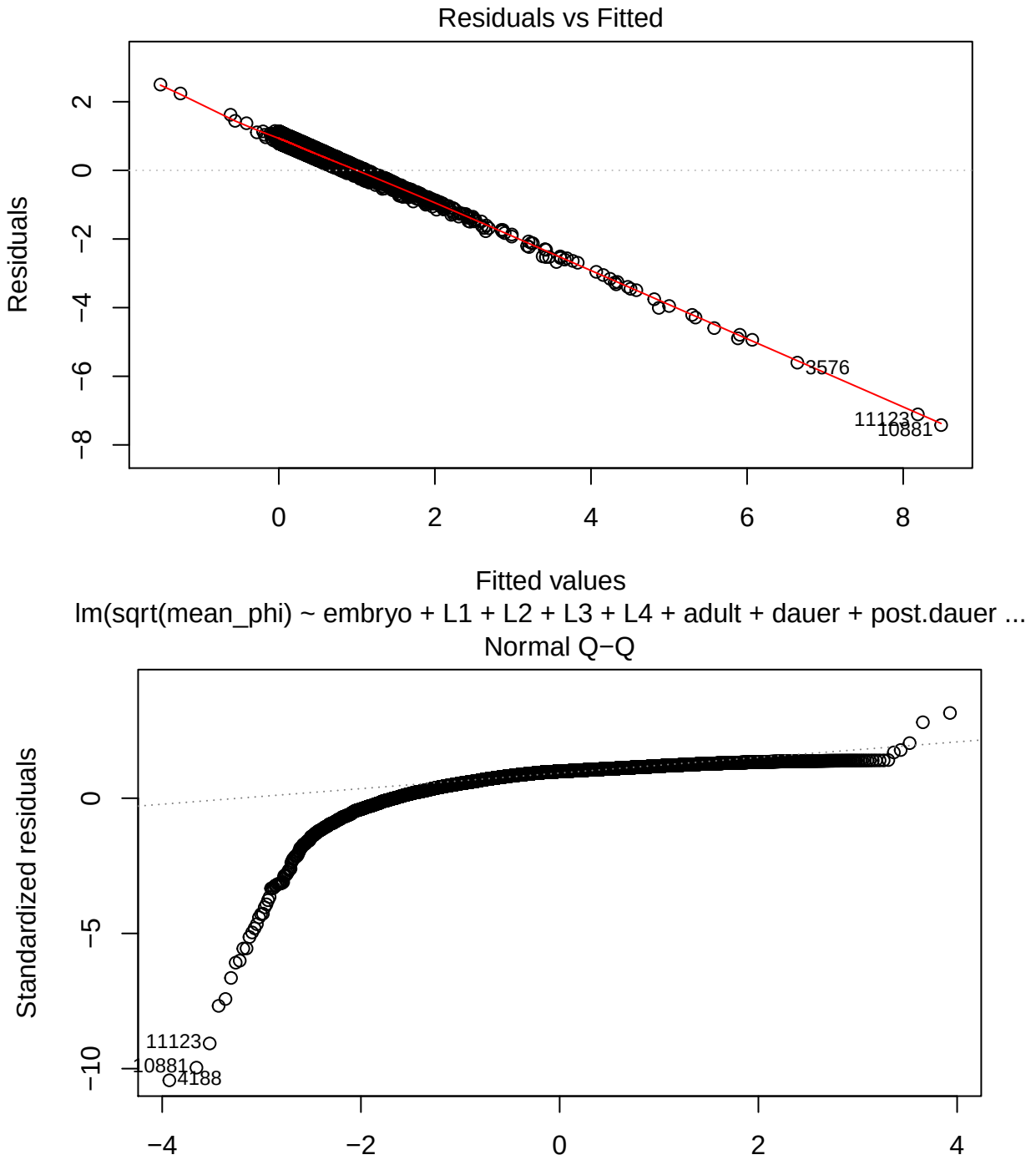


```
summary(secondOrderPhiFit)
```

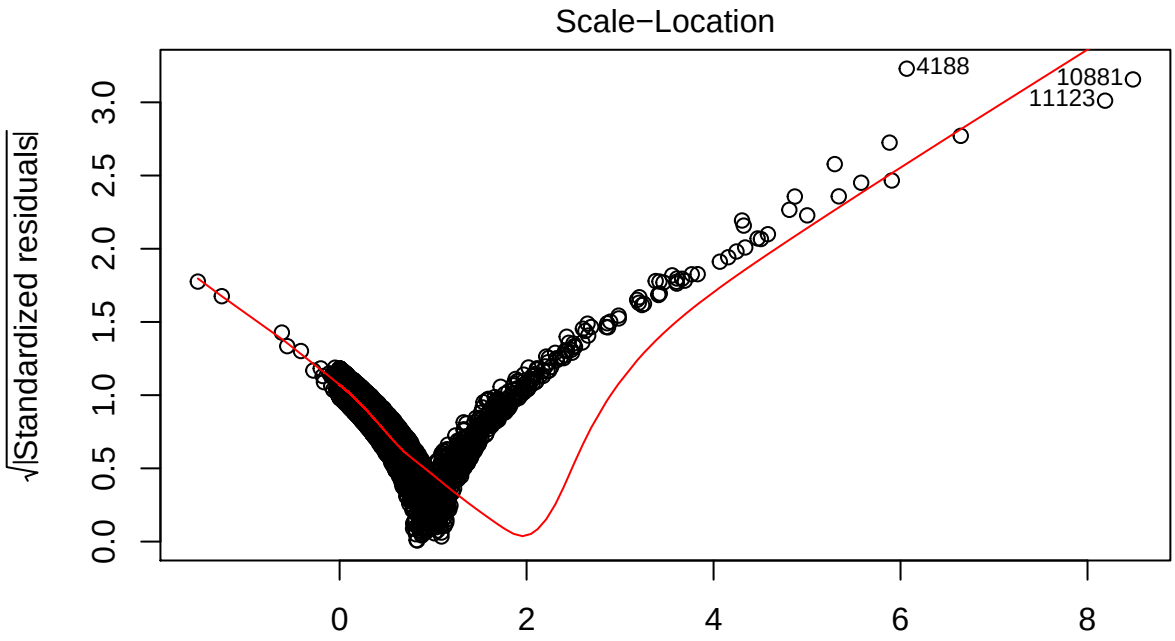
```
##
## Call:
## lm(formula = second.order.formula, data = filteredGeneData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6515  0.3851  0.6669  0.8541  6.3152
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## embryo           1.039e-01  5.975e-03  17.385  < 2e-16 ***
## L1               2.065e-02  3.737e-03   5.525 3.37e-08 ***
## L2               8.102e-02  1.057e-02   7.663 1.96e-14 ***
## L3               2.707e-02  9.358e-03   2.893 0.003823 **
## L4               5.720e-02  6.057e-03   9.443  < 2e-16 ***
## adult            5.417e-02  4.861e-03  11.143  < 2e-16 ***
## dauer           -6.192e-03  7.628e-03  -0.812 0.416933
## post.dauer       6.151e-02  9.927e-03   6.196 5.98e-10 ***
## I(embryo^2)     -2.680e-03  1.707e-04 -15.696  < 2e-16 ***
## I(L1^2)         -2.156e-04  2.850e-05  -7.562 4.25e-14 ***
## I(L2^2)         -4.056e-03  3.176e-04 -12.770  < 2e-16 ***
## I(L3^2)         -4.589e-04  1.286e-04  -3.570 0.000359 ***
## I(L4^2)         -4.157e-04  6.033e-05  -6.889 5.89e-12 ***
## I(adult^2)      -6.483e-04  6.556e-05  -9.887  < 2e-16 ***
## I(dauer^2)      -1.455e-06  2.149e-05  -0.068 0.946007
## I(post.dauer^2) -3.120e-04  4.870e-05  -6.407 1.54e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7325 on 11670 degrees of freedom
## Multiple R-squared:  0.3839, Adjusted R-squared:  0.3831
## F-statistic: 454.5 on 16 and 11670 DF,  p-value: < 2.2e-16
```

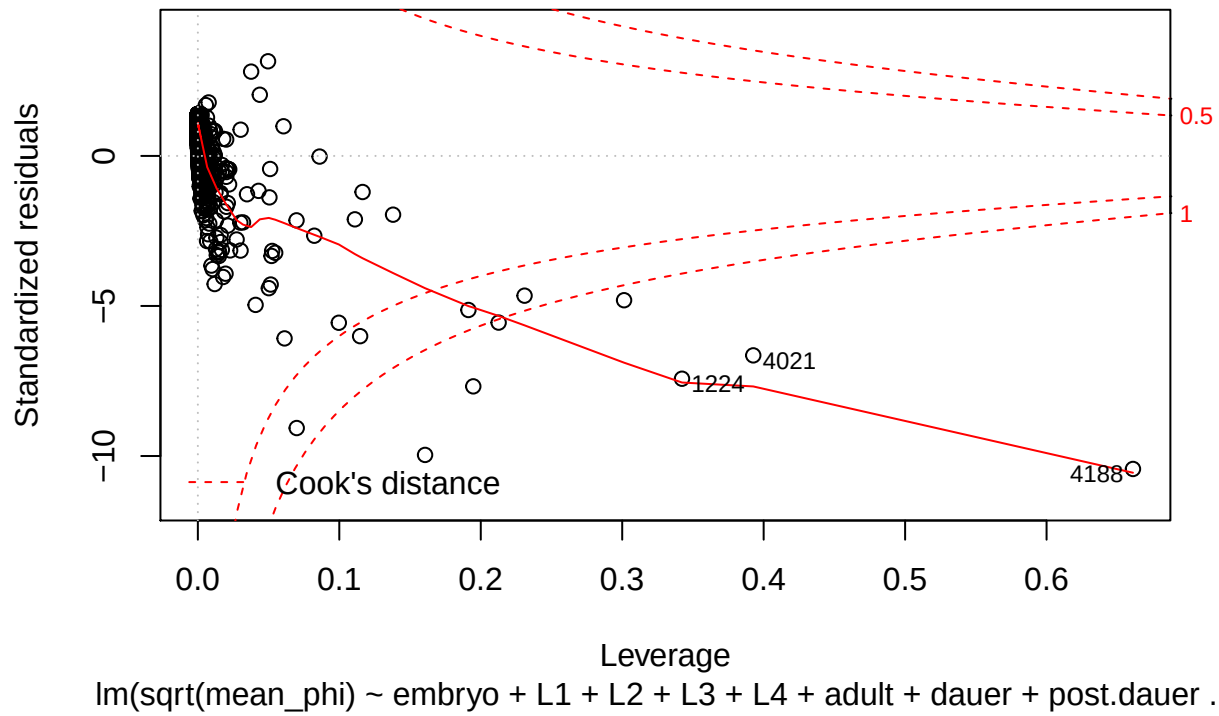
Sqrt Transformation of `mean_phi`

```
sqrtPhiFit  <- lm(sqrt(mean_phi) ~ embryo+L1+L2+L3+L4+adult+dauer+post.dauer-1, filteredGeneData)

plot(sqrtPhiFit)
```


Residuals vs Fitted
Residuals
Fitted values
lm(sqrt(mean_phi) ~ embryo + L1 + L2 + L3 + L4 + adult + dauer + post.dauer ...
3576
11123
10881

Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(sqrt(mean_phi) ~ embryo + L1 + L2 + L3 + L4 + adult + dauer + post.dauer ...
11123
10881
4188





