STAT 512 - Project Report

Michael Lee, David Luo, Sriram Malathi

December 14, 2022

1 Introduction

The core of computers, the Central Processing Unit (CPU), is the most fundamental part of modern computing systems. Since practically every field of industry in today's world heavily relies on computers to do the heavy-lifting of computationally expensive tasks, being able to monitor and analyze the performance of CPUs is critical. While there exist many benchmarks to analyze the performance of CPUs, it is oftentimes impractical and sometimes impossible to perform these benchmarks (e.g. before the computer has actually been manufactured). Further, there are hundreds of different attributes of modern computer hardware that can impact performance, and no single feature can be chiefly held as a catch-all measure of quality.

This dataset and the original paper utilizing it aimed to estimate Relative Performance, a dimensionless value that could be used to compare CPUs with vastly different attributes to each other. This could be a useful metric to compare CPUs with other CPUs manufactured by the same company, as well as a competitive metric to compare CPU performance across different manufacturers [1]. These published relative performance values were obtained via BYTE an influential technology magazine that reported on the latest tech news of the time. This dataset various CPU features to estimate this value with the aim to be approximately close to the published values. Note that this dataset and all its values are from the years 1981-1984.

The purpose of this study is to be able to obtain relative performance metrics on CPUs without needing a source like BYTE Magazine to publish them. For example, if hardware specifications are available but the CPU has not been evaluated and its performance metric published, one could use this model to obtain a prediction of the relative performance themselves. This could apply to CPUs whose specifications are known but the hardware itself is not even on the market yet.

2 Methods

2.1 Description of Dataset

The data has 8 attributes and one response variable, as well as an additional column containing the predicted relative performance values from the original paper. We will ignore these values for our report, since we aim to obtain these values ourselves.
Response:

1. `PRP` — published relative performance (integer)
2. `ERP` — predictions from original paper (integer, ignored for our analysis)

Predictors:

1. `Vendor` — name of the CPU's vendor; one of 30 values
2. `Model` — name of the CPU's model
3. `MYCT` — machine cycle time in nanoseconds (integer)
4. `MMIN` — minimum main memory in kilobytes (integer)
5. `MMAX` — maximum main memory in kilobytes (integer)
6. `CACH` — cache memory in kilobytes (integer)
7. `CHMIN` — minimum number of channels (integer)
8. `CHMAX` — maximum number of channels (integer)

The data consists of information about 209 different CPUs. Note that there are no missing values in the data.

2.2 Preliminary Exploratory Analyses

The first step taken was to look at the scatterplot matrix comparing the integer-valued predictors with each other and the response variable, `PRP`. We immediately note that there are no obvious linear relationships among any of the regressors, and observe that the majority of points lie close together in the bottom left corner of each scatterplot. This suggests that the data is not normally distributed, and instead heavily positively skewed. We confirm this more clearly by looking at the histogram of each variable, shown in figure 1.

While looking at the scatterplot matrix, we consider if there are any predictors that appear to be good candidates for transformation. In particular, `MYCT` looks like an inverse transformation would be appropriate. Using an inverse transformation plot (figure 2), we find that the optimal λ value for a power transformation is -1.12, and that an inverse power transformation is indeed appropriate. Based off the intrinsic meaning of several variables, we could consider using linear combinations to reduce the dimensionality of our data. In particular, the regressors `MMIN` and `MMAX` as well as regressors `CHMIN` and `CHMAX` could potentially be

reduced into simple averages, as intuitively it would make sense that the average captures similar information of either pair while combining two dimensions of our data into one. We create two more attributes $\texttt{MAVG} = \frac{\texttt{MMIN}+\texttt{MMAX}}{2}$ and $\texttt{CHAVG} = \frac{\texttt{CHMIN}+\texttt{CHMAX}}{2}$ and consider using them when we create our model. Note that we round these averages to the nearest integer since they represent memory sizes and channel counts, which in the context of this problem do not make sense to be decimal values.

We omit the regressor `Model`, since this value is unique for every entity in the dataset. Intuitively, the name of the model would obviously have no bearing on the functional utility of the CPU, and therefore have no impact on the relative performance metric.

2.3 Models

We considered two models:

$$\texttt{PRP} = \beta_0 + \beta_1 \texttt{invMYCT} + \beta_2 \texttt{CACH} + \beta_3 \texttt{MAVG} + \beta_4 \texttt{CHAVG} \tag{1}$$

$$\sqrt{\texttt{PRP}} = \beta_0 + \beta_1 \texttt{invMYCT} + \beta_2 \texttt{CACH} + \beta_3 \texttt{MAVG} + \beta_4 \texttt{CHAVG} \tag{2}$$

The second model was determined to be potentially appropriate after examining a inverse response plot of the first model, as seen in figure 3, which suggested that an appropriate power transformation using $\lambda = 0.5$ could be a good candidate for transformation.

