Fibrinolysis_Regression

Kamal Mandava

Contents

```r
library(car)
```

```
## Loading required package: carData
```

```r
library(MASS)
library(leaps)
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.5.2
```

```
##
## Attaching package: 'psych'
```

```
## The following object is masked from 'package:car':
##
##     logit
```

```r
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.5.2

## corrplot 0.95 loaded
```

Introduction

Motivation

Fibrinolysis is a process that involves the breakdown of blood clots, which is essential for maintaining proper blood flow. Understanding the relationship between various biochemical parameters and the rate of fibrinolysis has significant implications for both clinical medicine and pharmaceutical research. By modeling how different kinetic constants and binding affinities influence the median lysis time, we can gain insights into the underlying mechanisms of clot dissolution and potentially identify key factors that could be targeted for therapeutic interventions.

Research Questions

This analysis seeks to answer several important questions:

1. Which biochemical parameters (kinetic constants and binding affinities) are most predictive of median lysis time?
2. Can we develop a reliable statistical model to predict fibrinolysis rates based on these chemical affinity variables?
3. How do transformations of the variables affect the model's predictive power and interpretability?
4. What is the relative importance of different parameters in determining fibrinolysis outcomes?

Dataset Description

The dataset contains measurements from 42 experimental observations, each with 9 independent variables representing various kinetic constants and binding affinities related to the fibrinolysis process. These include parameters such as the rate constant for tissue plasminogen activator (tPA) binding, plasminogen activation rates, dissociation constants for various protein complexes, and the snap proportion. The dependent variable of interest is the median lysis time, which represents the time taken for clot dissolution under specific experimental conditions.

Results

ReadCSV

```r
lysis <- read.csv("target.csv")
```

Summary Statistics

```r
summary(lysis)
```

```
##  Snap.Proportion      ktPAon            kplioff             kplgon       
##  Min.   :0.1250   Min.   : 0.0010   Min.   :   0.576   Min.   : 0.0010  
##  1st Qu.:0.6667   1st Qu.: 0.1000   1st Qu.:  57.600   1st Qu.: 0.1000  
##  Median :0.6667   Median : 0.1000   Median :  57.600   Median : 0.1000  
##  Mean   :0.6319   Mean   : 0.3409   Mean   : 348.645   Mean   : 0.3526  
##  3rd Qu.:0.6667   3rd Qu.: 0.1000   3rd Qu.:  57.600   3rd Qu.: 0.1000  
##  Max.   :0.6667   Max.   :10.0000   Max.   :5760.000   Max.   :10.0000  
##   KdtPAyesplg        KdtPAnoplg        KdPLGnicked       KdPLGintact     
##  Min.   :0.00020   Min.   : 0.0036   Min.   :  0.022   Min.   :   0.38  
##  1st Qu.:0.02000   1st Qu.: 0.3600   1st Qu.:  2.200   1st Qu.:  38.00  
##  Median :0.02000   Median : 0.3600   Median :  2.200   Median :  38.00  
##  Mean   :0.07053   Mean   : 1.2695   Mean   :  7.758   Mean   : 134.00  
##  3rd Qu.:0.02000   3rd Qu.: 0.3600   3rd Qu.:  2.200   3rd Qu.:  38.00  
##  Max.   :2.00000   Max.   :36.0000   Max.   :220.000   Max.   :3800.00  
##  kdeg.and.kncat    medianlysis     
##  Min.   :  0.05   Min.   : 0.2645  
##  1st Qu.:  5.00   1st Qu.: 0.4538  
##  Median :  5.00   Median : 0.5399  
##  Mean   : 17.63   Mean   : 1.6513  
##  3rd Qu.:  5.00   3rd Qu.: 1.6116  
##  Max.   :500.00   Max.   :20.5649
```

```r
sapply(lysis, sd)
```

```
## Snap.Proportion          ktPAon         kplioff          kplgon     KdtPAyesplg 
##       0.1165935       1.5337694    1229.9751663       1.5314357       0.3062871 
##      KdtPAnoplg     KdPLGnicked     KdPLGintact  kdeg.and.kncat     medianlysis 
##       5.5131685      33.6915854     581.9455666      76.5717851       3.3365496
```

```r
describe(lysis)
```

```
##                   vars  n   mean      sd median trimmed  mad  min     max   range
## Snap.Proportion      1 42   0.63    0.12   0.67    0.67 0.00 0.12    0.67    0.54
## ktPAon               2 42   0.34    1.53   0.10    0.09 0.00 0.00   10.00   10.00
## kplioff              3 42 348.64 1229.98  57.60   57.60 0.00 0.58 5760.00 5759.42
## kplgon               4 42   0.35    1.53   0.10    0.10 0.00 0.00   10.00   10.00
## KdtPAyesplg          5 42   0.07    0.31   0.02    0.02 0.00 0.00    2.00    2.00
## KdtPAnoplg           6 42   1.27    5.51   0.36    0.36 0.00 0.00   36.00   36.00
## KdPLGnicked          7 42   7.76   33.69   2.20    2.20 0.00 0.02  220.00  219.98
## KdPLGintact          8 42 134.00  581.95  38.00   38.00 0.00 0.38 3800.00 3799.62
## kdeg.and.kncat       9 42  17.63   76.57   5.00    5.00 0.00 0.05  500.00  499.95
## medianlysis         10 42   1.65    3.34   0.54    0.92 0.28 0.26   20.56   20.30
##                    skew kurtosis     se
## Snap.Proportion -3.22     9.36   0.02
## ktPAon           5.94    34.45   0.24
## kplioff          4.04    14.84 189.79
## kplgon           5.95    34.50   0.24
## KdtPAyesplg      5.95    34.50   0.05
## KdtPAnoplg       5.95    34.50   0.85
## KdPLGnicked       5.95    34.50   5.20
## KdPLGintact       5.95    34.50  89.80
## kdeg.and.kncat    5.95    34.50  11.82
## medianlysis       4.50    21.92   0.51
```

```
pairs(lysis[, c("medianlysis", "Snap.Proportion", "ktPAon", "kplioff", "kplgon", "KdtPAyesplg", "KdtPAn
```

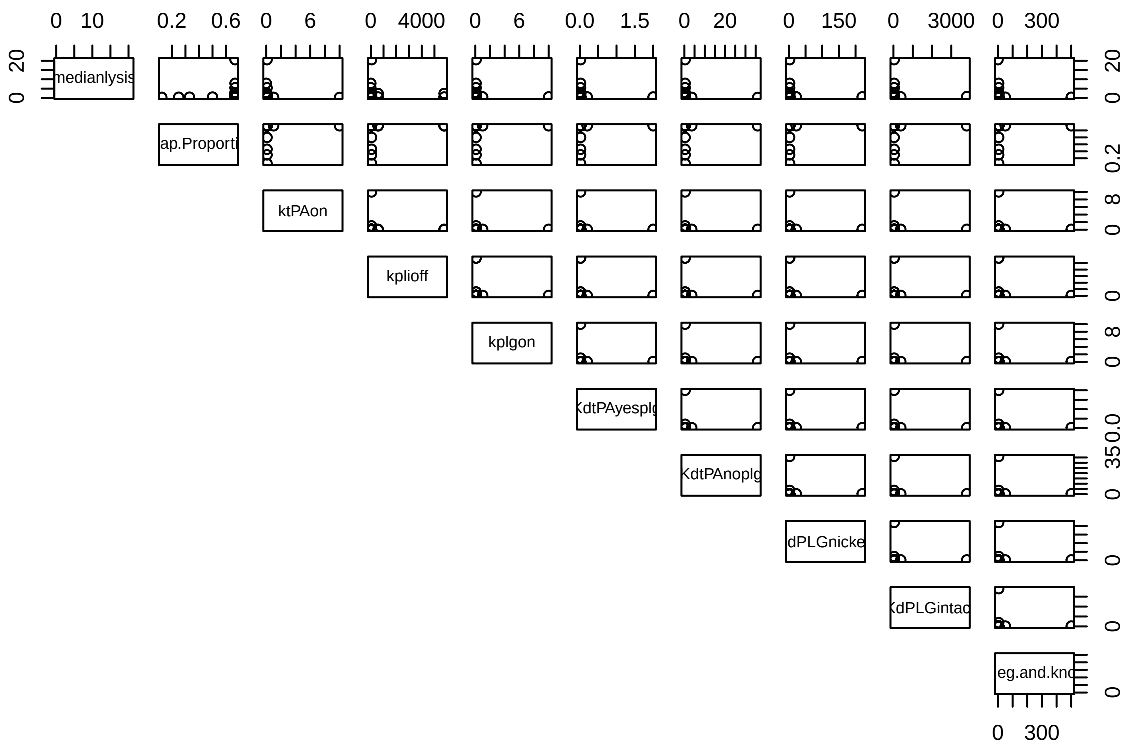


Figure 1: Pairwise scatterplots of all variables. This plot matrix provides a visual overview of the relationships between variables, helping to identify potential linear association before formal modeling. Visually we can see that there is no linear relationship present between median lysis and other variables which means that we might need to transform the variables necessarily.