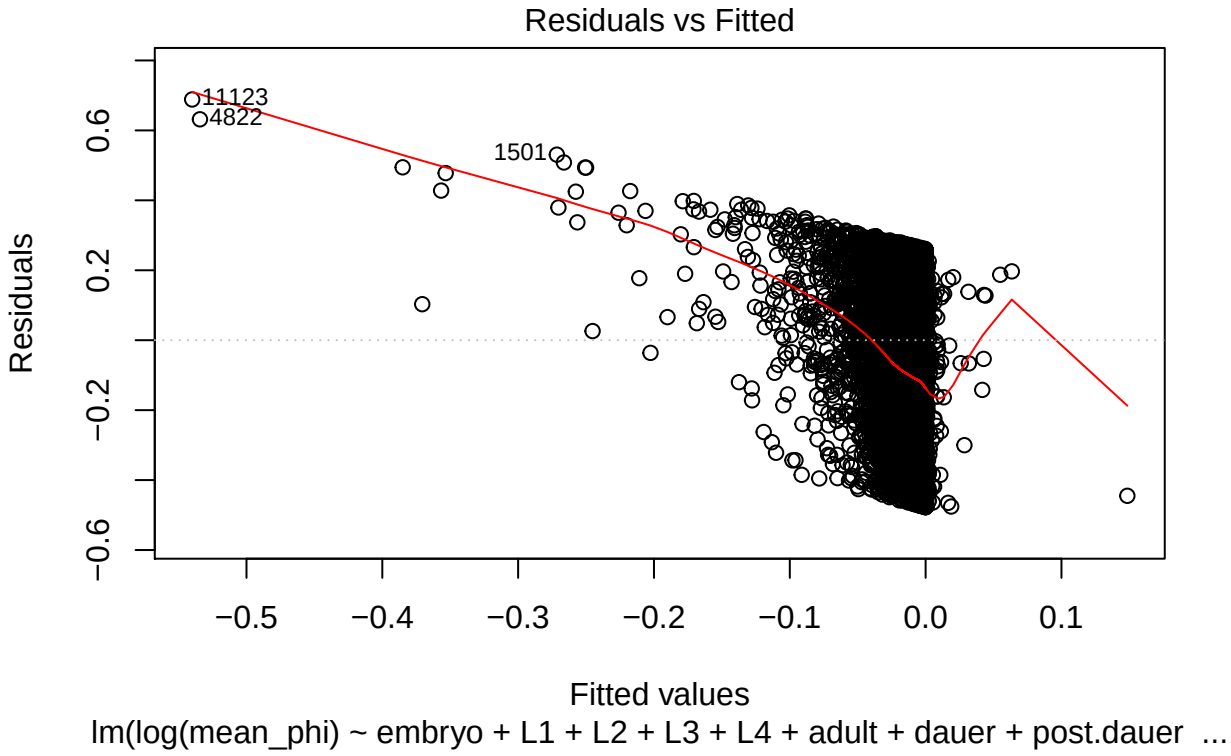

```
summary(sqrtPhiFit)
```

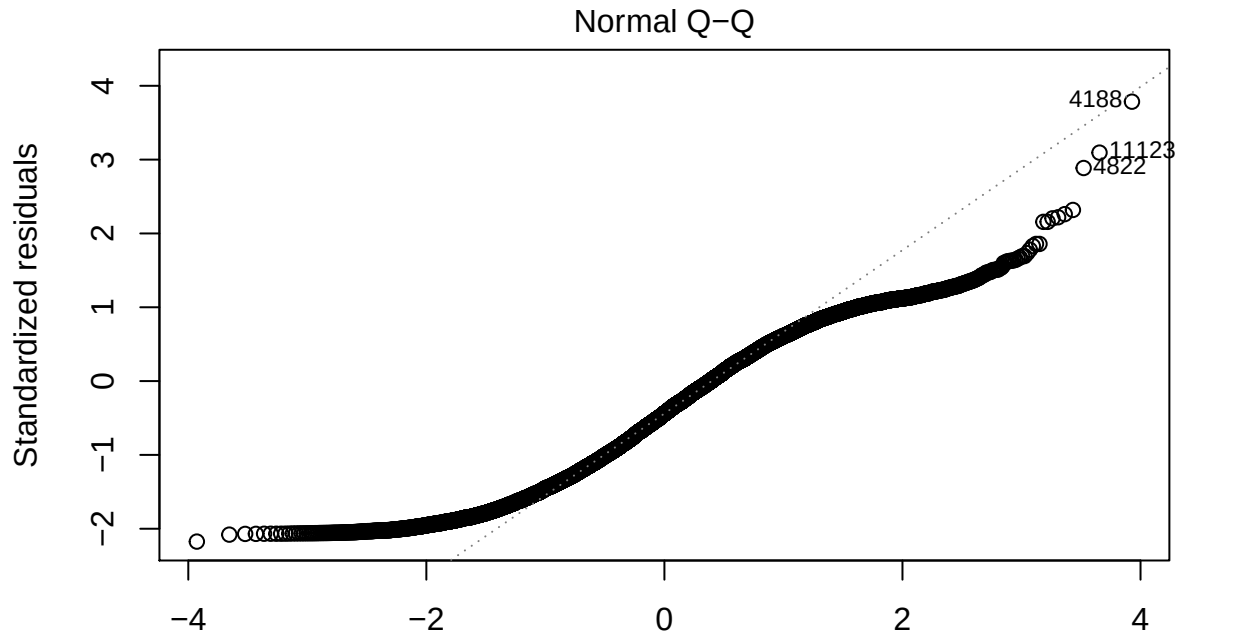
```
##
## Call:
## lm(formula = sqrt(mean_phi) ~ embryo + L1 + L2 + L3 + L4 + adult +
##     dauer + post.dauer - 1, data = filteredGeneData)
```