We examined the residual plots of both models (figure 4), which clearly shows that the second model is a much better fit. There is a very clear pattern in the residuals of model 1, whereas the residual plot of model 2 more closely resembles a null plot. We also compared the marginal model plots of either model (figure 5, figure 6), which further shows that the second model better fits the data for each included regressor than the first model. From here onwards, we only consider model 2.

After deciding that model 2 was much more appropriate, we checked the residual plots per regressor, along with tests for curvature (figure 7). We find that the residual plots for each regressor are appropriate, except for `CACH`, whose test for curvature results in a significant p-value. We tried to diagnose this by adding a quadratic term for `CACH` as well as experimenting with different interaction terms, but none of our attempted solutions improved the model.

2.4 Outliers and Influential Points

We then performed an outlier test and obtain Cook's Distance for each point in the dataset, the results of which are shown in figure 8. We see that there are outliers with indexes 32 and 169 — however, we see that index 32 is also an influential point as per its Cook's Distance. As such, we can remove point 169, but it would not be appropriate to do the same for point 32. Before removing the point, we examined its features to check if there was anything obviously wrong with the recorded values. Compared to the means and medians of each regressor and the response variable, nothing seemed obviously wrong with the point, although it's published performance rating seemed lower than it should have been given its specifications. Since there was nothing clearly wrong, there is not much to suggest that removing the point is necessary. We created a model without the point just to check what

the impact would be, and it did not appear to have a significant impact. We continue using model 2 (including both outliers in the dataset).

2.5 Inferential Methods

One of the transformations had included making use of the Box-Cox method. This was due to the response variable failing to meet the normality assumption. Since Box-Cox transformation is particularly useful since it allows compute correlation coefficient of a normal probability plot, while simultaneously varying the parameter λ, from the equation of the regression model below,

$$\mathtt{Y}^{\lambda} = X\beta + \epsilon$$

This allows us to identify the optimal transformation required to mitigate the violation of the normality assumption of the response variable. We also made extensive use of marginal model plots to determine the fit of the model. they contain scatter plots of two variables, its fit, and the predicted values as a function of independent variables. The parameters retrieved from the analysis of variance was then used to identify how much variability of data is contributed by each regressor. However, the statistical significance of the estimated parameters for the regression model was established through F-tests. The p-value thus recovered was used to infer whether the predictors account for the changes in the response. For all our tests we use a standard significance level of $\alpha = 0.05$.

3 Results

We will first briefly discuss the how our model compares to other possible regressions we could have examined. Then we will examine the results of our regression model by examining its estimated coefficients, then consider the results of our ANOVA table as well as our model's effectiveness when compared to the true values of our response variable as well as the predicted values obtained from the regression found in the original paper (represented by the `ERP` attribute, as explained in the Methods section). Finally, we will discus the interpretation of the model as it relates to the context of the original problem.

3.1 Selection Criteria

Since there are a relatively few number of regressors being used in this analysis, it did not seem unreasonable to perform an all-possible regression test to see if a different subset of regressors could be better. However, we find that our current model (including all of our chosen transformed regressors) was most appropriate since it had the lowest AIC metric. The AIC information criterion along with the respective adjusted R-squared values can be found in figure 9.

3.2 Parameter Estimates

We continue with our analysis using model (2), using the root-transformed response. The summary table of this model, along with confidence intervals for the parameter estiamtes,

can be found in figures 10 and 11 respectively. We obtain the following regression:

$$\sqrt{\texttt{PRP}} = 3.359 + 51.59\texttt{invMYCT} + 0.036\texttt{CACH} + 0.0004\texttt{MAVG} + 0.046\texttt{CHAVG} \tag{3}$$

Finally, we obtained an ANOVA table for the model, seen in figure 12. We can see here that all of our chosen regressors are shown to be statistically significant, and that there is evidence to suggest that their slopes are not zero.

To get a general idea of how well our regression predicts relative performance, we made several graphs which compare the predicted values from our regression to the published (actual) values as well as the estimated values `ERP`, which shows us how similar our predicted values were to the paper's predicted values. We also compared `PRP` to `ERP` to compare how our model performed relative to the regression found in the original paper. The results of these comparisons can be made in figure 13. We see that our model was actually fairly similar in performance to the regression of the original paper.

4 Discussion

The goal of this regression was to be able to predict relative performance based on four factors; namely: cycle time, average main memory, cache memory, and average channels. One could have used such a regression to inform decisions on what should be prioritized when manufacturing a new system.

