sara-experimental

Sara Shao

10/7/2021

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v tibble  3.0.6     v purrr   0.3.4
## v tidyr   1.1.2     v dplyr   1.0.4
## v readr   1.4.0     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x readr::guess_encoding()  masks rvest::guess_encoding()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x purrr::pluck()           masks rvest::pluck()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

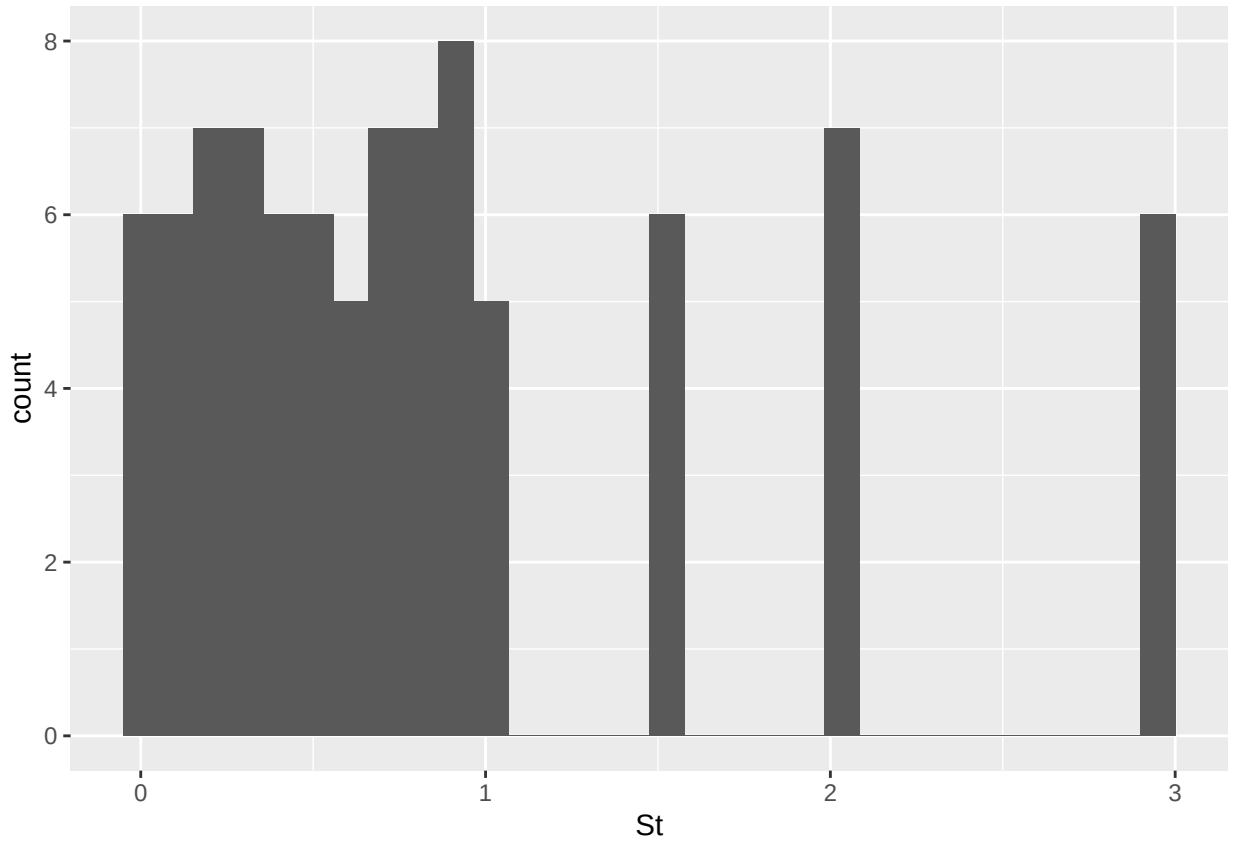
```
train <- read.csv('data-train.csv')
```

```
head(train)
```