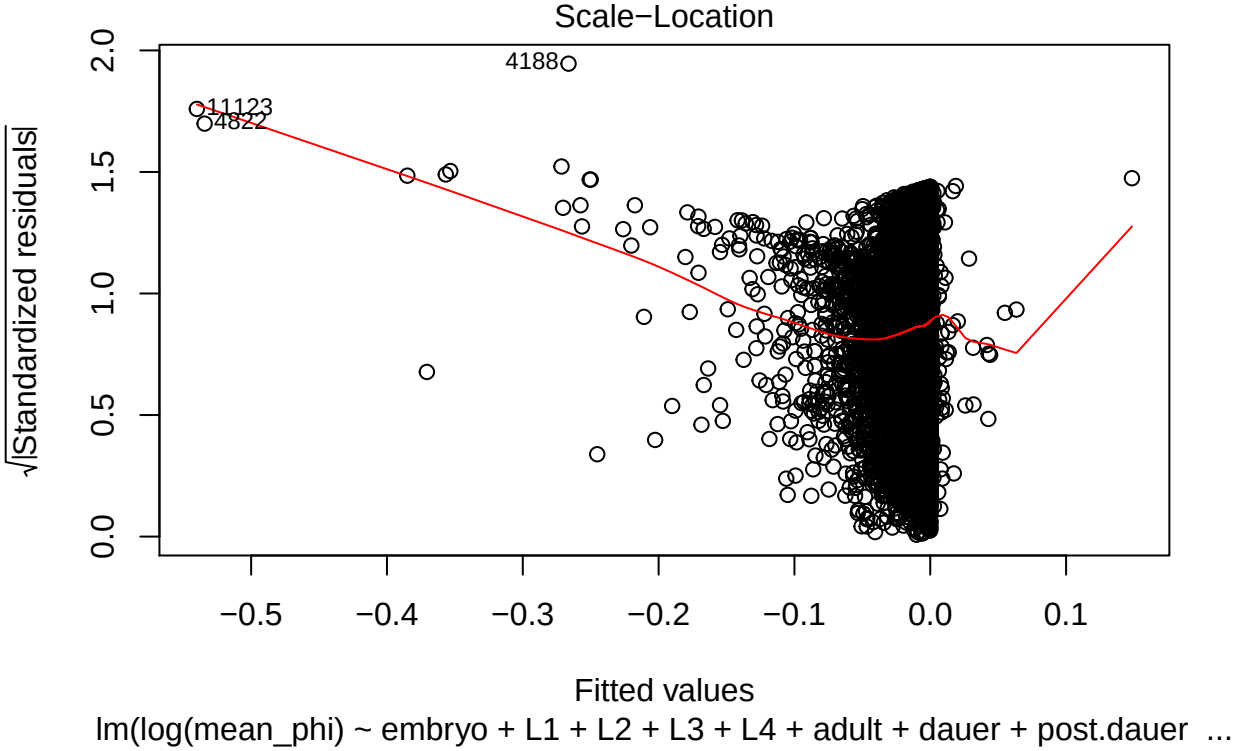
```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.4220  0.5983  0.8058  0.9143  2.4993
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## embryo      0.071223   0.004201  16.953  < 2e-16 ***
## L1          0.010231   0.002244   4.559 5.19e-06 ***
## L2          0.083301   0.007588  10.977  < 2e-16 ***
## L3          0.017521   0.006127   2.860  0.00425 **
## L4          0.008952   0.003830   2.337  0.01945 *
## adult       0.038135   0.003275  11.645  < 2e-16 ***
## dauer      -0.041610   0.004760  -8.741  < 2e-16 ***
## post.dauer  0.069691   0.006386  10.914  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8131 on 11678 degrees of freedom
## Multiple R-squared:  0.2771, Adjusted R-squared:  0.2766
## F-statistic: 559.4 on 8 and 11678 DF,  p-value: < 2.2e-16
```