Unfortunately, in order for the model to be the most accurate, the response variable PRP was power transformed by 0.5 (root transformed). While this improved the model overall, it means that predicting PRP is outside the scope of this model and is not very useful. When predicting square root PRP the variance is very low. However, when trying to use the model to predict PRP by squaring the result, the variance increases dramatically.

As seen in the residual plots (figure 4), for higher values the model seems to have a poorer fit, and while the square root transformation remedies it partially, there is stil evident curvature. The model is accurate for PRP under 200. So for machines with PRP greater than 200 the prediction will be less accurate. As this data is quite old (1987) it is likely that modern systems will have predicted PRPs that do not make sense.

One possibility for why the model is not accurate for values above this PRP is that those systems are likely specialized machines that have a different balance of memory, cache, and cycle time. Thus the model performs relatively well for computers with average specifications, but not as well when compared to machines with high-performing specifications.

From the regression it seems that the most significant contributor to the square root of PRP is the inverse of cycle time in nanoseconds. This can intuitively be interpreted as speed; as such, this suggests that minimizing cycle time, or increasing computational speed, is the highest contributor to PRP.

That being said, however, it is difficult to compare cycle time to other features because their units and spread are different. The variance of inverse cycle time is close to zero while the variance of average memory is in closer to 200,000 which could explain why a one unit increase in inverse cycle time contributes much more than all other variables.

If further analyses were to be done, it may be wise to find how a percent change in each of these variables leads to a change in PRP so they could be used to compare against each other.

5 References

[1] Dheeru Dua and Casey Graff. UCI machine learning repository. http://archive.ics.uci.edu/ml, 2017. Accessed: 2020-11-20.

6 Appendix A: Code

```
library(ggplot2)
library(car)

cpu <- read.csv('machine.data')

# Exploratory analysis
head(cpu, 10)
pairs(cpu[,3:9])
ggplot(melt(cpu[,3:9]), aes(x=value)) +
  facet_wrap(~variable, scales='free_x') +
  geom_histogram()

# Transforming predictors
with(cpu, invTranPlot(MYCT, PRP))
cpu$invMYCT = (1/cpu$MYCT)
cpu$MAVG = ((cpu$MMIN + cpu$MMAX) / 2)
cpu$CHAVG = round((cpu$CHMIN + cpu$CHMAX) / 2)
cpu$sqCACH = cpu$CACH^2

m1 <- lm(PRP ~ invMYCT + CACH + MAVG + CHAVG, data=cpu)
m2 <- lm(sqrt(PRP) ~ invMYCT + CACH + MAVG + CHAVG, data=cpu)

# Transforming response
boxCox(m1, grid=TRUE)
inverseResponsePlot(m1)

# Diagnostics comparison
par(mfrow=c(1,2))
rp1 <- residualPlot(m1)
rp2 <- residualPlot(m2)

residualPlots(m2)

mmps(m1)
mmps(m2)

# All possible regressions
k <- ols_step_all_possible(m2)
all <- data.frame(k$predictors, k$adjr, k$aic)
```

```r
all[with(all, order(k.aic)),]

# Outliers and influential points
outlierTest(m2)
influenceIndexPlot(m2)

# Check what model looks like without outlier
cpu2 <- cpu[-c(169),]
m3 <- lm(sqrt(PRP) ~ invMYCT + CACH + MAVG + CHAVG, data=cpu2)

# Results and Tests
summary(m2)
confint(m2)
anova(m2)

# Comparison to actual values and ERP from original paper
cpu$pred <- (predict(m2))^2
par(mfrow=c(1, 3))
plot(cpu$ERP, cpu$PRP, xlab='ERP', ylab='PRP')
title('PRP vs. ERP')
abline(a=0,b=1)
plot(cpu$pred, cpu$PRP, xlab='predicted', ylab='PRP')
title('PRP vs. Predicted')
abline(a=0,b=1)
plot(cpu$pred, cpu$ERP, xlab='predicted', ylab='ERP')
title('ERP vs. Predicted')
abline(a=0,b=1)
```