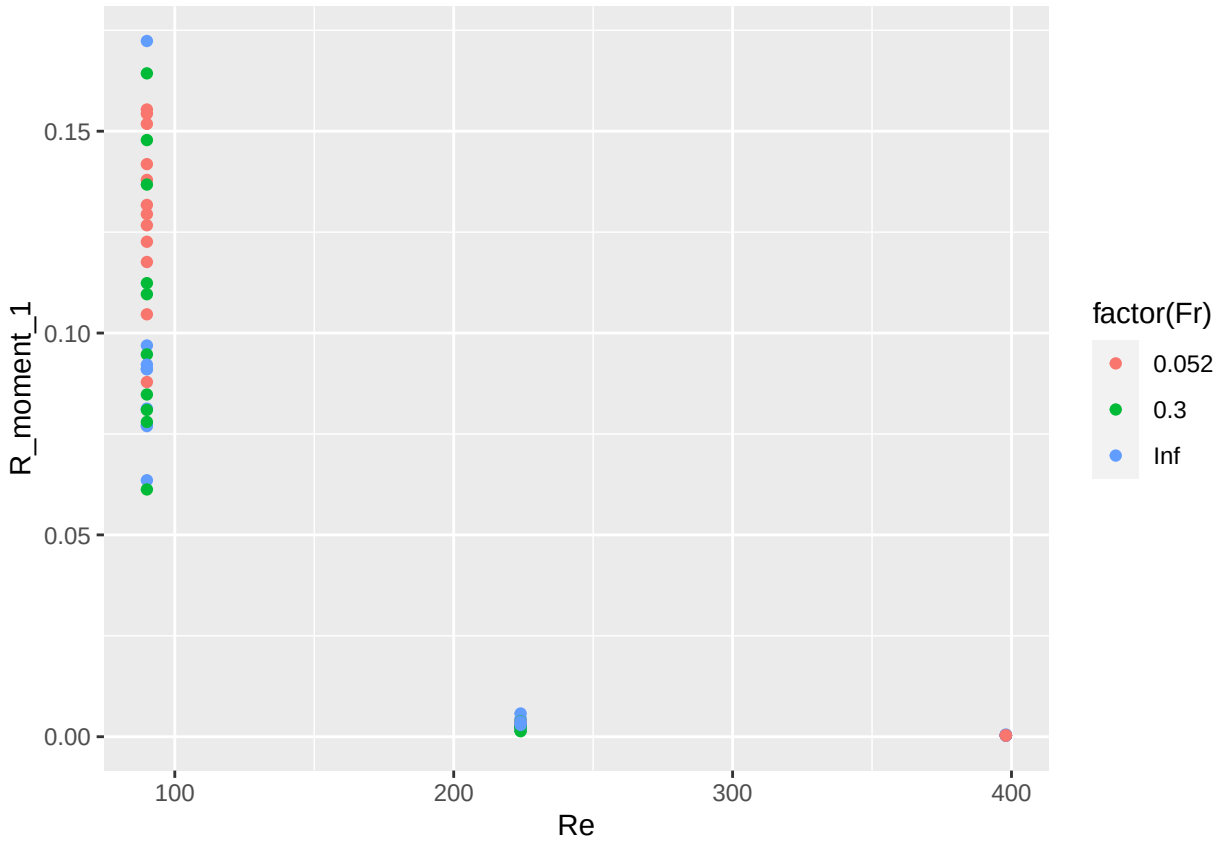
```
##     St  Re    Fr R_moment_1 R_moment_2 R_moment_3 R_moment_4
## 1 0.10 224 0.052 0.00215700  0.1303500   14.37400  1586.5000
## 2 3.00 224 0.052 0.00379030  0.4704200   69.94000 10404.0000
## 3 0.70 224   Inf 0.00290540  0.0434990    0.82200    15.5510
## 4 0.05  90   Inf 0.06352800  0.0906530    0.46746     3.2696
## 5 0.70 398   Inf 0.00036945  0.0062242    0.12649     2.5714
## 6 2.00  90 0.300 0.14780000  2.0068000   36.24900   671.6700
```

```
ggplot(data = train, mapping = aes(x = St)) +
  geom_histogram()
```