Fitting the Original Model

```
Org_model <- lm(medianlysis ~., data = lysis)
summary(Org_model)
```

```
##
## Call:
## lm(formula = medianlysis ~ ., data = lysis)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.8079 -1.6196 -0.5921  0.2176 18.3605
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -0.6004791  3.1540195  -0.190    0.850
## Snap.Proportion  4.3878629  4.9810891   0.881    0.385
## ktPAon          -0.2165594  0.3752938  -0.577    0.568
## kplioff         -0.0001510  0.0004705  -0.321    0.750
## kplgon          -0.1764185  0.3758938  -0.469    0.642
## KdtPAyesplg     -0.9089443  1.8794688  -0.484    0.632
## KdtPAnoplg      -0.0580584  0.1044149  -0.556    0.582
## KdPLGnicked     -0.0080217  0.0170861  -0.469    0.642
## KdPLGintact     -0.0004098  0.0009892  -0.414    0.681
## kdeg.and.kncat  -0.0043838  0.0075179  -0.583    0.564
##
## Residual standard error: 3.664 on 32 degrees of freedom
## Multiple R-squared:  0.05877,    Adjusted R-squared:  -0.2059
## F-statistic: 0.222 on 9 and 32 DF,  p-value: 0.989
```

Global F-test:

$H_0 : \beta_1 = \beta_2 = ... = 0$

H_1 : ALOI

$F = 0.222$

$p = 0.989 > 0.05$

Fail to Reject H_0, insufficient evidence to conclude the model is useful

This model is statistically insignificant

Correlation

```
cor(lysis)
```

```
##                 Snap.Proportion      ktPAon     kplioff      kplgon KdtPAyesplg
## Snap.Proportion      1.00000000  0.04802077  0.07235905  0.05044730  0.05044730
## ktPAon               0.04802077  1.00000000 -0.04588820 -0.02653830 -0.02653830
## kplioff              0.07235905 -0.04588820  1.00000000 -0.03998866 -0.03998866
## kplgon               0.05044730 -0.02653830 -0.03998866  1.00000000 -0.02787931
## KdtPAyesplg          0.05044730 -0.02653830 -0.03998866 -0.02787931  1.00000000
## KdtPAnoplg           0.05044730 -0.02653830 -0.03998866 -0.02787931 -0.02787931
## KdPLGnicked          0.05044730 -0.02653830 -0.03998866 -0.02787931 -0.02787931
## KdPLGintact          0.05044730 -0.02653830 -0.03998866 -0.02787931 -0.02787931
## kdeg.and.kncat       0.05044730 -0.02653830 -0.03998866 -0.02787931 -0.02787931
## medianlysis          0.11862230 -0.07600711 -0.01946428 -0.05631483 -0.05884849
##                  KdtPAnoplg KdPLGnicked KdPLGintact kdeg.and.kncat medianlysis
## Snap.Proportion  0.05044730  0.05044730  0.05044730     0.05044730  0.11862230
## ktPAon          -0.02653830 -0.02653830 -0.02653830    -0.02653830 -0.07600711
## kplioff         -0.03998866 -0.03998866 -0.03998866    -0.03998866 -0.01946428
## kplgon          -0.02787931 -0.02787931 -0.02787931    -0.02787931 -0.05631483
## KdtPAyesplg     -0.02787931 -0.02787931 -0.02787931    -0.02787931 -0.05884849
## KdtPAnoplg       1.00000000 -0.02787931 -0.02787931    -0.02787931 -0.07169104
## KdPLGnicked     -0.02787931  1.00000000 -0.02787931    -0.02787931 -0.05634297
## KdPLGintact     -0.02787931 -0.02787931  1.00000000    -0.02787931 -0.04654581
## kdeg.and.kncat  -0.02787931 -0.02787931 -0.02787931     1.00000000 -0.07649436
## medianlysis     -0.07169104 -0.05634297 -0.04654581    -0.07649436  1.00000000
```

```
corrplot(cor(lysis), method = "color", t1.cex=0.7)
```

```
## Warning in text.default(pos.xlabel[, 1], pos.xlabel[, 2], newcolnames, srt =
## tl.srt, : "t1.cex" is not a graphical parameter
```

```
## Warning in text.default(pos.ylabel[, 1], pos.ylabel[, 2], newrownames, col =
## tl.col, : "t1.cex" is not a graphical parameter
```

```
## Warning in title(title, ...): "t1.cex" is not a graphical parameter
```