7 Appendix B: Output

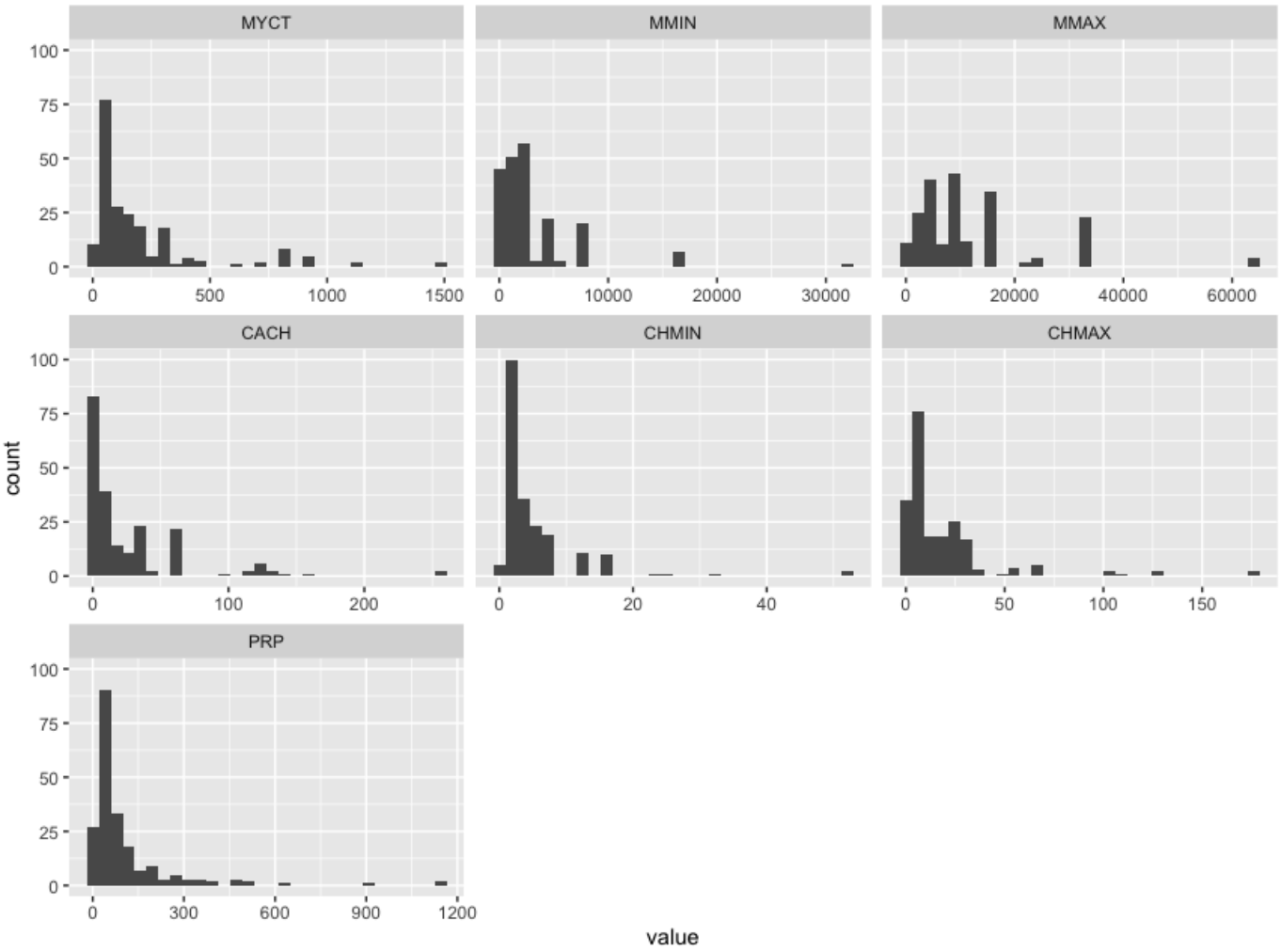

Figure 1: Distribution of each integer-valued regressor

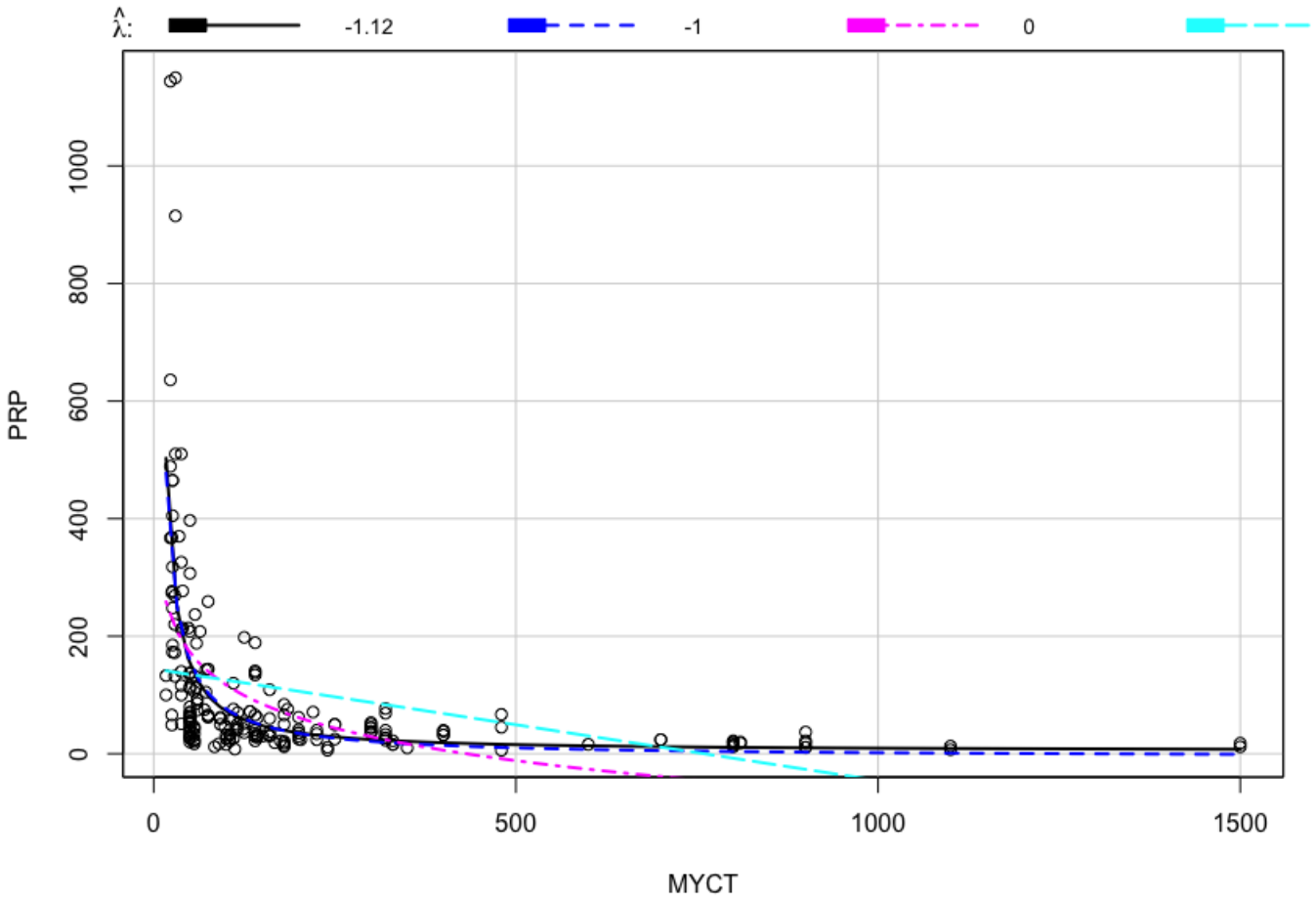

