Case Study: Modeling Liquid Mechanics

Conner Byrd, Eric Han, Ki Hyun, Sara Shao, Alex Shen, Mona Su, Dani Trejo, Steven Yuan

10/12/2021

Introduction

Our key research objectives include understanding and predicting how turbulence affects the dynamics of water droplets and ice crystals (how they collide and mix) in clouds. With our machine learning model, we are trying to infer the volume distribution of clusters within clouds.

To do this, we began by doing some basic Exploratory Data Analysis on the three predictor variables: Reynolds number (`Re`), gravitational acceleration (`Fr`), and particle characteristic (`St`). (Our graphs are included in Appendix: Section 1)

```
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
```

Methodology

Initial Approach

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

We decided at the beginning to treat Fr and Re as ordered factors. It seemed to make sense to treat Fr as a categorical variable because we are given only three unique values. In terms of prediction and inference, we believe our models can still be generalized to Fr and Re values similar to the ones we are working with.

```
first_lm_R1 <- lm(R_moment_1 ~ St + Re + Fr, data = train_data)
first_lm_R2 <- lm(R_moment_2 ~ St + Re + Fr, data = train_data)
first_lm_R3 <- lm(R_moment_3 ~ St + Re + Fr, data = train_data)
first_lm_R4 <- lm(R_moment_4 ~ St + Re + Fr, data = train_data)
summary(first_lm_R1)$r.squared
```

```
## [1] 0.9293093
```

```
summary(first_lm_R4)$r.squared
```

```
## [1] 0.4252642
```

We decided to initially fit the most basic linear model with all three predictors to see what it would look like. The model for the first moment had a fairly high R^2 value (0.929), but the model for the fourth moment had a much lower R^2 value (0.425). Additionally, in our diagnostic plots (see Appendix: Section 2), we saw a pattern in the residuals vs fitted values plots where the models would consistently under predict in some areas and over predict in others, indicating non-linearity. In addition, looking at the Normal Q-Q plots, the normality assumption also seemed to be violated for higher moments. This is consistent with the fact that our histogram of St in our EDA was not normally distributed.

This information lead us to try using a GAM to model the relationship between the predictors and the 4 moments due to the increased flexbility GAMs provide. However, we knew that using GAMs made interpretability an issue, because interpreting a complex smooth function of a continuous predictor is very hard.

Varaible transformation

As a result, we decided to use variable transformations and include interaction effects to make linear models with suitable model diagnostics for all 4 moments for the purpose of inference. To decide how to apply variable transformations, we observed the predictor and response variables, took a deeper look at the purpose of the study, and found three main problems.

1. Just converting Fr, and Re into categorical values may result in losing information as the exact numerical value could have high relationship with the moments. Furthermore, this model should be used later to predict behaviors of particles in environments with higher Re values.
2. The scale of Fr, St, and Re differs from each other. Fr ranges $[0, \infty)$, while St is $[0, 3]$. Furthermore, in real world, Re could be in the scale of 10^7.
3. The response variables (moments) are heavily skewed. For example, for the 4th moment, some response variables are at the scale of 10^{-10}, while some are at the scale of 10^{10}

As a result, we considered transformations on the potential predictor variables. For Froude number (Fr), a monotonic transformation of $1 - e^{-Fr}$ was made to change the range from $[0, \infty)$ to $[0, 1]$. For Stokes number (St), as shown in the EDA, log transformation was undertaken to address the skewness. For Reynolds number (Re), not only is the actual value in practice much larger than the given training data, but also within the training data itself, Reynolds number was much greater in scale. To address the difference in scale, log transformation was also made on Reynolds number.

Transformation on the response variable also needed to take place before fitting the models. For the four moments, we have considered box-cox transformation to normalize the variables and adjust for the difference in scale. As the EDA shows, the response variables have a skewed distribution. Thus, first, normalization was needed for the four moments. Moreover, the scale of the response variable becomes extremely larger as the order of the moments increase. (4th moment would have approximately to the power of 4 scale of the 1st moment) Hence, second, adjustments for scale were needed for the four moments. Box-cox transformation performs normalization and adjusts the scale of variables: apt for our response variable transformation.