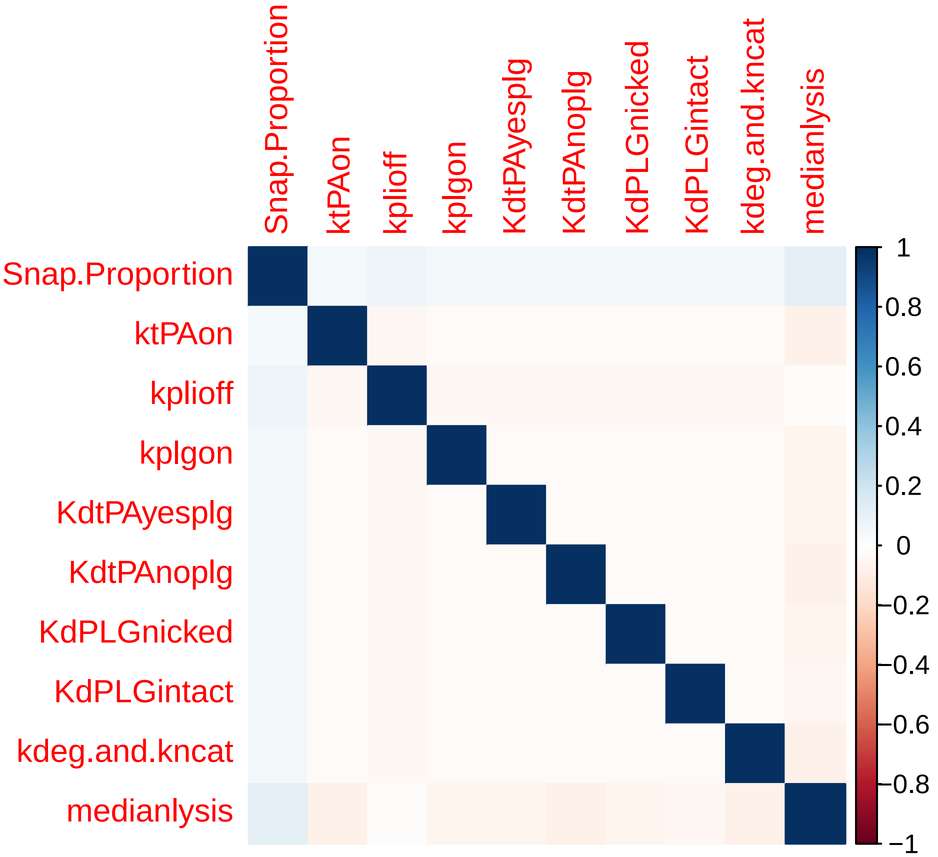
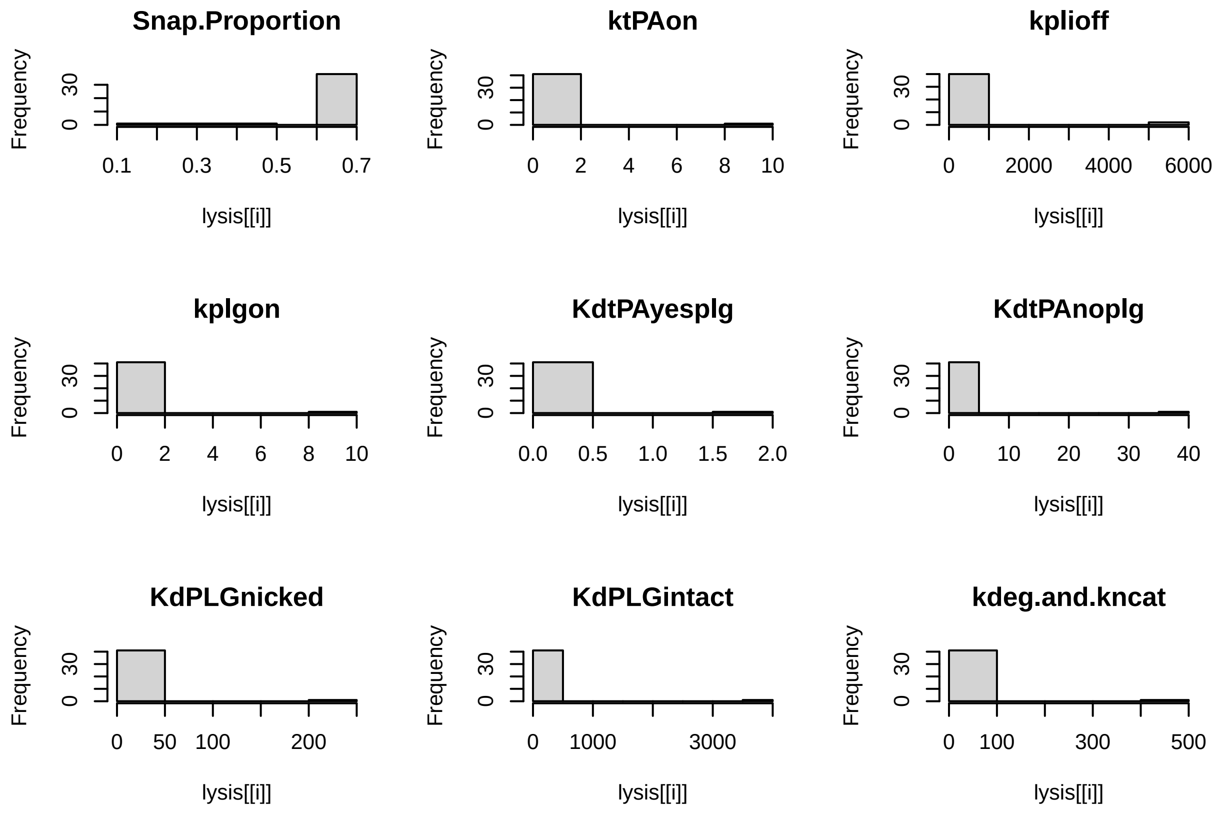


Figure 2: Correlation matrix visualization. The color coded correlation plot reveals the strength and direction of linear relationships between variables. Strong correlations (close to ±1) appear in darker colors, while weak correlations appear lighter. This helps identify potential multicollinearity issues that could affect regression modeling, which in this case there is no correlation relationship among the independent variables.

Diagnostics

```
par(mfrow=c(3,3))
for(i in 1:9) hist(lysis[[i]], main=names(lysis)[i])
```



```
hist(lysis$medianlysis, main="medianlysis")
```

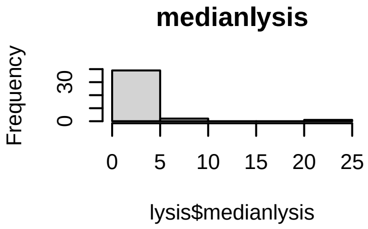


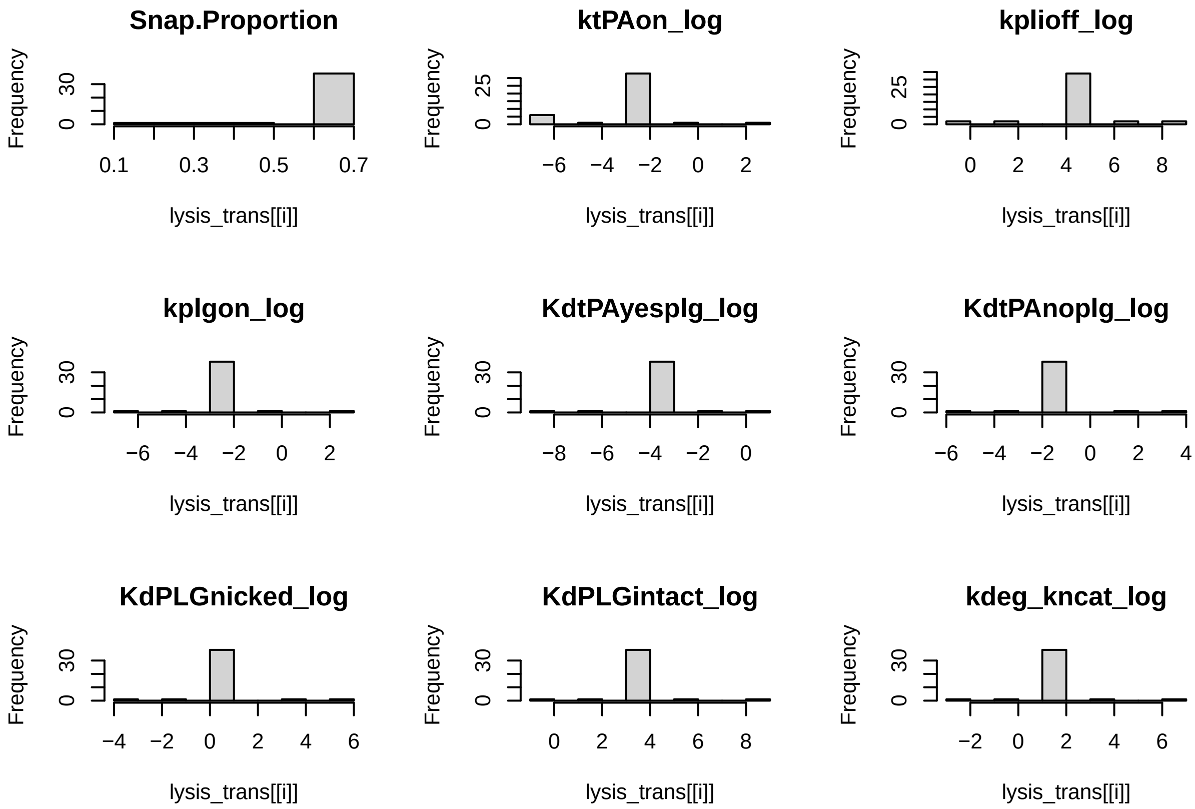
Figure 3: Distribution histograms for all variables. These histograms reveal the distributional properties of each variable, including skewness, potential outliers, and whether transformations might be necessary to meet normality assumptions for regression analysis. Log transformation is used to address right skewness and improve linearity, Log transformation compresses large values more than small ones, reducing the impact of extreme values and making the distribution more symmetric.