Figure 2: Inverse Transformation Plot of Regressor `MYCT` on Response `PRP`

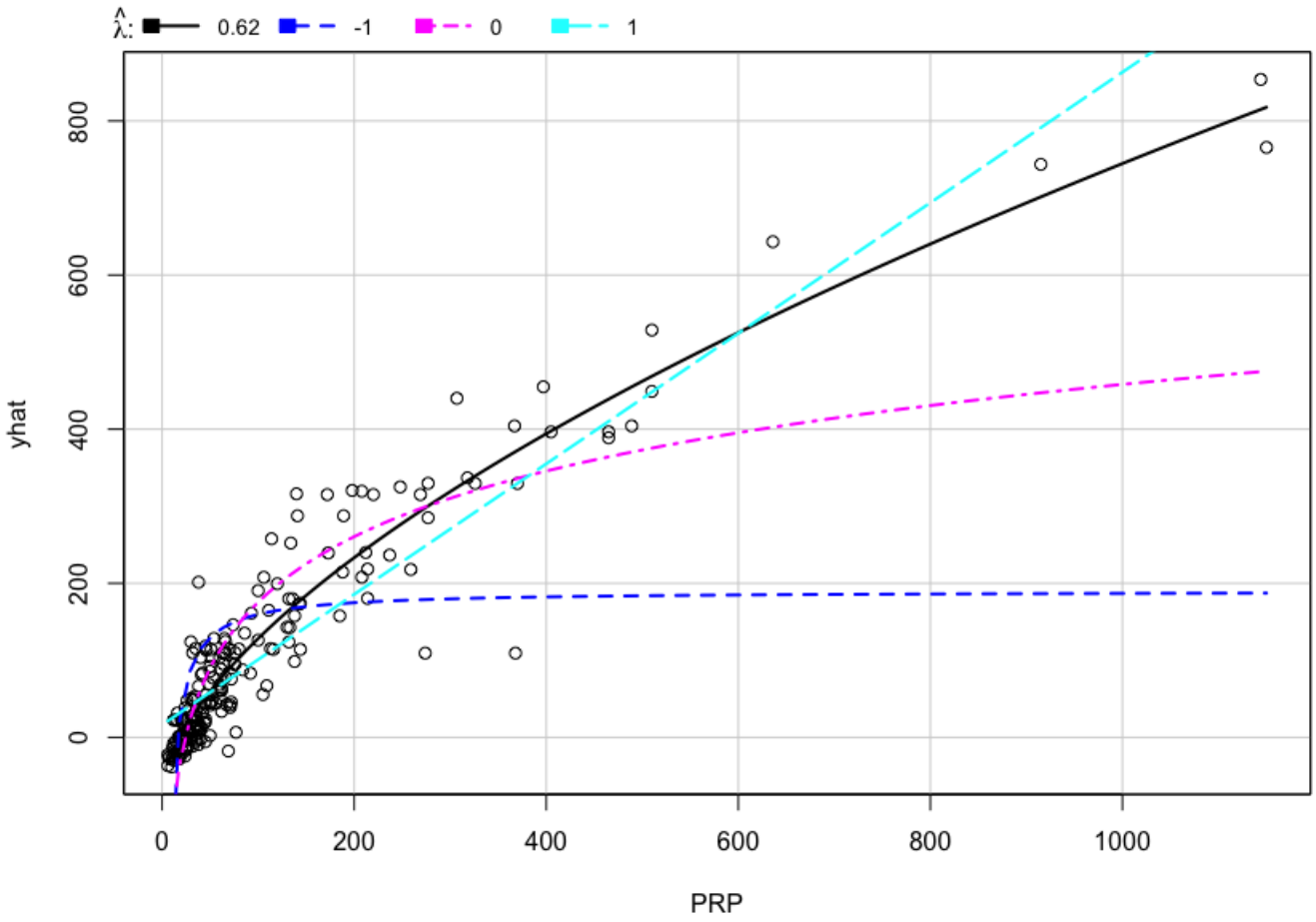