However, the λ for box-cox transformation needed to be specified for each of the four moments. We have calculated and speficied the four different lambdas as the maximum log-likelihood estimate as below.

Cross-Validation

For each moments, we considered various combinations of predictor variables and interactions of the transformed variables. In order to compare the models and choose the most optimal ones, we used 10-fold cross validation to estimate the test error. A major reason for choosing cross validation was that cross validation can provide a direct estimate for the test estimate, even when we do not have a clear picture of the noises of the variables. Furthermore, it is less likely to result in overfitting compared to LOOCV, and computationally more feasible.

To perform a valid 10-fold cross validation, we first shuffled the training dataset to ensure that similar data do not tend to be included in the same fold. Then, we divided the training dataset into 10 folds and used the same folds to perform cross validation and obtain estimated test error for each model. Below is how the folds were created.

At each fold, the validation set (8~9 data points for each fold) was reserved, and the other points were used to train a model. Then, the model was used to test on the validation set. RMSE was then calculated at each fold and the mean RSME was then calculated to compare the estimated test error with other model candidates. For each moment, the model with the lowest estimated test error was selected.

Results

Final Linear Model

```
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) -5.349488   0.027649 -193.480  < 2e-16 ***
## log(St)      0.145668   0.014562   10.003 1.89e-15 ***
## Re.L        -4.028476   0.027949 -144.139  < 2e-16 ***
## Re.Q         0.644476   0.022457   28.698  < 2e-16 ***
## Fr.L        -0.102135   0.019793   -5.160 1.95e-06 ***
## Fr.Q         0.109493   0.026874    4.074 0.000113 ***
## St           0.095896   0.021136    4.537 2.13e-05 ***
## Fr.L:St      0.095376   0.017271    5.522 4.60e-07 ***
## Fr.Q:St     -0.064244   0.022042   -2.915 0.004692 **
## Re.L:Fr.L    0.242947   0.024770    9.808 4.39e-15 ***
## Re.Q:Fr.L   -0.076314   0.022588   -3.379 0.001159 **
## Re.L:Fr.Q   -0.077781   0.046306   -1.680 0.097169 .
## Re.Q:Fr.Q         NA         NA       NA       NA
## Re.L:St     -0.008897   0.021672   -0.411 0.682581
## Re.Q:St     -0.025325   0.018376   -1.378 0.172252
## ---
```

```
## 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))
```

```
##
## Call:
## lm(formula = log(R_moment_2) ~ log(St) + Re + Fr + St * Fr +
##     Fr * Re + St * Re, data = train_data)
##
## Coefficients:
## (Intercept)       log(St)          Re.L          Re.Q          Fr.L          Fr.Q
##   -0.300240      1.500864     -4.269814      1.098368     -1.927717      0.946857
##          St        Fr.L:St       Fr.Q:St     Re.L:Fr.L     Re.Q:Fr.L     Re.L:Fr.Q
##   -0.997463     -0.007867     -0.043393      3.399373     -0.646249     -2.588037
##   Re.Q:Fr.Q        Re.L:St       Re.Q:St
##          NA     -0.470624     -0.128587
```

```
summary(lm2)$r.squared
```

```
## [1] 0.955342
```

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

```
##
## Call:
## lm(formula = log(R_moment_3) ~ log(St) + Re + Fr + St * Fr +
##     Fr * Re + St * Re, data = train_data)
##
## Coefficients:
## (Intercept)       log(St)          Re.L          Re.Q          Fr.L          Fr.Q
##     4.54701       2.36189      -4.89792       1.52364      -3.75114       1.83513
##          St        Fr.L:St       Fr.Q:St     Re.L:Fr.L     Re.Q:Fr.L     Re.L:Fr.Q
##    -1.72871      -0.09194      -0.03580       6.45803      -1.15825      -4.96308
##   Re.Q:Fr.Q        Re.L:St       Re.Q:St
##          NA      -0.79750      -0.17827
```

```
summary(lm3)$r.squared
```

```
## [1] 0.9458698
```