Actions to tackle skewness and make the normality assumpltion valid

```
# Snap.Proportion is a proportion (0-1), not a rate constant or concentration. It may already be bounde




lysis$ktPAon_log <- log(lysis$ktPAon)
lysis$kplioff_log <- log(lysis$kplioff)
lysis$kplgon_log <- log(lysis$kplgon)
lysis$KdtPAyesplg_log <- log(lysis$KdtPAyesplg)
lysis$KdtPAnoplg_log <- log(lysis$KdtPAnoplg)
lysis$KdPLGnicked_log <- log(lysis$KdPLGnicked)
lysis$KdPLGintact_log <- log(lysis$KdPLGintact)
lysis$kdeg_kncat_log <- log(lysis$kdeg.and.kncat)

# Keep transformed predictors + original Snap.Proportion
lysis_trans <- data.frame(
  Snap.Proportion = lysis$Snap.Proportion,
  ktPAon_log = lysis$ktPAon_log,
  kplioff_log = lysis$kplioff_log,
  kplgon_log = lysis$kplgon_log,
  KdtPAyesplg_log = lysis$KdtPAyesplg_log,
  KdtPAnoplg_log = lysis$KdtPAnoplg_log,
  KdPLGnicked_log = lysis$KdPLGnicked_log,
  KdPLGintact_log = lysis$KdPLGintact_log,
  kdeg_kncat_log = lysis$kdeg_kncat_log,
  medianlysis = lysis$medianlysis
)

# Quick check
summary(lysis_trans)
```

```
##  Snap.Proportion    ktPAon_log      kplioff_log        kplgon_log
##  Min.   :0.1250   Min.   :-6.908   Min.   :-0.5516   Min.   :-6.908
##  1st Qu.:0.6667   1st Qu.:-2.303   1st Qu.: 4.0535   1st Qu.:-2.303
##  Median :0.6667   Median :-2.303   Median : 4.0535   Median :-2.303
##  Mean   :0.6319   Mean   :-2.851   Mean   : 4.0535   Mean   :-2.303
##  3rd Qu.:0.6667   3rd Qu.:-2.303   3rd Qu.: 4.0535   3rd Qu.:-2.303
##  Max.   :0.6667   Max.   : 2.303   Max.   : 8.6587   Max.   : 2.303
##  KdtPAyesplg_log   KdtPAnoplg_log   KdPLGnicked_log   KdPLGintact_log
##  Min.   :-8.5172   Min.   :-5.627   Min.   :-3.8167   Min.   :-0.9676
##  1st Qu.:-3.9120   1st Qu.:-1.022   1st Qu.: 0.7885   1st Qu.: 3.6376
##  Median :-3.9120   Median :-1.022   Median : 0.7885   Median : 3.6376
##  Mean   :-3.9120   Mean   :-1.022   Mean   : 0.7885   Mean   : 3.6376
##  3rd Qu.:-3.9120   3rd Qu.:-1.022   3rd Qu.: 0.7885   3rd Qu.: 3.6376
##  Max.   : 0.6931   Max.   : 3.584   Max.   : 5.3936   Max.   : 8.2428
##  kdeg_kncat_log    medianlysis
##  Min.   :-2.996   Min.   : 0.2645
##  1st Qu.: 1.609   1st Qu.: 0.4538
##  Median : 1.609   Median : 0.5399
##  Mean   : 1.609   Mean   : 1.6513
##  3rd Qu.: 1.609   3rd Qu.: 1.6116
##  Max.   : 6.215   Max.   :20.5649
```

```
par(mfrow=c(3,3))
for(i in 1:9) hist(lysis_trans[[i]], main=names(lysis_trans)[i])
```



So based on this transformation i can interpret that:

There are concentrated bars after transformation, this is mainly due to the fact that: 1) Limited distinct values in the original data. 2) Small sample size (n = 42)

I don't necessarily consider it to be a problem. For regression You don't need perfectly normal distributions for predictors. The thing that matters the most is the relationships are more linear and variance is more stable. The concentration in the center is often better than the original extreme right skew.

Hence the log transformation successfully addressed the extreme right skewness observed in most of the kinetic parameters.

Model Fitting after Log Transformation

```
#Model Fitting

lysis_lm <- lm(medianlysis ~., data = lysis_trans)

summary(lysis_lm)
```

```
##
## Call:
## lm(formula = medianlysis ~ ., data = lysis_trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.4956 -1.0023 -0.7485  0.0317 10.5510
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.513138   3.375143   1.041   0.3057
## Snap.Proportion  2.847863   3.544619   0.803   0.4277
## ktPAon_log      -0.380244   0.218683  -1.739   0.0917 .
## kplioff_log     -0.476968   0.255970  -1.863   0.0716 .
## kplgon_log       0.013815   0.361997   0.038   0.9698
## KdtPAyesplg_log -0.005168   0.361997  -0.014   0.9887
## KdtPAnoplg_log  -0.241107   0.361997  -0.666   0.5102
## KdPLGnicked_log -0.004321   0.361997  -0.012   0.9905
## KdPLGintact_log -0.022688   0.361997  -0.063   0.9504
## kdeg_kncat_log  -1.839663   0.361997  -5.082 1.57e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.636 on 32 degrees of freedom
## Multiple R-squared:  0.5129, Adjusted R-squared:  0.3759
## F-statistic: 3.744 on 9 and 32 DF,  p-value: 0.002637
```

The initial model with all predictors and the untransformed response variable performed poorly. The global F-test p-value of 0.989 indicates that the model as a whole is not statistically significant at any conventional level ($\alpha = 0.05$). This means we cannot reject the null hypothesis that all regression coefficients are zero. These results clearly indicate that transformations or alternative modeling approaches are necessary.

After the Logrithmic Transformation of independent variables:

Global F-test:

$H_0 : \beta_1 = \beta_2 = ... = 0$

H_1 : ALOI

$F = 3.744$

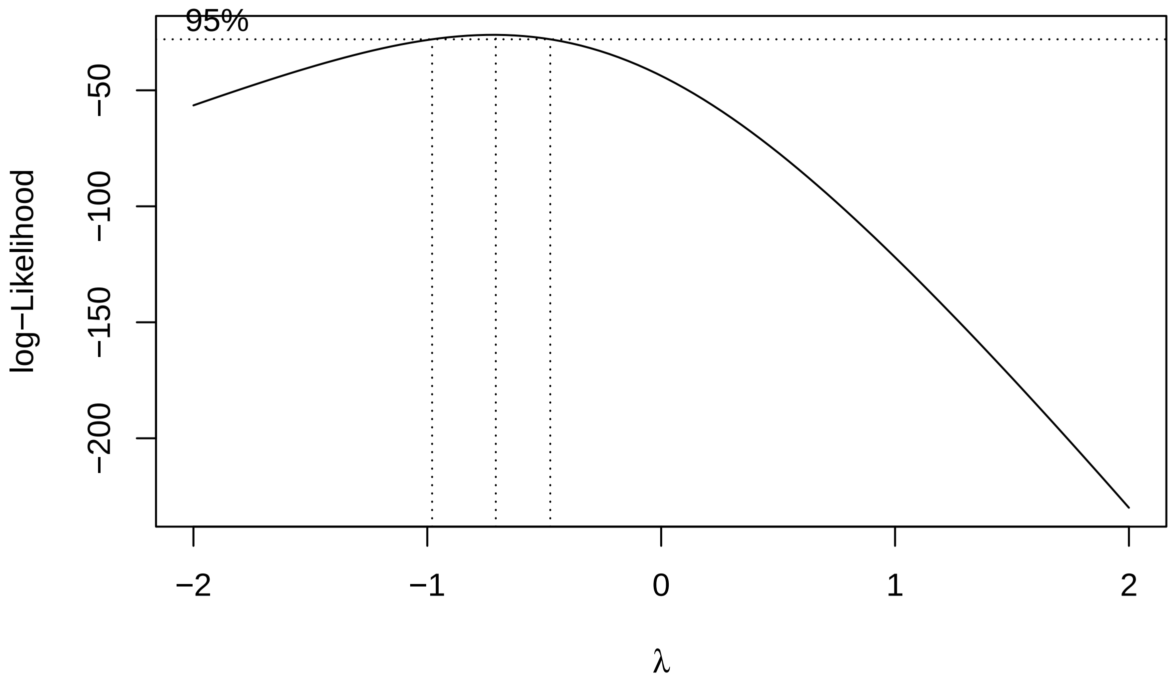
$p = 0.002637 < 0.05$

Reject H_0, sufficient evidence to conclude the model is useful

The Transformed regression model performed well and is statistically significant. with an Adjusted R squared of 0.3759 which means that the regression model explains 37.59% of variance in median lysis which is far better than our original model, but i consider to improve the adjusted R squared since only ~37% of variance is being explained

boxcox transformation

```
boxcox(lysis_lm)
```



The Box-Cox transformation analysis suggests an optimal lambda value close to -1, which corresponds to the reciprocal transformation (1/Y). This indicates that the response variable (median lysis) also requires transformation to better meet the assumptions of linear regression.