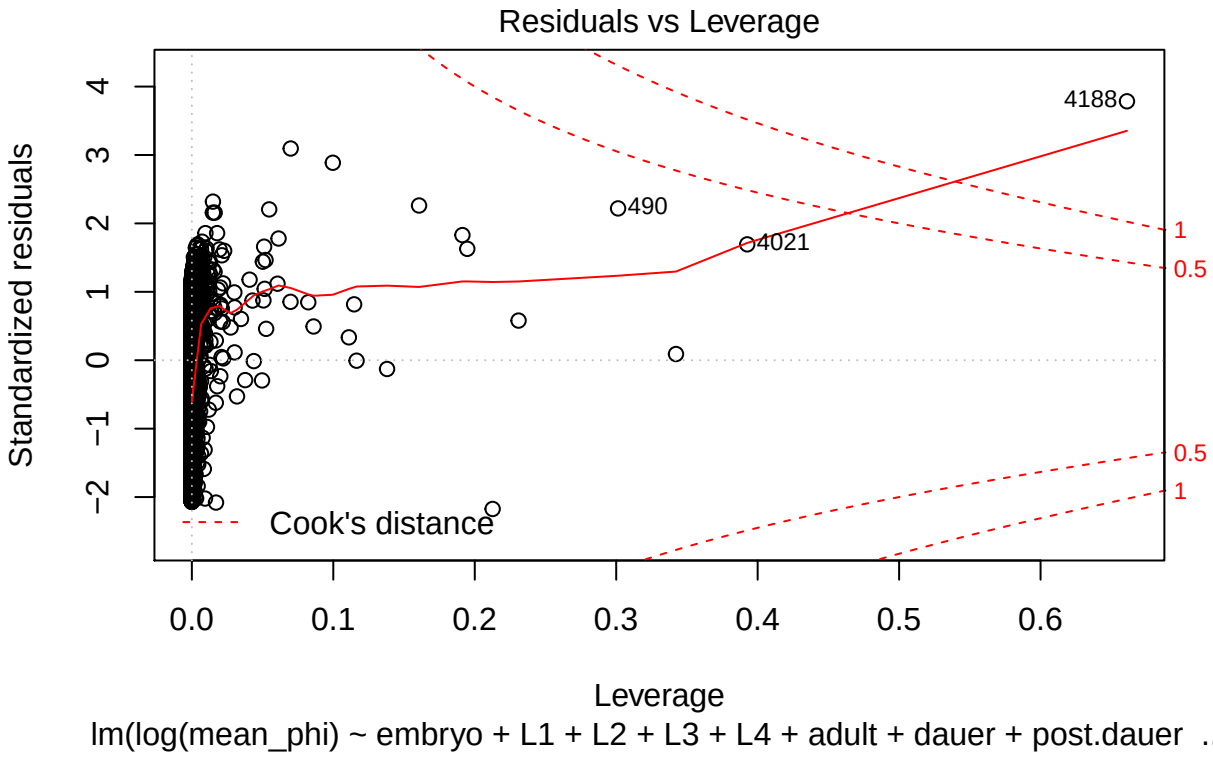
Log Transformation of `mean_phi`

```
logPhiFit  <- lm(log(mean_phi) ~ embryo+L1+L2+L3+L4+adult+dauer+post.dauer-1, filteredGeneData)

plot(logPhiFit)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(log(mean_phi) ~ embryo + L1 + L2 + L3 + L4 + adult + dauer + post.dauer ...
4188
11123
4822


Scale−Location
√|Standardized residuals|
Fitted values
lm(log(mean_phi) ~ embryo + L1 + L2 + L3 + L4 + adult + dauer + post.dauer ...
4188
11123
4822

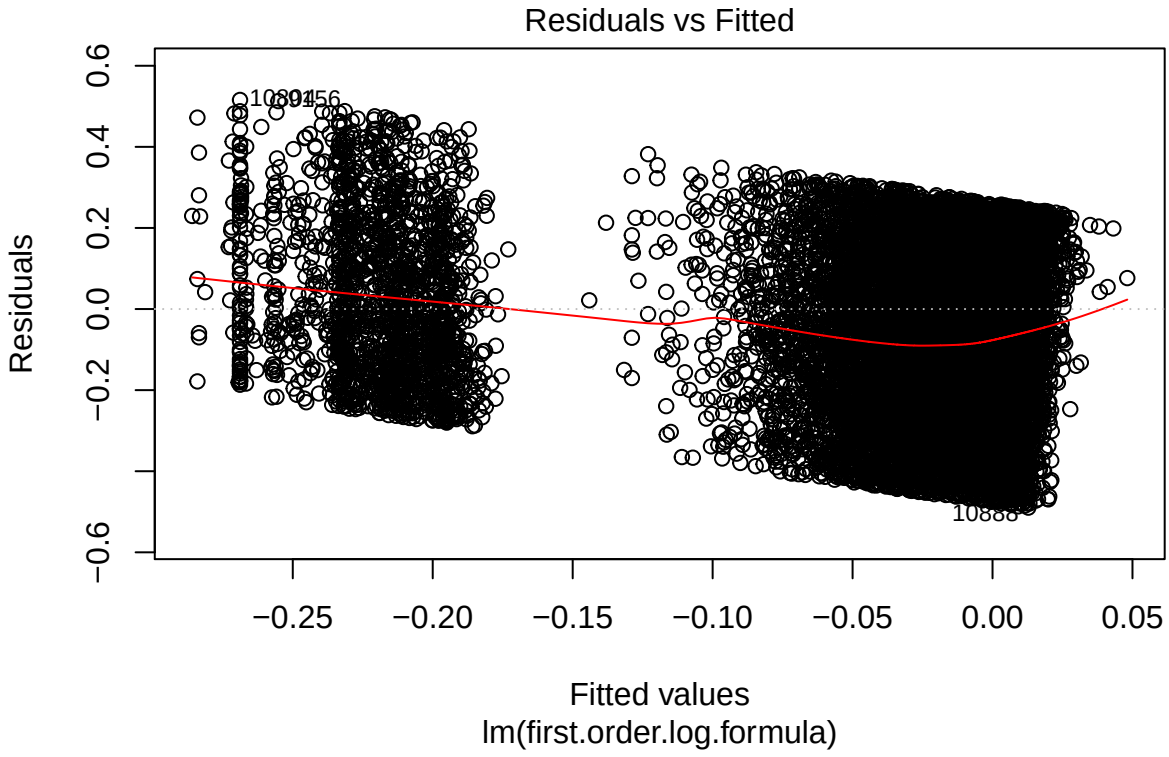


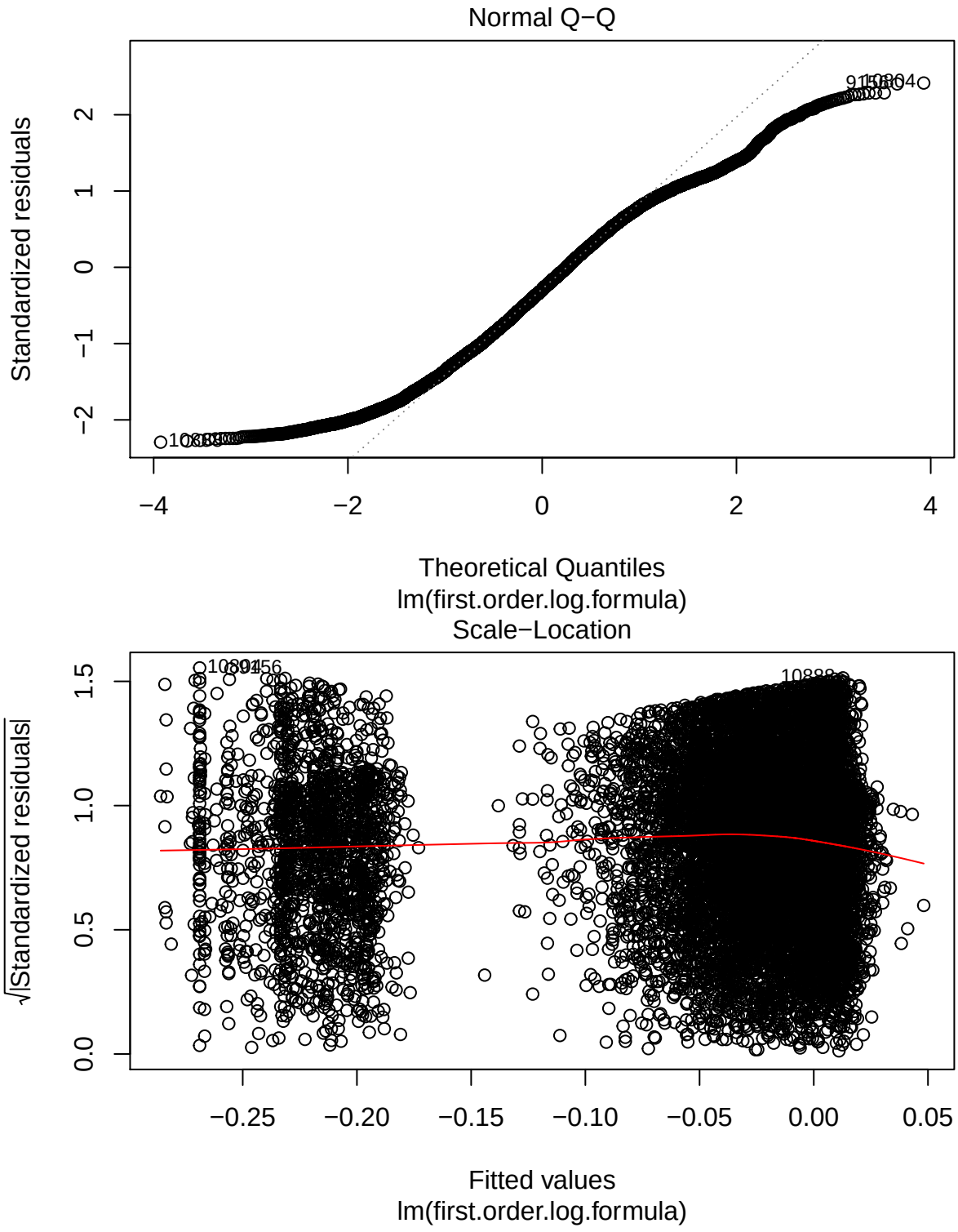