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

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

```
##
## Coefficients:
## (Intercept)        log(St)           Re.L           Re.Q           Fr.L           Fr.Q
##     9.28287        3.10948       -5.63270        1.94835       -5.55550        2.71540
##          St        Fr.L:St        Fr.Q:St      Re.L:Fr.L      Re.Q:Fr.L      Re.L:Fr.Q
##    -2.36662       -0.17635       -0.01987        9.47956       -1.66019       -7.28081
##    Re.Q:Fr.Q       Re.L:St        Re.Q:St
##          NA       -1.08269       -0.22245
```

```r
summary(lm4)$r.squared
```

```
## [1] 0.9457368
```

All three of our main predictors: Fr, St, and Re, seem to be significant predictors of the R moments. A one percent increase in Stokes number is associated with 0.146% increase in R moment 1, holding all other predictors constant. When the Reynolds number is 224, the R moment 1 is expected to decrease by 403% from when the Reynolds number is 90, holding all other predictors constant. When the Reynolds number is 398, the R moment 1 is expected to increase by 64% compared to when the Reynolds number is 90.

The interaction terms Fr*St and Re*Fr seem to be significant as well. This is consistent with what we expected from our EDA. When Fr is 0.3, the relationship between R moment 1 and St is positive, however, when Fr is infinity, the relationship between R moment 1 and St is negative. Additionally, we see that when the Froud number is 0.3 and the Reynolds number is 224, the R moment 1 is expected to be an additional 24% lower compared to when either of those conditions are not met.

What these results mean for the research question is that Reynolds number (`Re`), gravitational acceleration (`Fr`), and particle characteristic (`St`) all predict cluster formation in clouds to some degree, even after accounting for the effects of each of the other variables. We can also conclude that direction of the relationship between cluster density and Stokes number is different based on what the Froud number is.

Predicted Test Error For R Moments 1-4 With Linear Model
Predicted Mean-Squared Error For R Moment 1

```
## [1] 2.864351e-05
```

Predicted Mean-Squared Error For R Moment 2

```
## [1] 4621.783
```
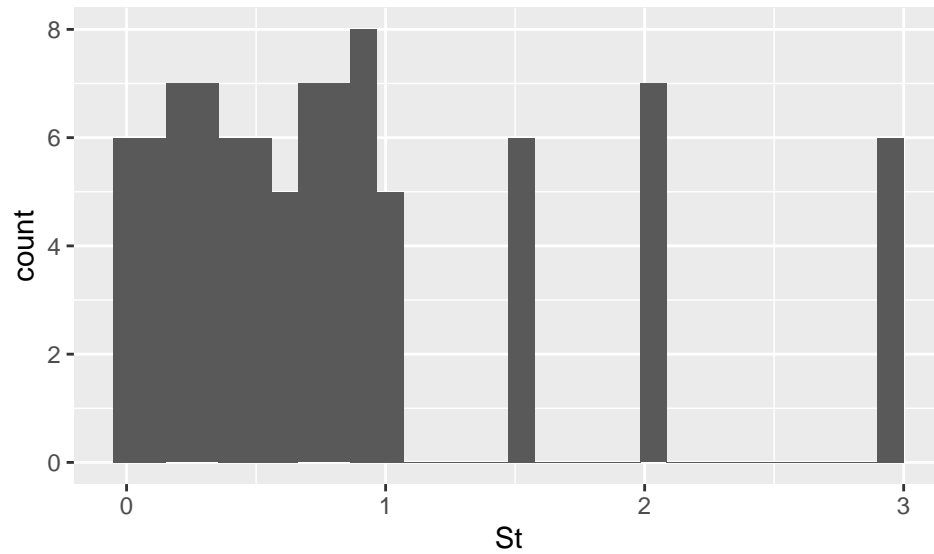
Predicted Mean-Squared Error For R Moment 3

```
## [1] 578718160228
```