Reciprocal Transformation of Y

```
lysis_trans$medianlysis <- 1/(lysis_trans$medianlysis) # this would push the inverse of medianlysis to

# refitting the model with transformed data

trans_lysis_lm <- lm(medianlysis ~., data = lysis_trans)

summary(trans_lysis_lm)
```

```
##
## Call:
## lm(formula = medianlysis ~ ., data = lysis_trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.07513 -0.12941  0.02219  0.20119  0.75753
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.47635    0.54362   6.395 3.48e-07 ***
## Snap.Proportion -2.47596    0.57092  -4.337 0.000135 ***
## ktPAon_log       0.34749    0.03522   9.865 3.15e-11 ***
## kplioff_log      0.11895    0.04123   2.885 0.006948 **
## kplgon_log      -0.06423    0.05831  -1.102 0.278858
## KdtPAyesplg_log  0.01949    0.05831   0.334 0.740365
## KdtPAnoplg_log   0.41269    0.05831   7.078 4.99e-08 ***
## KdPLGnicked_log  0.01389    0.05831   0.238 0.813157
## KdPLGintact_log  0.02104    0.05831   0.361 0.720549
## kdeg_kncat_log   0.34842    0.05831   5.976 1.16e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4245 on 32 degrees of freedom
## Multiple R-squared:  0.8732, Adjusted R-squared:  0.8375
## F-statistic: 24.48 on 9 and 32 DF,  p-value: 6.366e-12
```

After applying the reciprocal transformation to the dependent variable, the model performance improved substantially. This suggests that the model with all predictors does have some predictive power. The adjusted R-squared increased to 0.8375, meaning the model now explains approximately 83.75% of the variance in the transformed median lysis time. While this is a significant improvement from the initial original model, i would like to take a step further and perform analysis on which predictors best explain the variance in median lysis. Also the model still indicates that a substantial portion of the variability remains unexplained. This is mainly due to the experimental data that has been obtained. The experimental data had observations that have been scaled 10x times for each experiments which might have been marked as potential outliers.

Further Analysis

1) VIF

```
# Checking for multi-collinearity
vif(trans_lysis_lm) # No multi-collinearity present all are less than 10 (cut-off)
```

```
## Snap.Proportion      ktPAon_log     kplioff_log      kplgon_log KdtPAyesplg_log
##        1.007932        1.007932        1.000000        1.000000        1.000000
##  KdtPAnoplg_log KdPLGnicked_log KdPLGintact_log  kdeg_kncat_log
##        1.000000        1.000000        1.000000        1.000000
```

The variance inflation factors (VIFs) for all predictors are below the commonly used threshold of 10, indicating that multi-collinearity is not a serious concern in this model. This is an important finding because high multi-collinearity can make coefficient estimates unstable and difficult to interpret. The fact that all VIFs are reasonable suggests that each predictor provides somewhat unique information about the response variable.

2) Variable screening

```
fwd <- lm(medianlysis ~ 1, data = lysis_trans)
out_fwd <- step(fwd, scope = ~ Snap.Proportion + ktPAon_log + kplioff_log + kplgon_log + KdtPAyesplg_lo
## Start:  AIC=5.35
## medianlysis ~ 1
##
##                   Df Sum of Sq    RSS      AIC
## + ktPAon_log       1   19.0872 26.396 -15.5064
## + KdtPAnoplg_log   1    9.0297 36.454  -1.9480
## + kdeg_kncat_log   1    6.4363 39.047   0.9385
## + Snap.Proportion  1    4.9349 40.549   2.5231
## <none>                         45.484   5.3467
## + kplioff_log      1    1.5003 43.983   5.9379
## + kplgon_log       1    0.2187 45.265   7.1443
## + KdPLGintact_log  1    0.0235 45.460   7.3250
## + KdtPAyesplg_log  1    0.0201 45.464   7.3281
## + KdPLGnicked_log  1    0.0102 45.473   7.3373
##
## Step:  AIC=-15.51
## medianlysis ~ ktPAon_log
##
##                   Df Sum of Sq    RSS     AIC
## + KdtPAnoplg_log   1    9.0297 17.367 -31.091
## + kdeg_kncat_log   1    6.4363 19.960 -25.245
## + Snap.Proportion  1    3.3899 23.007 -19.279
## + kplioff_log      1    1.5003 24.896 -15.964
## <none>                         26.396 -15.506
## + kplgon_log       1    0.2187 26.178 -13.856
## + KdPLGintact_log  1    0.0235 26.373 -13.544
## + KdtPAyesplg_log  1    0.0201 26.376 -13.539
## + KdPLGnicked_log  1    0.0102 26.386 -13.523
##
## Step:  AIC=-31.09
## medianlysis ~ ktPAon_log + KdtPAnoplg_log
##
##                   Df Sum of Sq    RSS     AIC
## + kdeg_kncat_log   1    6.4363 10.931 -48.537
## + Snap.Proportion  1    3.3899 13.977 -38.211
## + kplioff_log      1    1.5003 15.866 -32.885
## <none>                         17.367 -31.091
## + kplgon_log       1    0.2187 17.148 -29.623
## + KdPLGintact_log  1    0.0235 17.343 -29.147
## + KdtPAyesplg_log  1    0.0201 17.347 -29.139
## + KdPLGnicked_log  1    0.0102 17.357 -29.115
##
## Step:  AIC=-48.54
## medianlysis ~ ktPAon_log + KdtPAnoplg_log + kdeg_kncat_log
##
##                   Df Sum of Sq     RSS     AIC
## + Snap.Proportion  1    3.3899  7.5406 -62.130
## + kplioff_log      1    1.5003  9.4301 -52.738
## <none>                         10.9305 -48.537
## + kplgon_log       1    0.2187 10.7118 -47.386
## + KdPLGintact_log  1    0.0235 10.9070 -46.627
## + KdtPAyesplg_log  1    0.0201 10.9103 -46.614
## + KdPLGnicked_log  1    0.0102 10.9202 -46.576
##
```

```
## Step:  AIC=-62.13
## medianlysis ~ ktPAon_log + KdtPAnoplg_log + kdeg_kncat_log +
##     Snap.Proportion
##
##                   Df Sum of Sq    RSS     AIC
## + kplioff_log      1   1.50034 6.0402 -69.448
## <none>                         7.5406 -62.130
## + kplgon_log       1   0.21872 7.3219 -61.366
## + KdPLGintact_log  1   0.02348 7.5171 -60.261
## + KdtPAyesplg_log  1   0.02014 7.5204 -60.242
## + KdPLGnicked_log  1   0.01024 7.5303 -60.187
##
## Step:  AIC=-69.45
## medianlysis ~ ktPAon_log + KdtPAnoplg_log + kdeg_kncat_log +
##     Snap.Proportion + kplioff_log
##
##                   Df Sum of Sq    RSS     AIC
## <none>                         6.0402 -69.448
## + kplgon_log       1  0.218720 5.8215 -68.997
## + KdPLGintact_log  1  0.023475 6.0168 -67.611
## + KdtPAyesplg_log  1  0.020138 6.0201 -67.588
## + KdPLGnicked_log  1  0.010236 6.0300 -67.519
```

```
summary(out_fwd)
```

```
##
## Call:
## lm(formula = medianlysis ~ ktPAon_log + KdtPAnoplg_log + kdeg_kncat_log +
##     Snap.Proportion + kplioff_log, data = lysis_trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.07513 -0.13543  0.04008  0.18990  0.75753
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.63550    0.40749   8.922 1.19e-10 ***
## ktPAon_log       0.34749    0.03398  10.225 3.42e-12 ***
## KdtPAnoplg_log   0.41269    0.05625   7.336 1.20e-08 ***
## kdeg_kncat_log   0.34842    0.05625   6.194 3.83e-07 ***
## Snap.Proportion -2.47596    0.55084  -4.495 6.95e-05 ***
## kplioff_log      0.11895    0.03978   2.990    0.005 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4096 on 36 degrees of freedom
## Multiple R-squared:  0.8672, Adjusted R-squared:  0.8488
## F-statistic: 47.02 on 5 and 36 DF,  p-value: 8.568e-15
```

The forward stepwise selection procedure identified a more parsimonious model by systematically adding variables that improve model fit.