```
summary(logPhiFit)
```

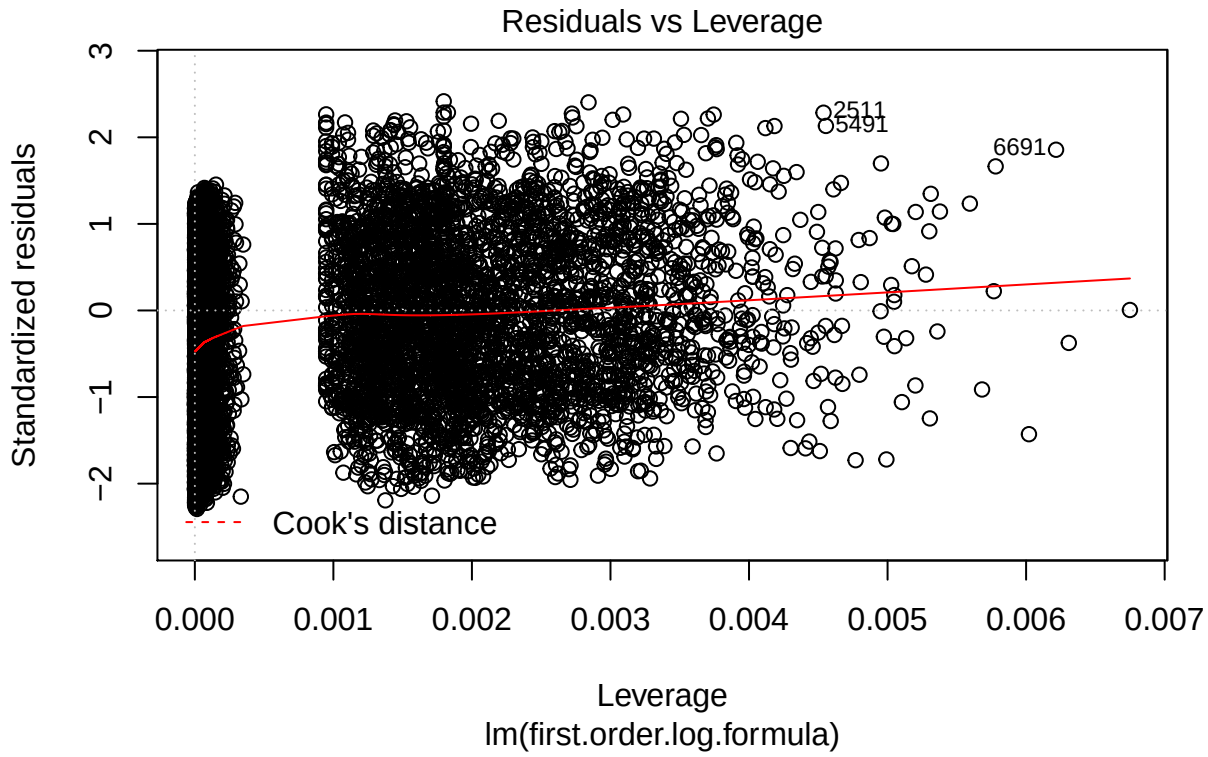
```
##
## Call:
## lm(formula = log(mean_phi) ~ embryo + L1 + L2 + L3 + L4 + adult +
##     dauer + post.dauer - 1, data = filteredGeneData)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.47745 -0.27352 -0.10154  0.07075  0.68841
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## embryo     -0.0081484  0.0011918  -6.837 8.47e-12 ***
## L1         -0.0003956  0.0006366  -0.621   0.5343
## L2          0.0002392  0.0021526   0.111   0.9115
## L3         -0.0038337  0.0017380  -2.206   0.0274 *
## L4          0.0002796  0.0010866   0.257   0.7969
## adult       0.0011956  0.0009290   1.287   0.1981
## dauer       0.0008823  0.0013504   0.653   0.5135
## post.dauer -0.0016375  0.0018114  -0.904   0.3660
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2306 on 11678 degrees of freedom
## Multiple R-squared:  0.01122,    Adjusted R-squared:  0.01055
## F-statistic: 16.57 on 8 and 11678 DF,  p-value: < 2.2e-16
```

Log All Data