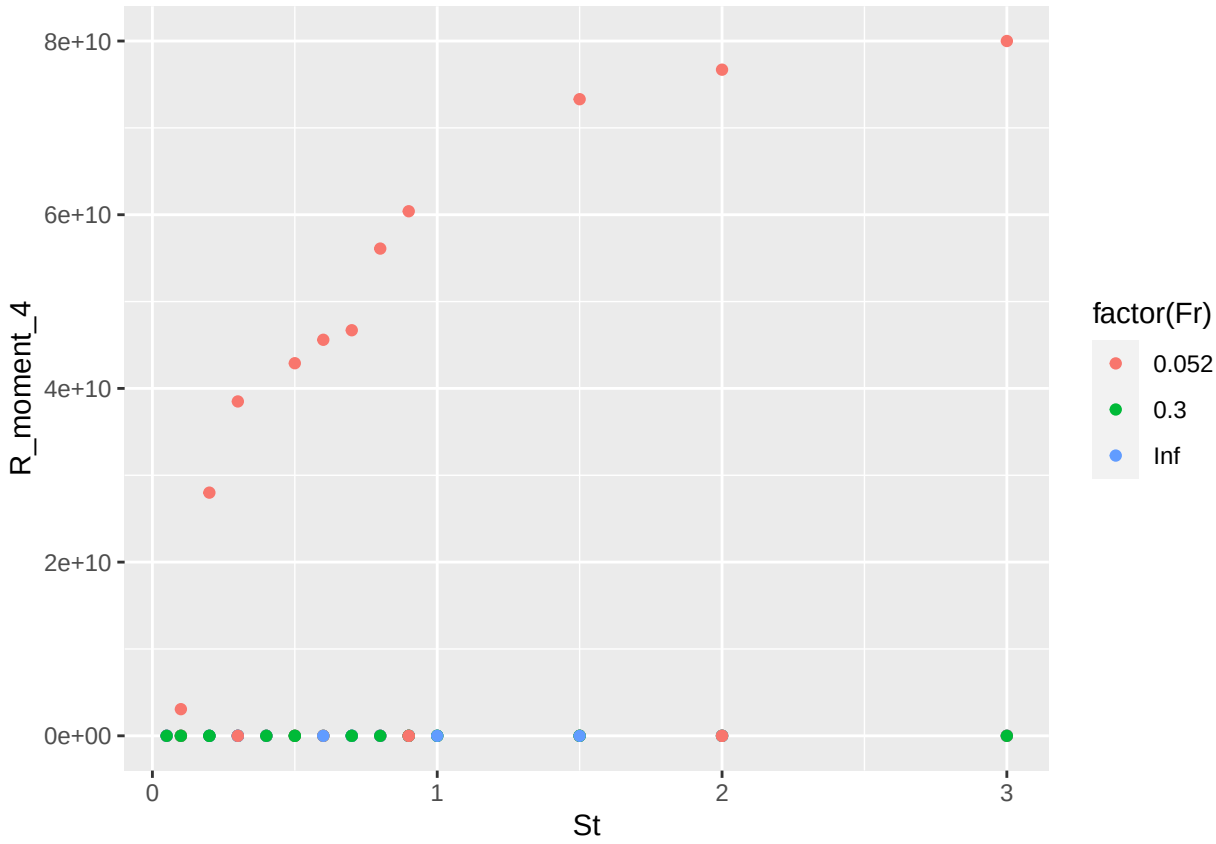
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(data = train, mapping = aes(x = Re, y = R_moment_1, color = factor(Fr))) +
  geom_point()
```

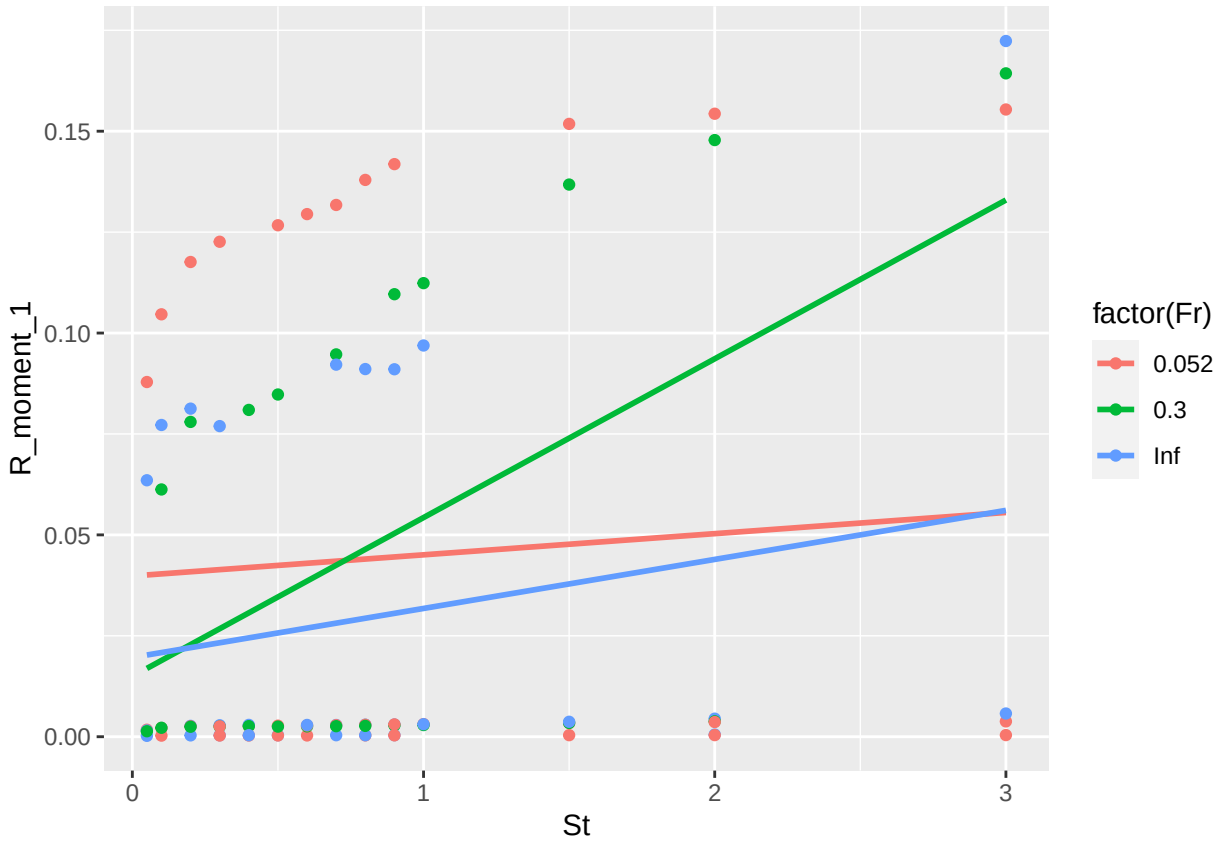



```
ggplot(data = train, mapping = aes(x = St, y = R_moment_4, color = factor(Fr))) +
  geom_point()
```



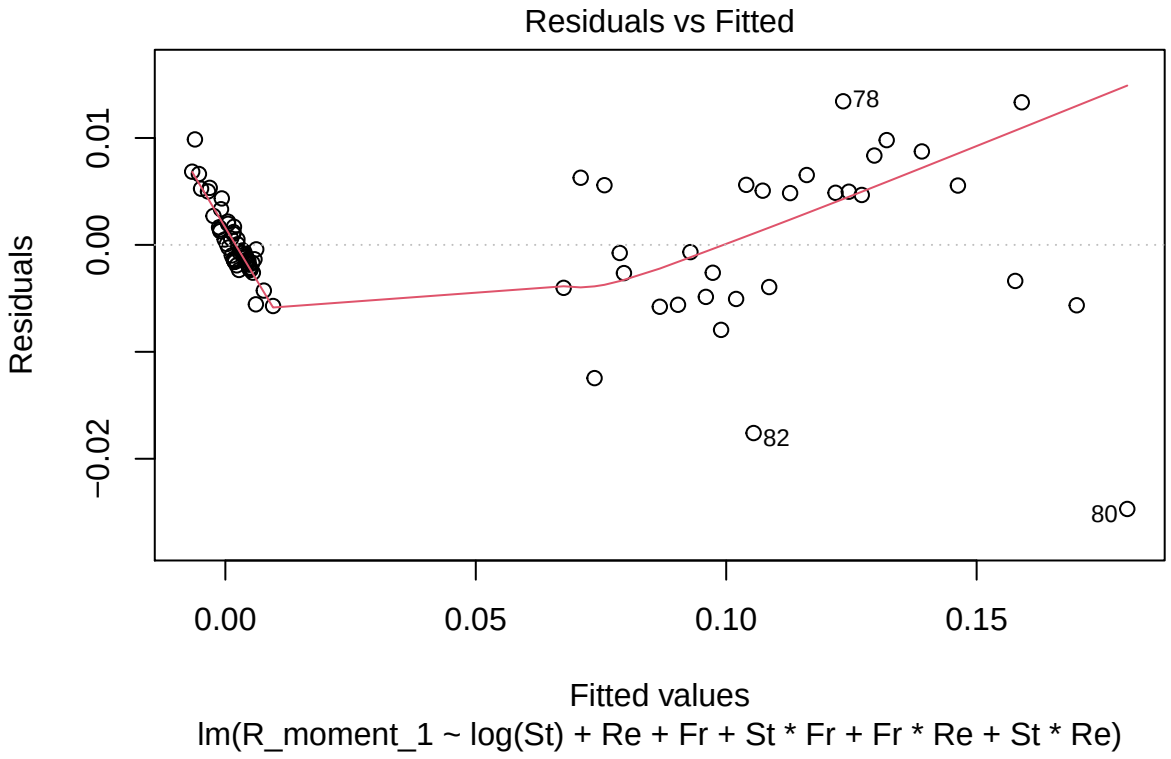