The AIC decreased from -42.69 (for the null model with only an intercept) to -50.18 for the final selected model. Since lower AIC values indicate better model fit while penalizing for model complexity, this substantial drop confirms that the selected predictors significantly improve the model beyond what we'd expect from chance alone. The final model from forward selection shows little increase in Adjusted R squared when compared to full model, indicating stronger statistical significance. suggesting that while we've identified the most important predictors, there may be inherent limitations in how well these variables can predict fibrinolysis rates.

3) All-Possible-Regressions Selection Procedure

```
subset_model <- regsubsets(medianlysis ~ kdeg_kncat_log + kplioff_log +
                           Snap.Proportion + ktPAon_log + KdtPAnoplg_log +
                           KdtPAyesplg_log + kplgon_log + KdPLGnicked_log +
                           KdPLGintact_log,
                           data = lysis_trans)

summary(subset_model)$outmat
```

```
##          kdeg_kncat_log kplioff_log Snap.Proportion ktPAon_log KdtPAnoplg_log
## 1  ( 1 ) " "            " "         " "             "*"        " "
## 2  ( 1 ) " "            " "         " "             "*"        "*"
## 3  ( 1 ) "*"            " "         " "             "*"        "*"
## 4  ( 1 ) "*"            " "         "*"             "*"        "*"
## 5  ( 1 ) "*"            "*"         "*"             "*"        "*"
## 6  ( 1 ) "*"            "*"         "*"             "*"        "*"
## 7  ( 1 ) "*"            "*"         "*"             "*"        "*"
## 8  ( 1 ) "*"            "*"         "*"             "*"        "*"
##          KdtPAyesplg_log kplgon_log KdPLGnicked_log KdPLGintact_log
## 1  ( 1 ) " "             " "        " "             " "
## 2  ( 1 ) " "             " "        " "             " "
## 3  ( 1 ) " "             " "        " "             " "
## 4  ( 1 ) " "             " "        " "             " "
## 5  ( 1 ) " "             " "        " "             " "
## 6  ( 1 ) " "             "*"        " "             " "
## 7  ( 1 ) " "             "*"        " "             "*"
## 8  ( 1 ) "*"             "*"        " "             "*"
```

```
names(summary(subset_model))
```

```
## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
```

```
which.min(summary(subset_model)$bic) # gives 1 variable model; which means 1 variable model is best acc
```

```
## [1] 5
```

```
which.max(summary(subset_model)$adjr2) # gives 5 variable model; which means 5 variable model is best a
```

```
## [1] 6
```

```
plot(subset_model, scale = "adjr2")
```

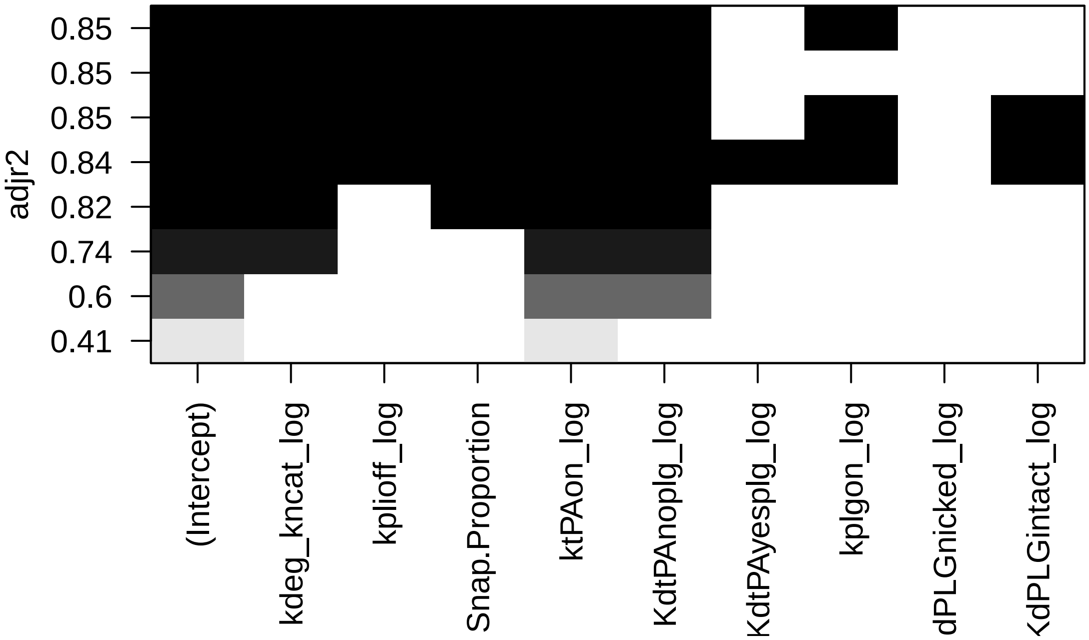


Figure 4: All-possible regressions comparison plot. This visualization shows how adjusted R-squared changes as we add more variables to the model.

My Main goal is to have a good prediction model, so i am considering adjusted-R-squared over BIC which is a 6 variable model.

Modelling Best Possible Regression Model

```
best_model <- lm(medianlysis ~ kdeg_kncat_log + kplioff_log +Snap.Proportion + ktPAon_log + KdtPAnoplg_

summary(best_model)
```

```
##
## Call:
## lm(formula = medianlysis ~ kdeg_kncat_log + kplioff_log + Snap.Proportion +
##     ktPAon_log + KdtPAnoplg_log + kplgon_log, data = lysis_trans)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.07513 -0.13543  0.03373  0.20983  0.75753
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.48761    0.42573   8.192 1.19e-09 ***
## kdeg_kncat_log     0.34842    0.05601   6.221 3.96e-07 ***
## kplioff_log      0.11895    0.03961   3.003  0.00491 **
## Snap.Proportion -2.47596    0.54844  -4.515 6.89e-05 ***
## ktPAon_log       0.34749    0.03384  10.270 4.21e-12 ***
## KdtPAnoplg_log   0.41269    0.05601   7.368 1.29e-08 ***
## kplgon_log      -0.06423    0.05601  -1.147  0.25927
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4078 on 35 degrees of freedom
## Multiple R-squared:  0.872,  Adjusted R-squared:  0.8501
## F-statistic: 39.74 on 6 and 35 DF,  p-value: 3.309e-14
```

Global F-Test and Model Summary

The final best model includes six predictors: `kdeg_kncat_log`, `kplioff_log`, `Snap.Proportion`, `ktPAon_log`, `KdtPAnoplg_log`, and `kplgon_log`. Global F-test

$H_0 : \beta_1 = \beta_2 = \cdots = \beta_9 = 0$

$H_1 :$ ALOI

$F = 39.74$

$p < 0.05 $

Reject H_0, sufficient evidence to conclude the model is useful

Regression Equation

Based on the final model, the regression equation can be written as:

1/medianlysis = 3.48761 + 0.34842 * (kdeg_kncat_log) + 0.11895 * (kplioff_log) - 2.47596 (Snap.Proportion) + 0.34749 * (ktPAon_log) + 0.41269 * (KdtPAnoplg_log) - 0.06423 * (kplgon_log)

Each coefficient (β_i) represents the expected change in the reciprocal of median lysis time for a one-unit increase in the corresponding predictor, holding all other predictors constant. Since we're working with log-transformed predictors, the interpretation involves percentage changes rather than absolute changes, which is more appropriate for rate constants and binding affinities.