```
logStages  <- paste0("log(", stages, "+ 1E-10)")
first.order.log.formula<- as.formula(paste(c("log(mean_phi) ~ -1", logStages), collapse = " + "))

firstOrderLogFit  <- lm(first.order.log.formula, filteredGeneData)


## Residuals are linear with the response variable!!
plot(firstOrderLogFit)
```




Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(first.order.log.formula)
Scale−Location
√|Standardized residuals|
Fitted values
lm(first.order.log.formula)

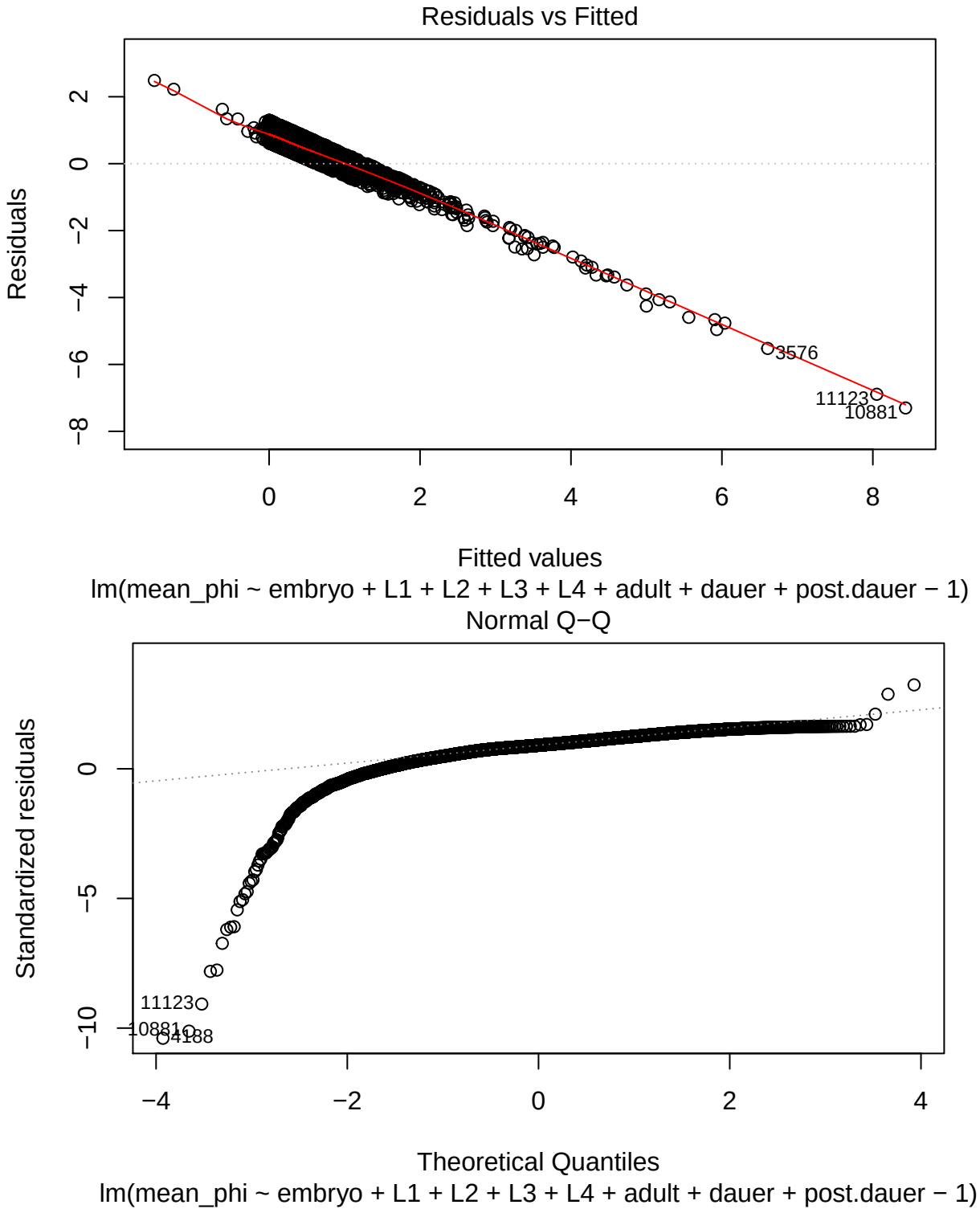


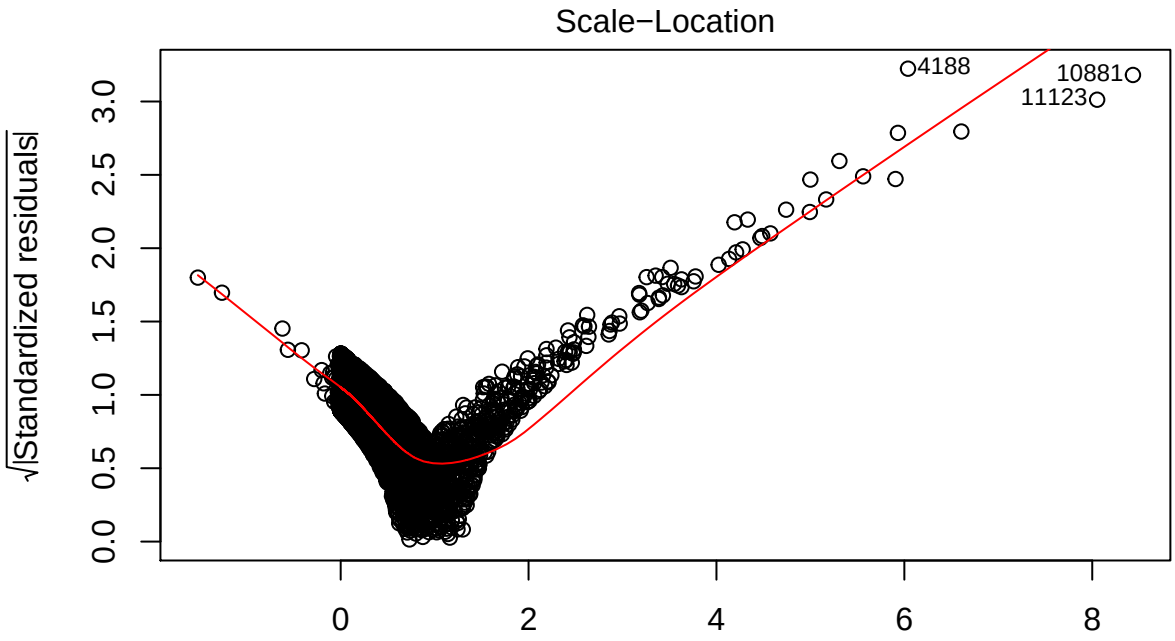