```
ggplot(data = train, mapping = aes(x = St, y = R_moment_1, color = factor(Fr))) +
  geom_point() +
  geom_smooth(method = lm, se = F)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

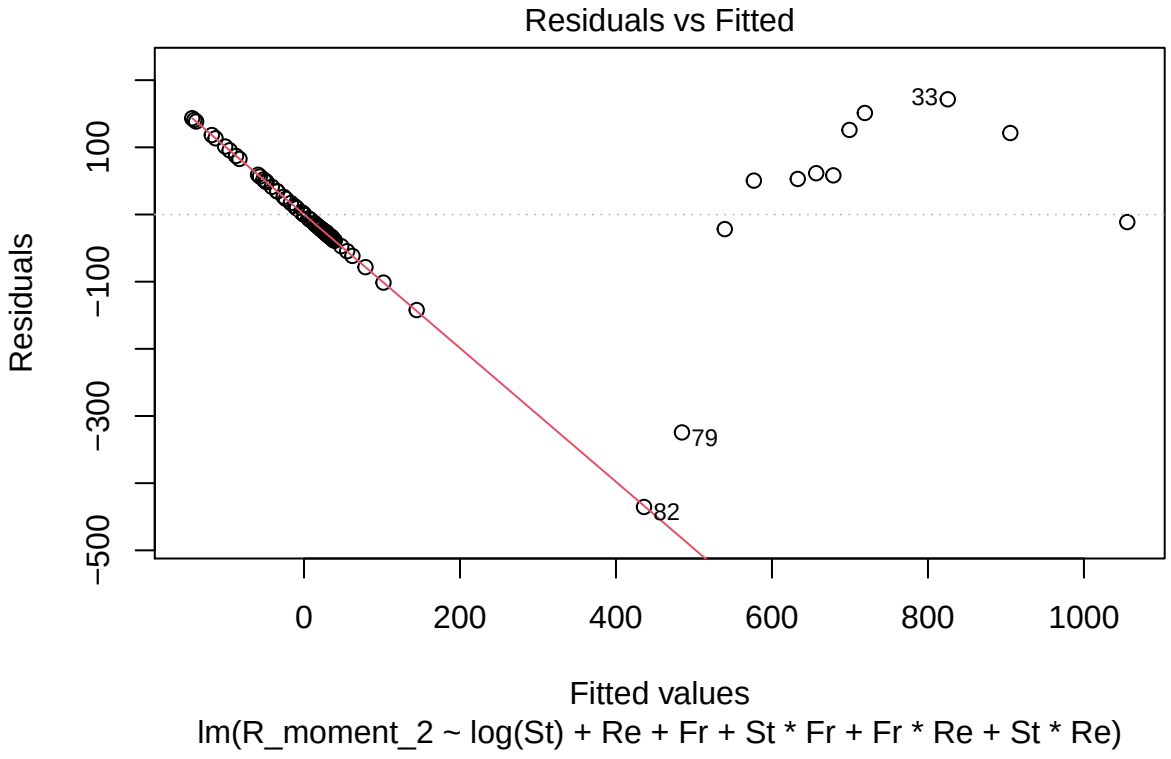



```
train_data <- train %>%
  mutate(Fr = as.factor(Fr)) %>%
  mutate(Re = as.factor(Re))
```