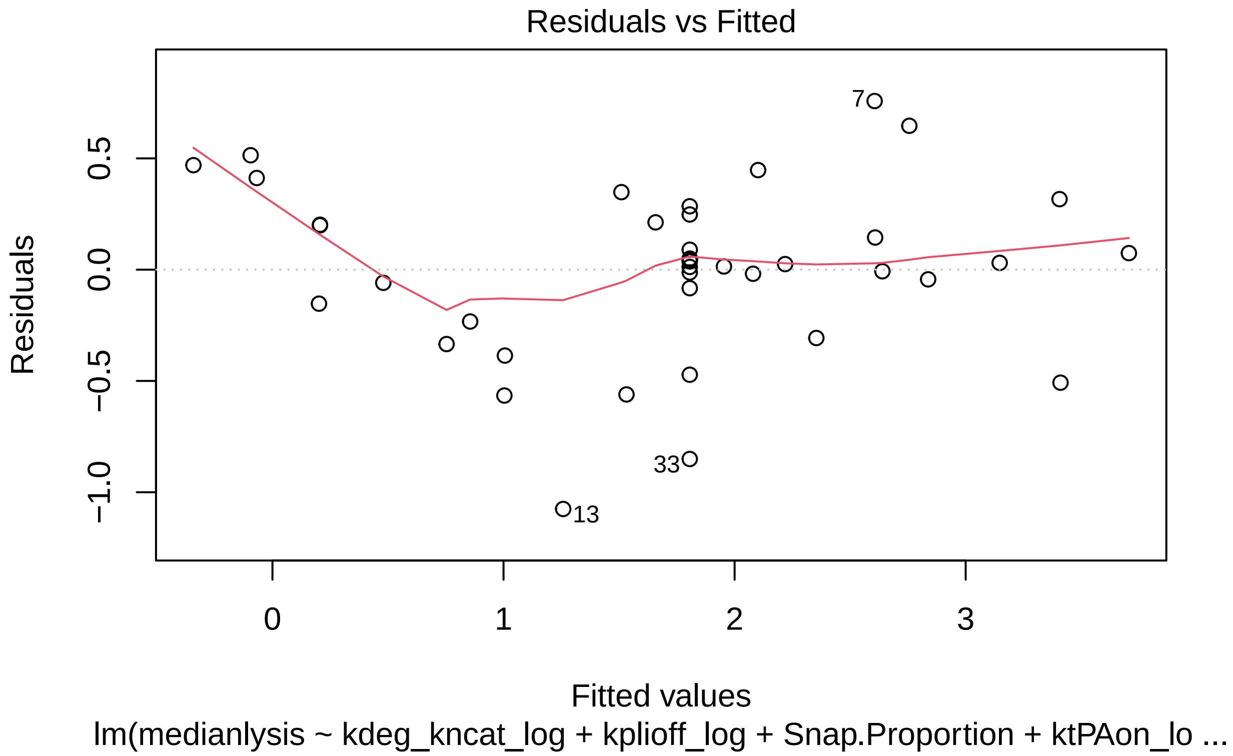
Interpretation of Individual Coefficients

The individual coefficient estimates and their significance tests tell us which predictors have statistically significant relationships with the response variable. A significant coefficient (typically p < 0.05) suggests that the corresponding variable has a meaningful association with median lysis time after accounting for all other variables in the model. The sign of the coefficient indicates the direction of the relationship - positive coefficients suggest that increases in the predictor are associated with increases in the response (or decreases in median lysis time, since we're modeling the reciprocal), while negative coefficients suggest the opposite relationship.

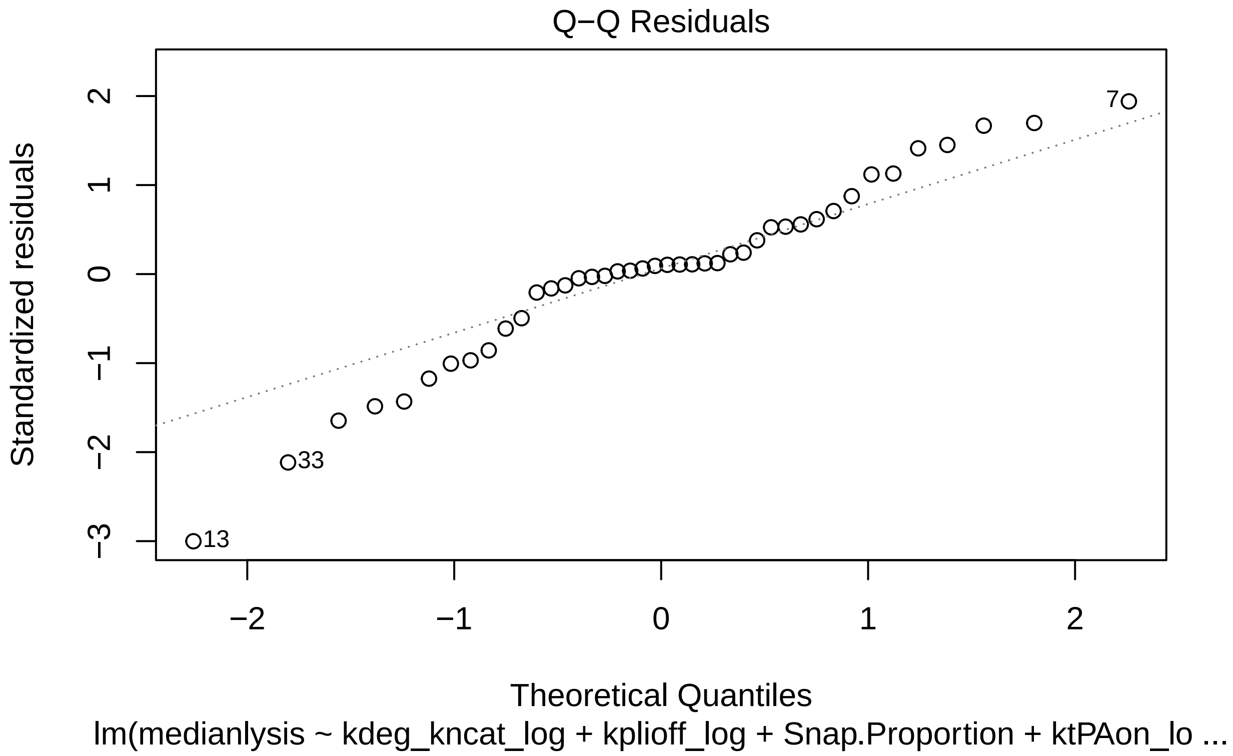
The Best Adjusted R Squared that i can get is from a 6 predictors model

##Residual Analysis

```r
# Constant variance check
plot(best_model, 1)
```



```r
# Normality check
plot(best_model, 2)
```



```
#Outlier check
rstd <- rstandard(best_model)
which(abs(rstd)>3)
```

```
## 13
## 13
```

```
# High Leverage
h <- hatvalues(best_model)
# cut-off value would be: (2(k+1))/n = (2*(5+1))/42 = 12/42 = 0.2857143
which(h > 0.2857143)
```

```
##  5 14 17 22 25 34 37 38 42
##  5 14 17 22 25 34 37 38 42
```

```
# influentual Observations
ckd <- cooks.distance(best_model)
which(ckd > 0.5)
```

```
## named integer(0)
```

```
which(ckd > 1)
```

```
## named integer(0)
```

```
plot(best_model, 5)
```