Predicted Mean-Squared Error For R Moment 4

```
## [1] 5.488601e+19
```

Appendix

Figure 1.1: Distribution of St

We will try using a log transform on the St variable since the distribution for the St variable is not normally distributed.
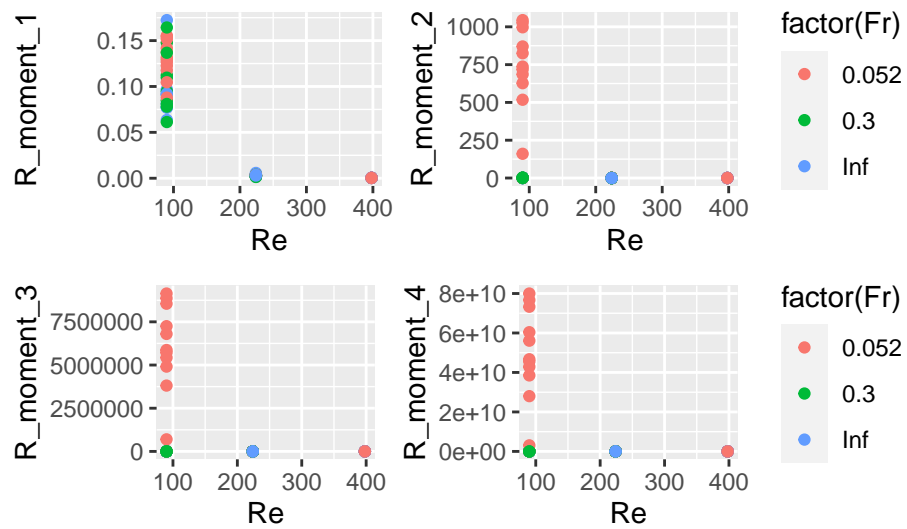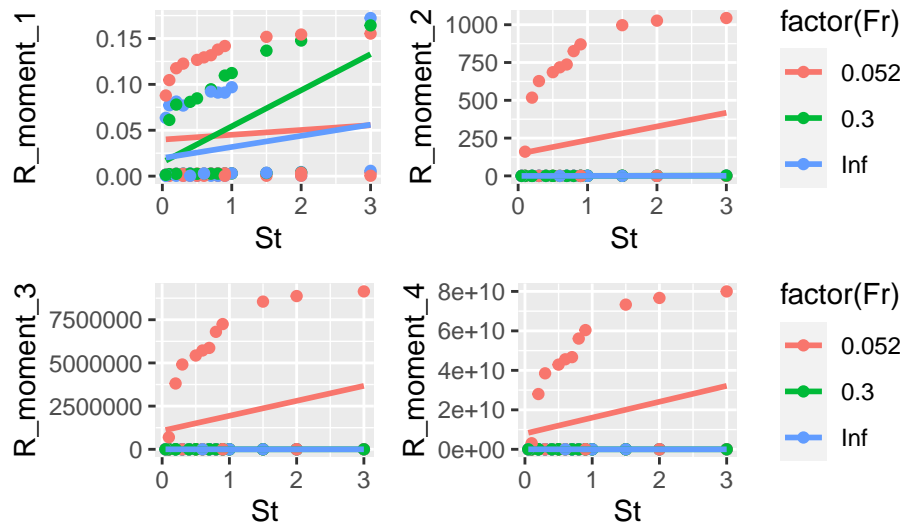
Figure 1.2: R Moments and Re Colored by Fr

Figure 1.3: R Moments and St Colored by Fr

The graphs above show some evidence of interactions, so we will explore interaction terms in our model.