```
summary(firstOrderPhiFit)
```

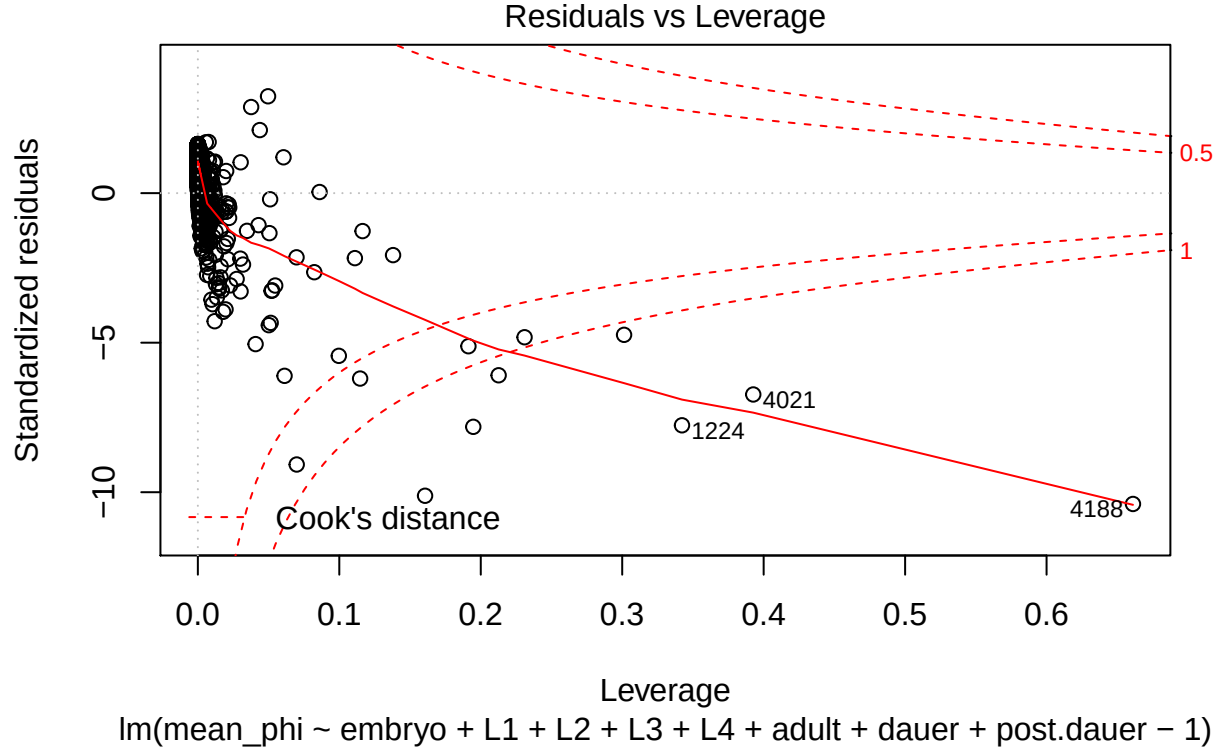
```
##
## Call:
## lm(formula = first.order.formula, data = filteredGeneData)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3010  0.5326  0.7158  0.8969  2.4863
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## embryo      0.068461   0.004070  16.822  < 2e-16 ***
## L1          0.010119   0.002174   4.655 3.28e-06 ***
## L2          0.084720   0.007351  11.525  < 2e-16 ***
## L3          0.015925   0.005935   2.683   0.0073 **
## L4          0.009090   0.003711   2.450   0.0143 *
## adult       0.039206   0.003172  12.358  < 2e-16 ***
## dauer      -0.041717   0.004611  -9.047  < 2e-16 ***
## post.dauer  0.069863   0.006186  11.294  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7876 on 11678 degrees of freedom
## Multiple R-squared:  0.2872, Adjusted R-squared:  0.2867
## F-statistic: 588.2 on 8 and 11678 DF,  p-value: < 2.2e-16
```

```
quadPhiFit  <- lm(mean_phi ~ embryo+L1+L2+L3+L4+adult+dauer+post.dauer-1, filteredGeneData)
```

```
## Residuals are linear with the response variable!!
plot(quadPhiFit)
```








```
summary(quadPhiFit)
```

```
##
## Call:
## lm(formula = mean_phi ~ embryo + L1 + L2 + L3 + L4 + adult +
##     dauer + post.dauer - 1, data = filteredGeneData)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.3010  0.5326  0.7158  0.8969  2.4863
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## embryo      0.068461   0.004070  16.822  < 2e-16 ***
## L1          0.010119   0.002174   4.655 3.28e-06 ***
## L2          0.084720   0.007351  11.525  < 2e-16 ***
## L3          0.015925   0.005935   2.683   0.0073 **
## L4          0.009090   0.003711   2.450   0.0143 *
## adult       0.039206   0.003172  12.358  < 2e-16 ***
## dauer      -0.041717   0.004611  -9.047  < 2e-16 ***
## post.dauer  0.069863   0.006186  11.294  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7876 on 11678 degrees of freedom
## Multiple R-squared:  0.2872, Adjusted R-squared:  0.2867
## F-statistic: 588.2 on 8 and 11678 DF,  p-value: < 2.2e-16
```

```
## Need to do a contrained fit where
## sum(coefficients) =1 and coefficients > 0
```