Figure 3: Inverse Response Plot of Model 1

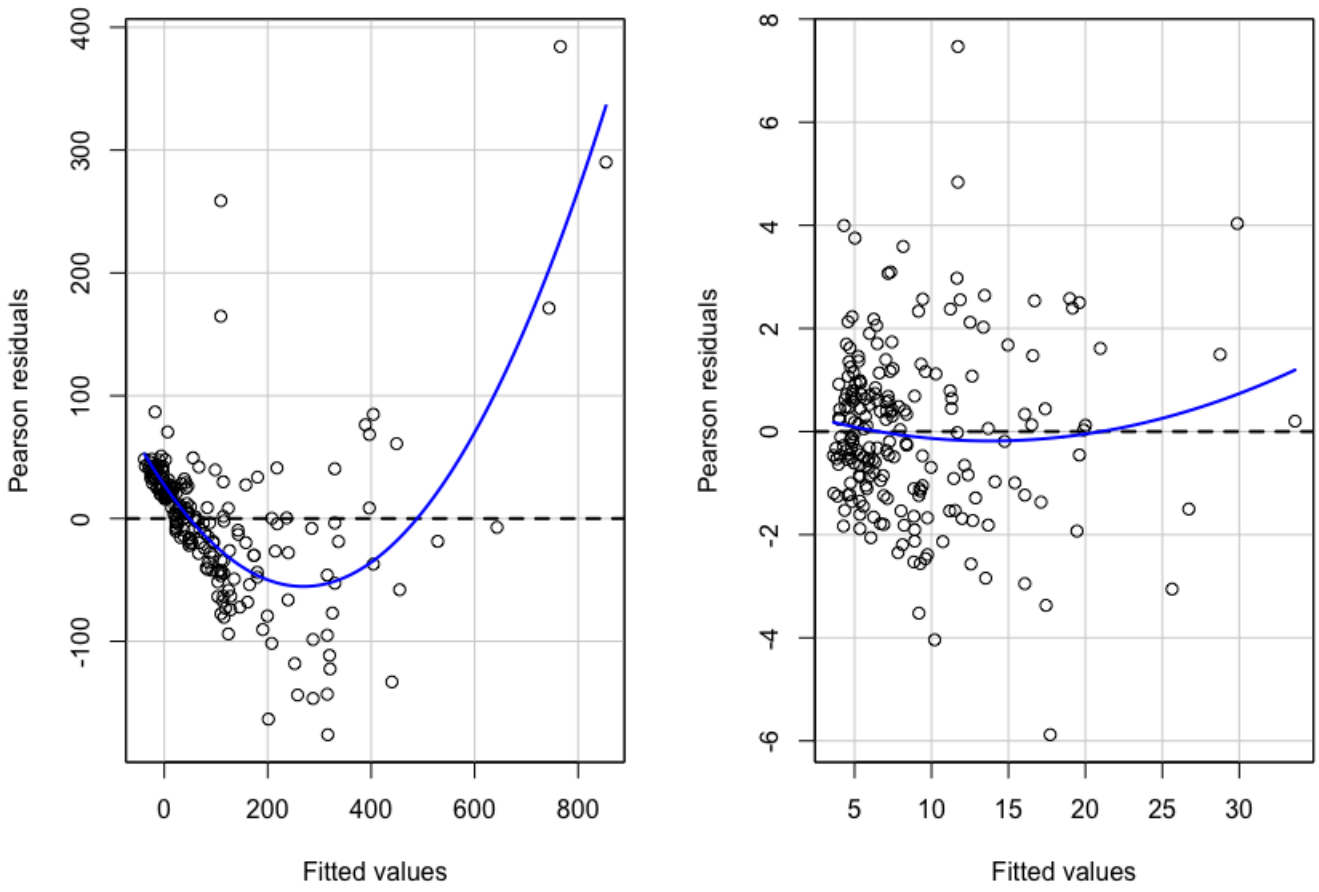

Figure 4: Residual plots of model 1 (left) and model 2 (right)