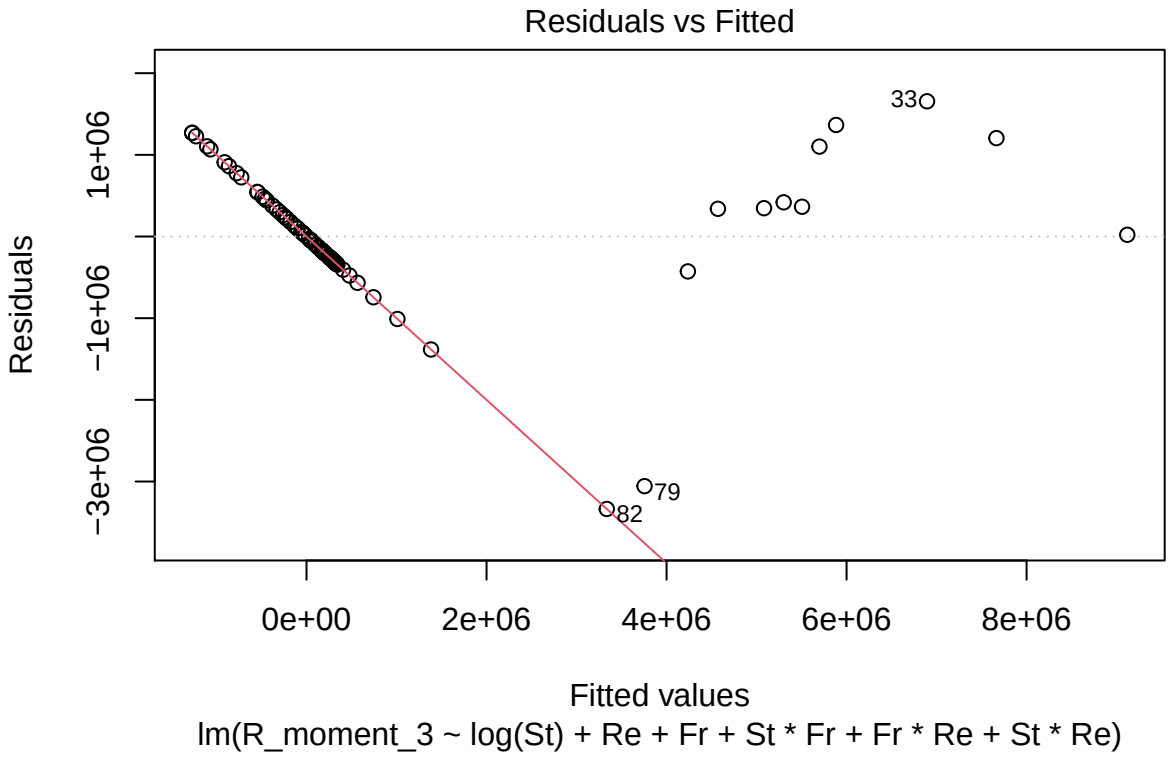
```
lm_R1 <- lm(R_moment_1 ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
lm_R2 <- lm(R_moment_2 ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
lm_R3 <- lm(R_moment_3 ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
lm_R4 <- lm(R_moment_4 ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
plot(lm_R1, 1)
```



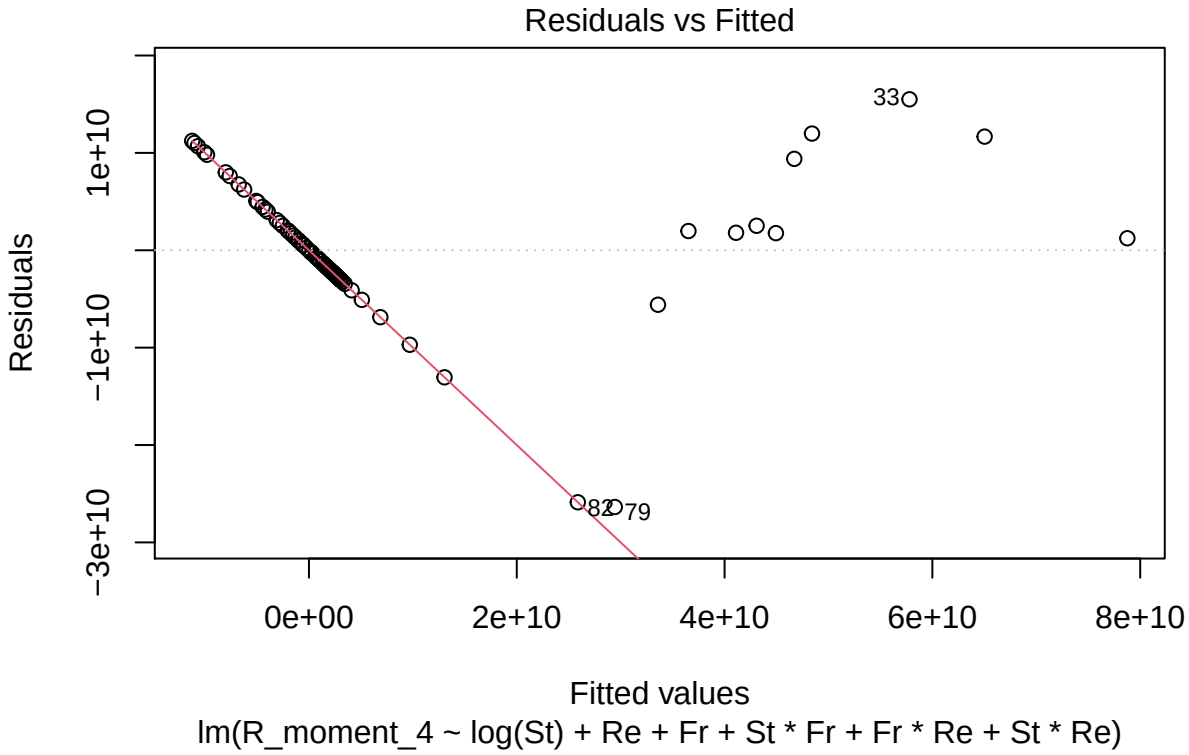

```
plot(lm_R2, 1)
```




```
plot(lm_R3, 1)
```




```
plot(lm_R4, 1)
```




```
lm1 <- lm(log(R_moment_1) ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
summary(lm1)
```

```
##
```

```
## Call:
## lm(formula = log(R_moment_1) ~ log(St) + Re + Fr + St * Fr +
##     Fr * Re + St * Re, data = train_data)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.211809 -0.042926 -0.006391  0.038831  0.171243
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error  t value Pr(>|t|)
## (Intercept) -1.95494    0.03716  -52.606  < 2e-16 ***
## log(St)      0.14567    0.01456   10.003 1.89e-15 ***
## Re224       -3.84790    0.03653 -105.330  < 2e-16 ***
## Re398       -5.98498    0.04408 -135.783  < 2e-16 ***
## Fr0.3       -0.41718    0.04329   -9.637 9.23e-15 ***
## FrInf       -0.43145    0.03943  -10.942  < 2e-16 ***
## St          -0.00182    0.02631   -0.069    0.945
## Fr0.3:St     0.14612    0.02916    5.011 3.51e-06 ***
## FrInf:St     0.13488    0.02442    5.522 4.60e-07 ***
## Re224:Fr0.3  0.25492    0.04471    5.701 2.22e-07 ***
## Re398:Fr0.3       NA         NA       NA       NA
## Re224:FrInf  0.37513    0.04533    8.275 3.58e-12 ***
## Re398:FrInf  0.48589    0.04954    9.808 4.39e-15 ***
## Re224:St     0.02473    0.02376    1.041    0.301
## Re398:St    -0.01258    0.03065   -0.411    0.683
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.07645 on 75 degrees of freedom
## Multiple R-squared:  0.999,  Adjusted R-squared:  0.9988
## F-statistic:  5797 on 13 and 75 DF,  p-value: < 2.2e-16
```

```
lm2 <- lm(log(R_moment_2) ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
summary(lm2)
```

```
##
## Call:
## lm(formula = log(R_moment_2) ~ log(St) + Re + Fr + St * Fr +
##     Fr * Re + St * Re, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.6442 -0.2697 -0.0561  0.3429  1.8016
##
## Coefficients: (1 not defined because of singularities)
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.55038    0.41301  18.282  < 2e-16 ***
## log(St)       1.50086    0.16184   9.274 4.50e-14 ***
## Re224        -7.37089    0.40601 -18.155  < 2e-16 ***
## Re398       -10.93201    0.48987 -22.316  < 2e-16 ***
## Fr0.3        -6.65031    0.48112 -13.823  < 2e-16 ***
## FrInf        -6.49869    0.43823 -14.830  < 2e-16 ***
## St           -0.72933    0.29241  -2.494   0.0148 *
## Fr0.3:St      0.04758    0.32411   0.147   0.8837
## FrInf:St     -0.01113    0.27145  -0.041   0.9674
```

```
## Re224:Fr0.3   4.50066    0.49695   9.057 1.16e-13 ***
## Re398:Fr0.3        NA         NA      NA       NA
## Re224:FrInf   4.51871    0.50382   8.969 1.71e-13 ***
## Re398:FrInf   6.79875    0.55058  12.348  < 2e-16 ***
## Re224:St     -0.17530    0.26403  -0.664   0.5088
## Re398:St     -0.66556    0.34062  -1.954   0.0544 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8497 on 75 degrees of freedom
## Multiple R-squared:  0.9553, Adjusted R-squared:  0.9476
## F-statistic: 123.4 on 13 and 75 DF,  p-value: < 2.2e-16
```

```
lm3 <- lm(log(R_moment_3) ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
summary(lm3)
```