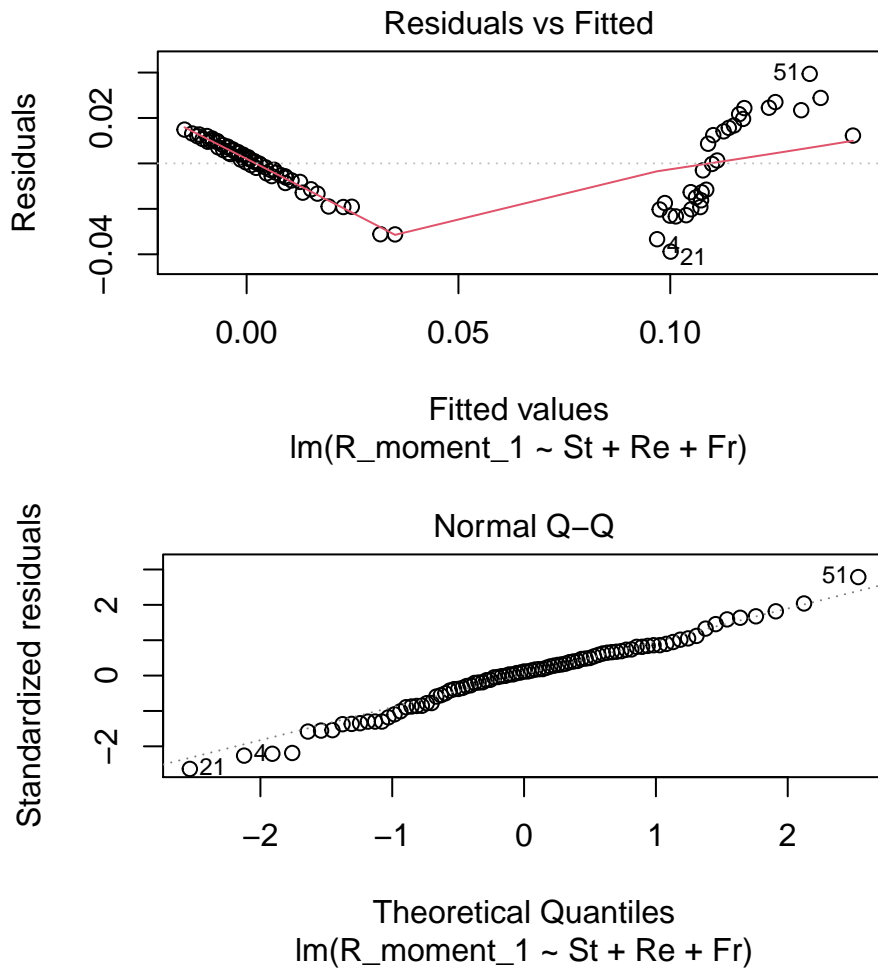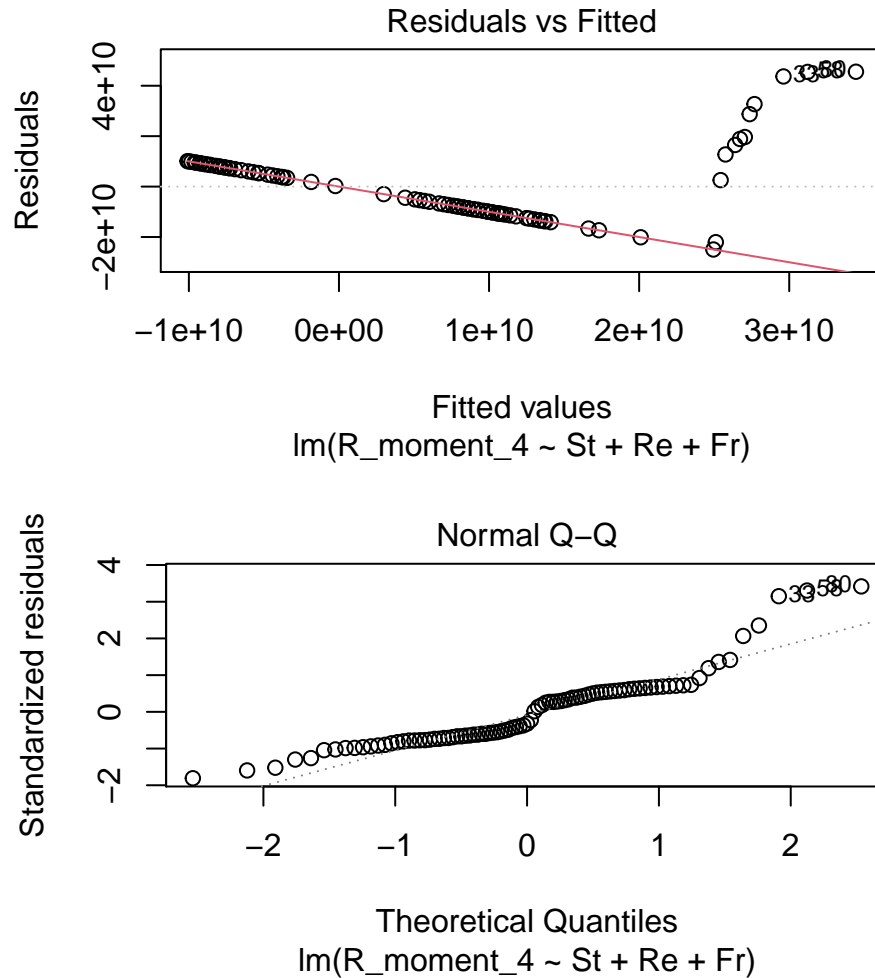**Figure 2.1-2: Diagnostic Plots For Inital Linear Model of R Moment 1**