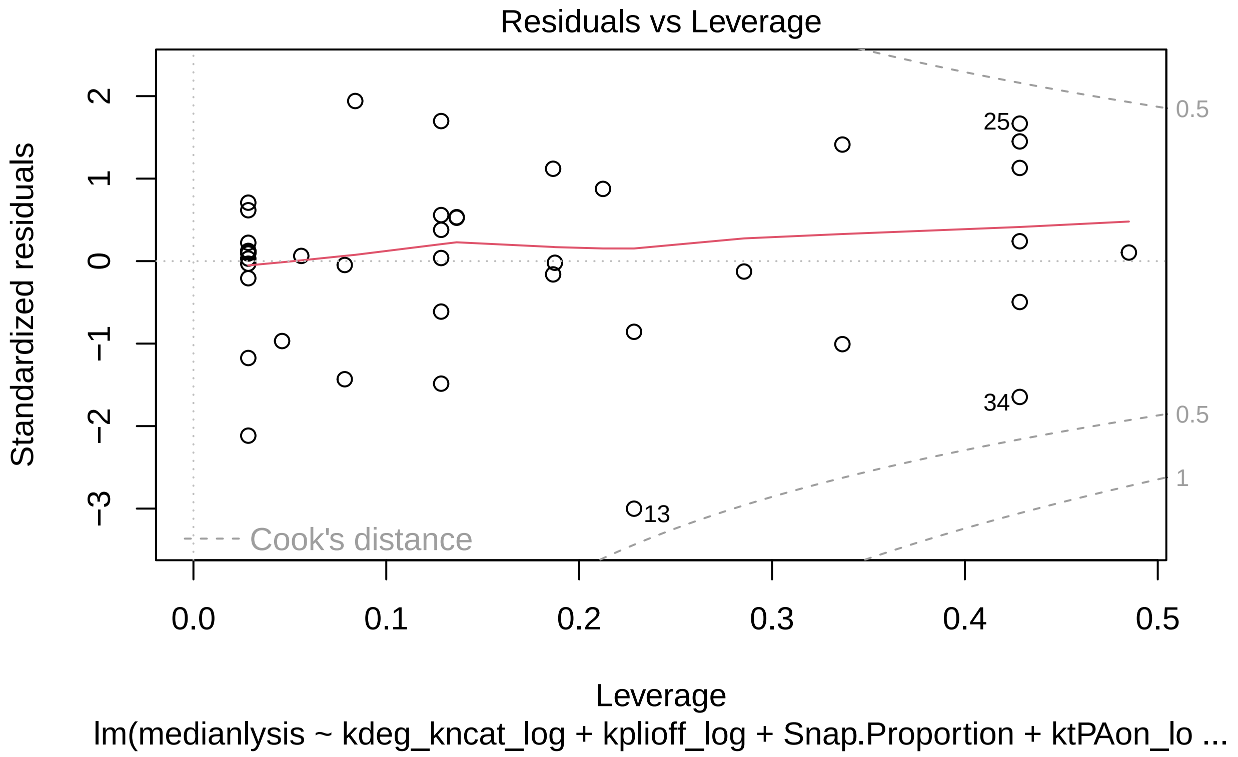


Figure 5: Residual diagnostic plots. These plots assess whether the model assumptions are met and identify potential problematic observations.

Residual Analysis Summary

Constant Variance: The residuals vs fitted plot shows a wavy pattern and non-constant spread (wider at extremes, narrower in the middle), indicating mild heteroscedasticity.

Normality: The Q-Q plot shows an S-shape with deviations in both tails (observations 13, 33 in lower tail; 7 in upper tail), suggesting heavier tails than normal.

Outliers: Observation 13 has a standardized residual < -3.

High Leverage: Nine observations (5, 14, 17, 22, 25, 34, 37, 38, 42) have high leverage.

Influence: No observations exceed Cook's distance thresholds of 0.5 or 1.0, though observations 34 and 250 are near the 0.5 contour and may have some influence.

Discussion

Main Findings and Concluding Remarks

This analysis successfully developed a predictive model for fibrinolysis rates, achieving an adjusted R-squared of 0.8501 (85.01%) with six predictors. This represents a dramatic improvement from the original model, which had a negative adjusted R-squared (-0.2059) and was statistically insignificant (p = 0.989). The transformation approach proved essential: log transformations of the kinetic parameters addressed extreme right skewness, while the reciprocal transformation of the response variable (suggested by Box-Cox analysis) was crucial for achieving both statistical significance and high explanatory power.

The final model identifies five highly significant predictors: the degradation and catalytic rate constant (kdeg_kncat_log), plasminogen dissociation rate (kplioff_log), snap proportion (negative relationship), tPA binding rate (ktPAon_log), and dissociation constant for tPA without plasminogen (KdtPAnoplg_log). Interestingly, removing three variables from the full model (KdtPAyesplg_log, KdPLGnicked_log, KdPLGintact_log) actually improved the adjusted R-squared from 0.8375 to 0.8501, indicating these variables added noise rather than meaningful information. The sixth variable (kplgon_log) is included in the model but is not statistically significant (p = 0.259), suggesting it may have limited predictive value.

The most important finding is that fibrinolysis rates depend on multiple kinetic constants and binding affinities working together, rather than any single parameter. These variables represent different aspects of the complex biochemical network involved in clot dissolution. From a practical standpoint, this model could be useful for predicting fibrinolysis outcomes in experimental settings, with the 85% variance explained providing strong predictive capability.

Limitations of the Analysis

I should acknowledge several limitations.

First, the sample size of 42 observations is relatively small for a model with 6 predictors. While sufficient for this analysis, a larger sample would provide more stable coefficient estimates and greater statistical significance.

Second, despite explaining 85% of the variance, 15% remains unexplained. I am actually satisfied with the 85% of variance getting explained especially for a complex biological data like this. But the rest of the 15% remains unexplained which might be a bit might be achieved by adding more observations into the data set.

Third, residual analysis revealed violations of some regression assumptions. The residuals vs fitted plot shows mild heteroscedasticity (non-constant variance), and the Q-Q plot shows an S-shape with heavier tails than normal. While these violations don't necessarily invalidate the model, they suggest that standard errors and p-values should be interpreted with some caution. The presence of one outlier (observation 13) and nine high leverage points also suggests some unusual cases in the data, though no observations exceed Cook's distance thresholds, indicating no highly influential points.

Fourth, model selection involved multiple steps (forward selection, all-possible regressions), and different criteria suggested different optimal models. BIC favored a 5-variable model, while adjusted R-squared favored a 6-variable model. The choice to prioritize adjusted R-squared for prediction purposes highlights the subjective nature of model selection.

Finally, this analysis is correlational and observational. We cannot establish causal relationships, and the associations could be confounded by unmeasured variables or reflect correlations rather than direct causal effects.

What Would Be Done Differently

If redoing this project, several improvements could be made:

First, I would collect a larger sample size (ideally 100-150 observations) to provide more statistical power and allow for proper train-test validation (Cross Validation), which would give a more realistic assessment of predictive performance on unseen data.

Second, I might also explore robust regression methods that are less sensitive to outliers.

Third, I would consider alternative modeling approaches. Given the mild heteroscedasticity, weighted least squares regression might be appropriate. If nonlinear relationships are suspected, polynomial terms or interaction effects could be explored. Machine learning approaches like random forests could provide complementary insights, though with reduced interpretability.

Fourth, I would explore collecting additional variables that might be relevant to fibrinolysis but weren't included in the original dataset.

Finally, I would conduct a more thorough sensitivity analysis to understand how robust the conclusions are to different modeling choices, transformations, and handling of outliers.

Future Research Directions

Several research directions emerged from this analysis. First, expanding the dataset to include more observations and potentially more variables would allow for more sophisticated modeling and more reliable conclusions. This could include additional kinetic parameters, environmental factors (like temperature, pH, or ion concentrations), or characteristics of the clot itself.

Second, investigating nonlinear relationships and potential interactions between variables could reveal more complex patterns. For instance, the effect of one kinetic constant might depend on the value of another, suggesting that interaction terms could improve the model. Polynomial terms or spline functions could capture curvilinear relationships that the linear model misses.

Third, exploring the biological mechanisms underlying the statistical associations could provide deeper insights. Understanding why certain parameters are predictive could lead to testable hypotheses about the fibrinolysis process and potentially identify new therapeutic targets.

Fourth, developing more sophisticated models that account for the heteroscedasticity and other assumption violations could improve both prediction accuracy and the reliability of statistical inferences. This might involve generalized least squares, robust regression methods, or Bayesian approaches.

Finally, translating these statistical findings into practical applications could involve developing predictive tools for clinical or pharmaceutical research. If the model can reliably predict fibrinolysis rates, it could be used to screen potential therapeutic compounds or optimize experimental conditions without requiring extensive laboratory work.

In conclusion, while this analysis has limitations, it provides a solid foundation for understanding the relationships between biochemical parameters and fibrinolysis rates. The transformation approach and model selection process successfully improved the model from statistically insignificant to explaining 85% of the variance. The identified predictors represent a starting point for further investigation, and the methodological approaches developed here could be refined and extended in future work.