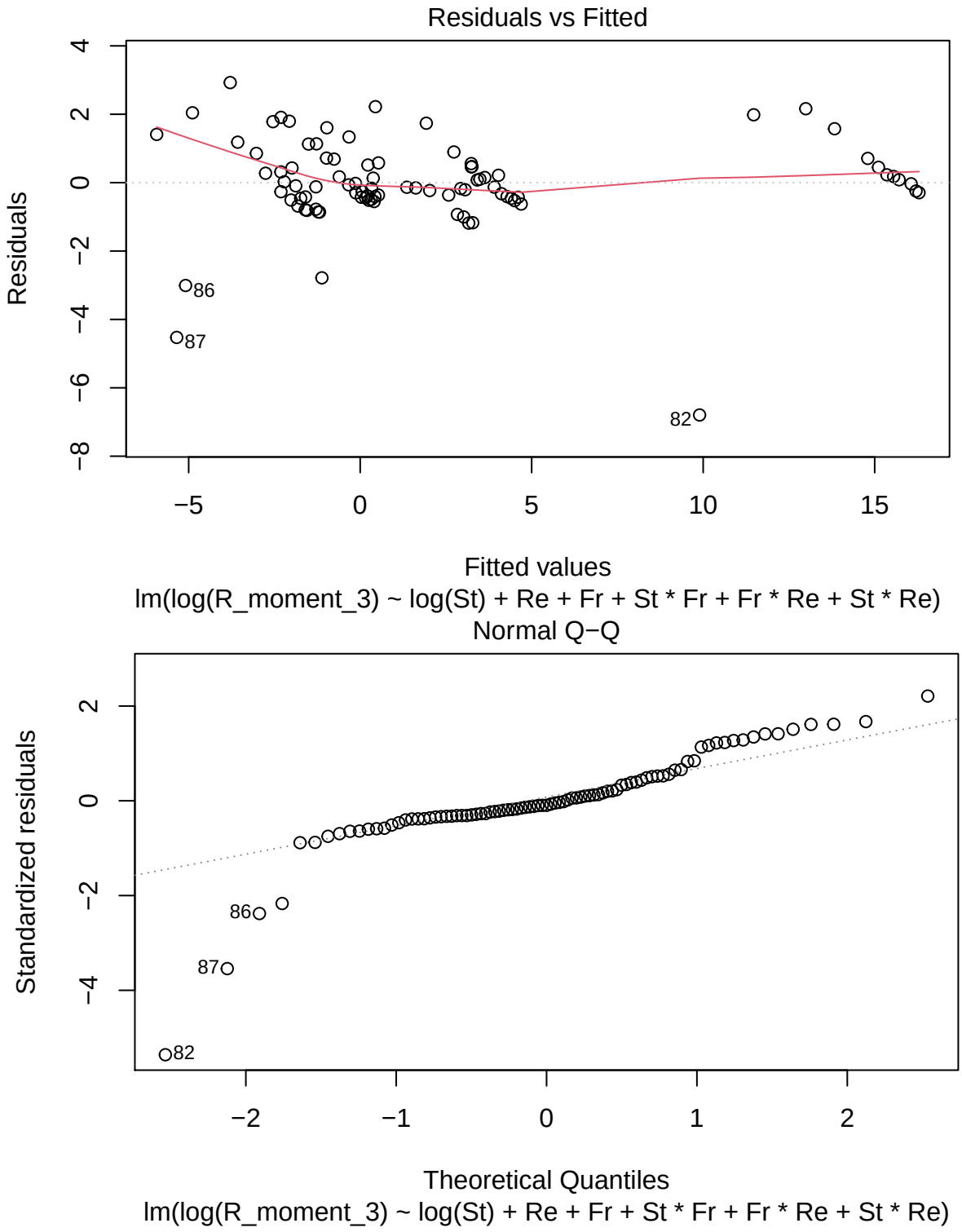
```
##
## Call:
## lm(formula = log(R_moment_3) ~ log(St) + Re + Fr + St * Fr +
##     Fr * Re + St * Re, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.7949 -0.4431 -0.1224  0.5575  2.9257
##
## Coefficients: (1 not defined because of singularities)
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  17.03012    0.69563  24.481  < 2e-16 ***
## log(St)       2.36189    0.27259   8.665 6.47e-13 ***
## Re224       -10.99422    0.68384 -16.077  < 2e-16 ***
## Re398       -16.25017    0.82509 -19.695  < 2e-16 ***
## Fr0.3       -12.76155    0.81035 -15.748  < 2e-16 ***
## FrInf       -12.43166    0.73811 -16.842  < 2e-16 ***
## St           -1.18717    0.49251  -2.410   0.0184 *
## Fr0.3:St     -0.02117    0.54590  -0.039   0.9692
## FrInf:St     -0.13003    0.45721  -0.284   0.7769
## Re224:Fr0.3   8.53024    0.83702  10.191 8.37e-16 ***
## Re398:Fr0.3        NA         NA      NA       NA
## Re224:FrInf   8.46417    0.84860   9.974 2.14e-15 ***
## Re398:FrInf  12.91606    0.92735  13.928  < 2e-16 ***
## Re224:St     -0.34559    0.44472  -0.777   0.4395
## Re398:St     -1.12784    0.57371  -1.966   0.0530 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.431 on 75 degrees of freedom
## Multiple R-squared:  0.9459, Adjusted R-squared:  0.9365
## F-statistic: 100.8 on 13 and 75 DF,  p-value: < 2.2e-16
```