Residuals vs Fitted

Fitted values
lm(R_moment_1 ~ St + Re + Fr)

Normal Q–Q

Theoretical Quantiles
lm(R_moment_1 ~ St + Re + Fr)

Figure 2.3-4: Diagnostic Plots For Inital Linear Model of R Moment 4

Residuals vs Fitted

Fitted values
lm(R_moment_4 ~ St + Re + Fr)

Normal Q–Q

Theoretical Quantiles
lm(R_moment_4 ~ St + Re + Fr)

Because the linearity condition is not fulfilled in the above Residuals vs. Fitted plots, we will consider performing a log transformation on our response variables (R moments 1-4).

Figure 2.5: Forward Selection With AIC On Linear Models

```
lm1 <- lm(log(R_moment_1) ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
step(lm1, direction = "forward")
```

```
## Start:  AIC=-444.89
## log(R_moment_1) ~ log(St) + Re + Fr + St * Fr + Fr * Re + St *
##      Re


##
## Call:
## lm(formula = log(R_moment_1) ~ log(St) + Re + Fr + St * Fr +
##      Fr * Re + St * Re, data = train_data)
##
```

8

```
## Coefficients:
## (Intercept)        log(St)          Re.L          Re.Q          Fr.L          Fr.Q
##   -5.349488       0.145668     -4.028476      0.644476     -0.102135      0.109493
##          St        Fr.L:St       Fr.Q:St      Re.L:Fr.L     Re.Q:Fr.L     Re.L:Fr.Q
##    0.095896       0.095376     -0.064244      0.242947     -0.076314     -0.077781
##    Re.Q:Fr.Q       Re.L:St       Re.Q:St
##          NA      -0.008897     -0.025325
```

```
lm2 <- lm(log(R_moment_2) ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
#step(lm2, direction = "forward")

lm3 <- lm(log(R_moment_3) ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
#step(lm3, direction = "forward")

lm4 <- lm(log(R_moment_4) ~ log(St) + Re + Fr + St*Fr + Fr*Re + St*Re, data = train_data)
step(lm4, direction = "forward")
```

```
## Start:  AIC=131.52
## log(R_moment_4) ~ log(St) + Re + Fr + St * Fr + Fr * Re + St *
##      Re


##
## Call:
## lm(formula = log(R_moment_4) ~ log(St) + Re + Fr + St * Fr +
##      Fr * Re + St * Re, data = train_data)
##
## Coefficients:
## (Intercept)        log(St)          Re.L          Re.Q          Fr.L          Fr.Q
##     9.28287        3.10948      -5.63270       1.94835      -5.55550       2.71540
##          St        Fr.L:St       Fr.Q:St      Re.L:Fr.L     Re.Q:Fr.L     Re.L:Fr.Q
##    -2.36662       -0.17635      -0.01987       9.47956      -1.66019      -7.28081
##    Re.Q:Fr.Q       Re.L:St       Re.Q:St
##          NA       -1.08269      -0.22245
```