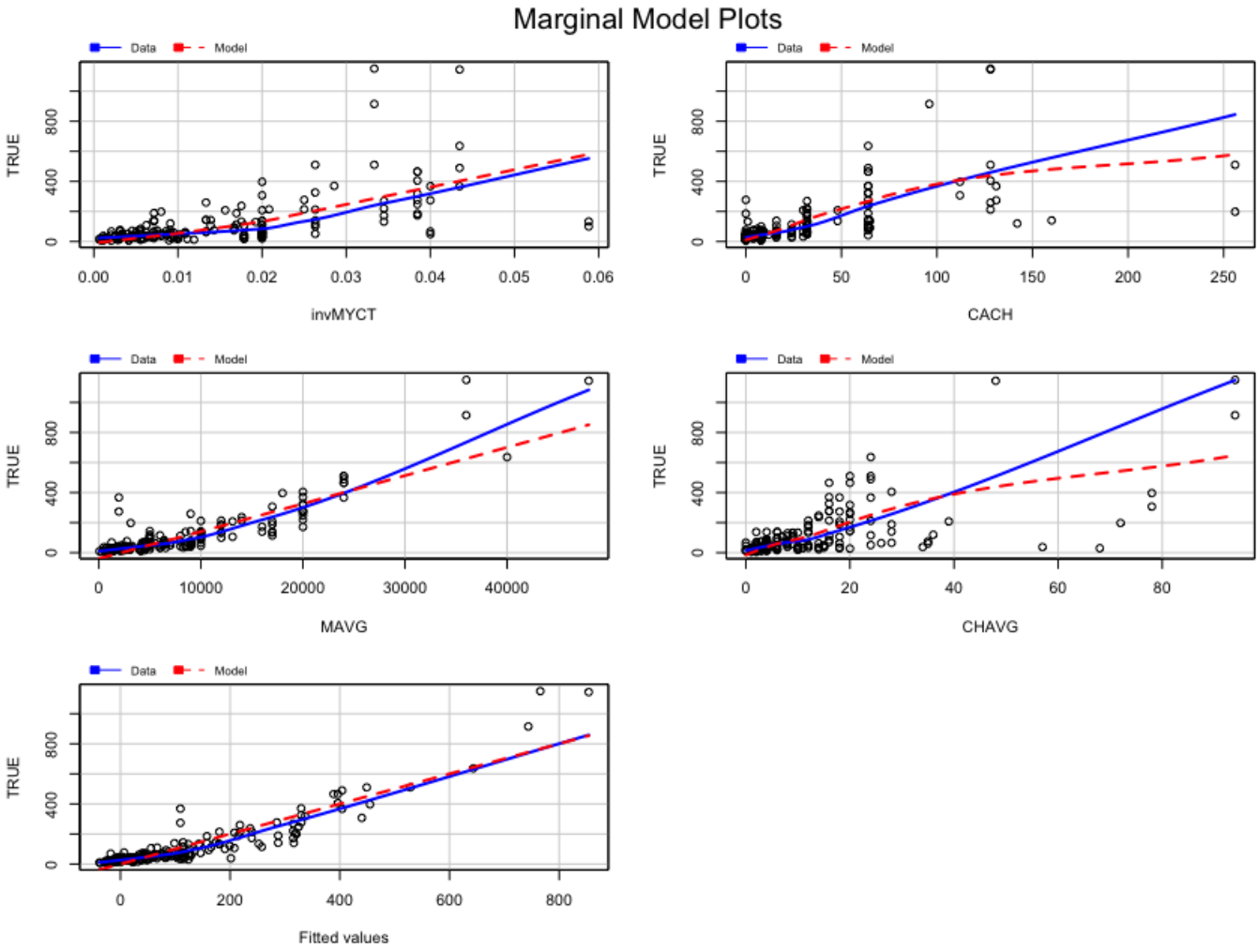

Figure 5: Marginal model plots for model 1