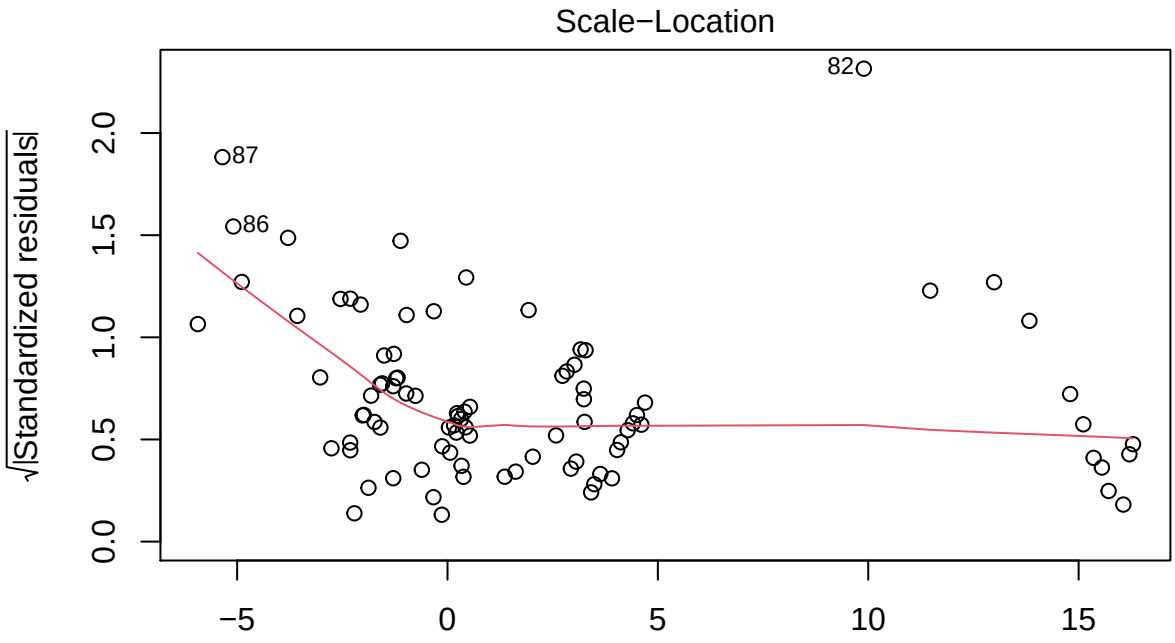
```
lm4 <- lm(log(R_moment_4) ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
summary(lm4)
```

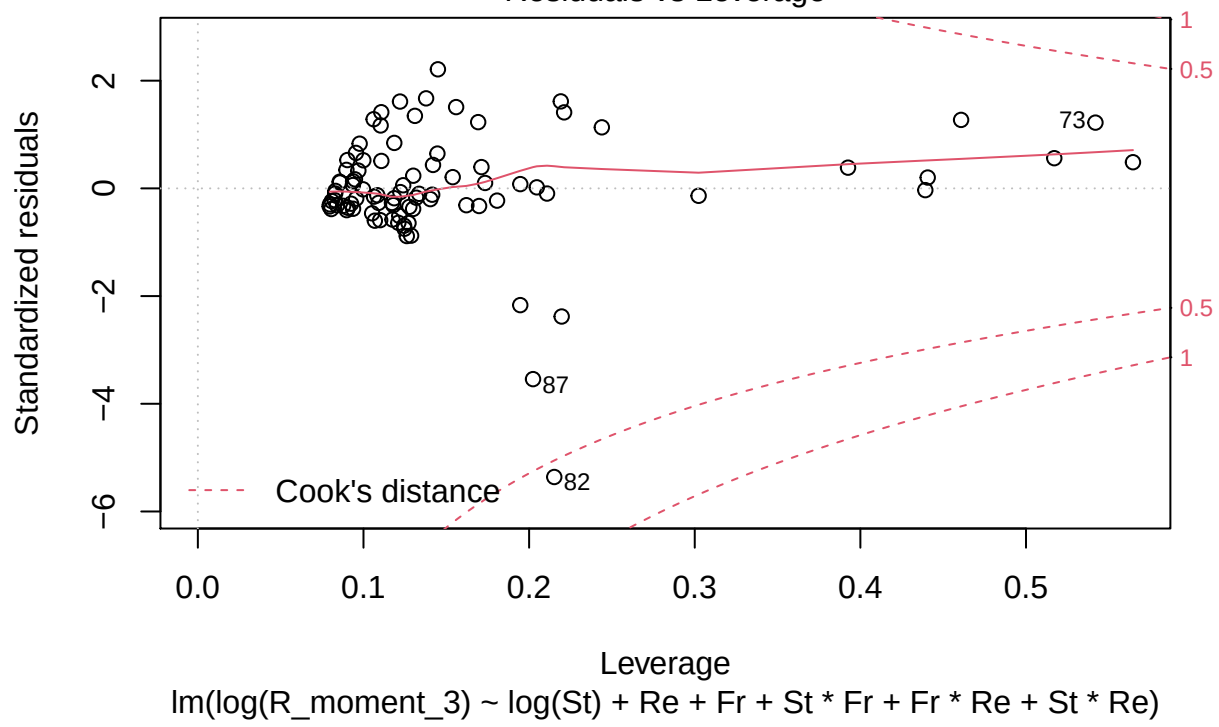
```
##
## Call:
## lm(formula = log(R_moment_4) ~ log(St) + Re + Fr + St * Fr +
```

```
##     Fr * Re + St * Re, data = train_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.6675 -0.6183 -0.1392  0.7410  3.8875
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  26.4189     0.9472  27.891  < 2e-16 ***
## log(St)       3.1095     0.3712   8.377 2.28e-12 ***
## Re224       -14.6485     0.9312 -15.731  < 2e-16 ***
## Re398       -21.6490     1.1235 -19.269  < 2e-16 ***
## Fr0.3       -18.7784     1.1034 -17.018  < 2e-16 ***
## FrInf       -18.2947     1.0051 -18.202  < 2e-16 ***
## St           -1.5753     0.6706  -2.349   0.0215 *
## Fr0.3:St     -0.1004     0.7433  -0.135   0.8930
## FrInf:St     -0.2494     0.6226  -0.401   0.6899
## Re224:Fr0.3  12.4829     1.1398  10.952  < 2e-16 ***
## Re398:Fr0.3       NA         NA      NA       NA
## Re224:FrInf  12.3551     1.1555  10.692  < 2e-16 ***
## Re398:FrInf  18.9591     1.2628  15.014  < 2e-16 ***
## Re224:St     -0.4931     0.6056  -0.814   0.4180
## Re398:St     -1.5312     0.7812  -1.960   0.0537 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.949 on 75 degrees of freedom
## Multiple R-squared:  0.9457, Adjusted R-squared:  0.9363
## F-statistic: 100.6 on 13 and 75 DF,  p-value: < 2.2e-16
```

```
plot(lm3)
```


Residuals vs Fitted
Residuals
Fitted values
lm(log(R_moment_3) ~ log(St) + Re + Fr + St * Fr + Fr * Re + St * Re)
Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(log(R_moment_3) ~ log(St) + Re + Fr + St * Fr + Fr * Re + St * Re)
86
87
82






```
set.seed(123)
errors <- c()

for(i in c(1:nrow(train_data))){
  subset <- train_data[-c(i),]
  model <- lm(log(R_moment_1) ~ log(St) + Re + Fr + St*Fr + Re*Fr + St*Re, data = subset)
```

```
  prediction <- exp(predict(model, train_data[c(i),]))
  error <- (train_data[c(i),]$R_moment_1 - prediction)^2
  errors[i] <- error
}

mean(errors)
```

```
## [1] 2.720867e-05
```

```
set.seed(123)
errors <- c()

for(i in c(1:nrow(train_data))){
  subset <- train_data[-c(i),]
  model <- lm(log(R_moment_2) ~ log(St) + Re + Fr + St*Fr + Re*Fr + St*Re, data = subset)
  prediction <- exp(predict(model, train_data[c(i),]))
  error <- (train_data[c(i),]$R_moment_2 - prediction)^2
  errors[i] <- error
}

mean(errors)
```

```
## [1] 6312.875
```

```
set.seed(123)
errors <- c()

for(i in c(1:nrow(train_data))){
  subset <- train_data[-c(i),]
  model <- lm(log(R_moment_3) ~ log(St) + Re + Fr + St*Fr + Re*Fr + St*Re, data = subset)
  prediction <- exp(predict(model, train_data[c(i),]))
  error <- (train_data[c(i),]$R_moment_3 - prediction)^2
  errors[i] <- error
}

mean(errors)
```

```
## [1] 777311609607
```

```
set.seed(123)
errors <- c()

for(i in c(1:nrow(train_data))){
  subset <- train_data[-c(i),]
  model <- lm(log(R_moment_4) ~ log(St) + Re + Fr + St*Fr + Re*Fr + St*Re, data = subset)
  prediction <- exp(predict(model, train_data[c(i),]))
  error <- (train_data[c(i),]$R_moment_4 - prediction)^2
  errors[i] <- error
}

mean(errors)
```

```
## [1] 7.403514e+19
```