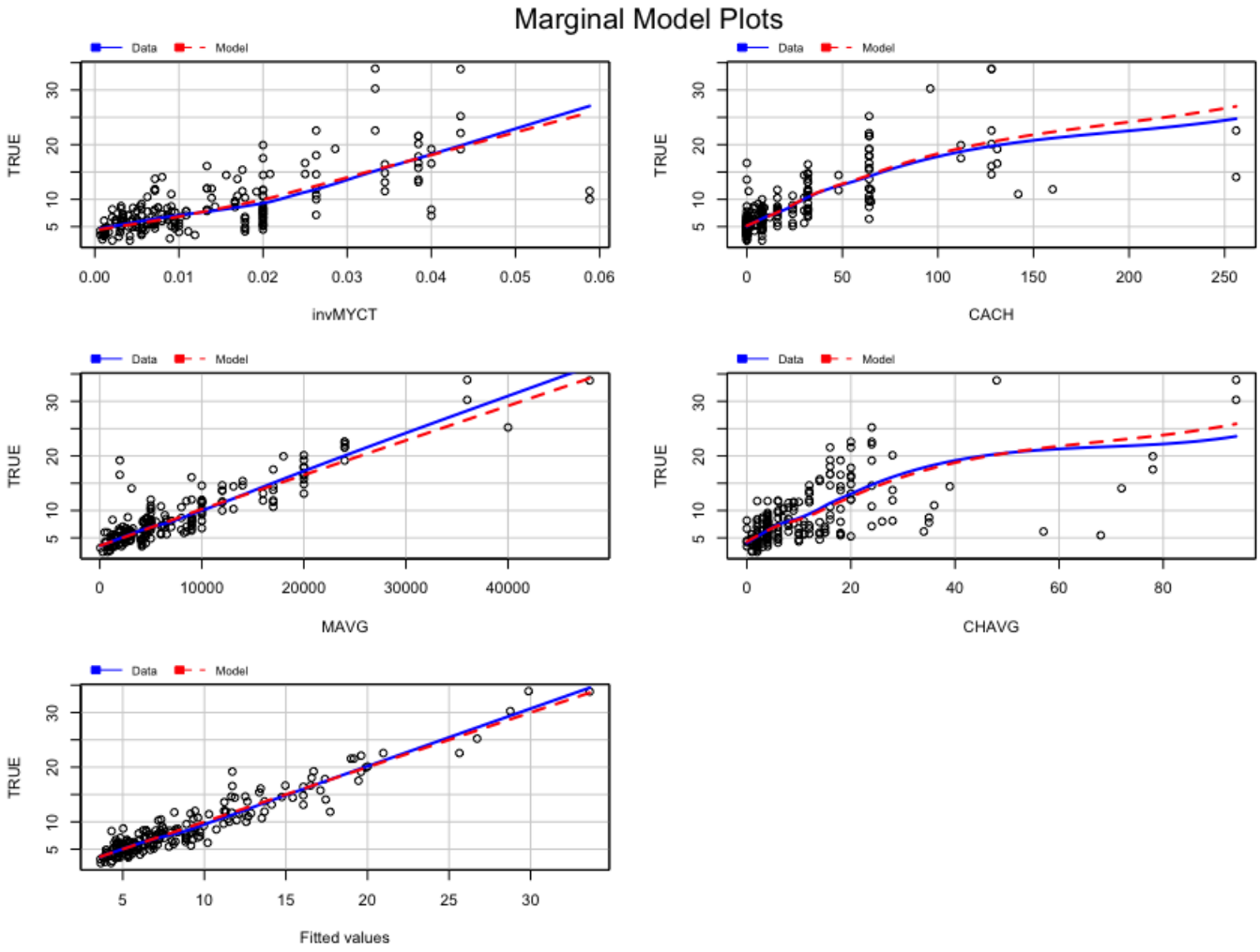

Figure 6: Marginal model plots for model 2

	Test stat	Pr(>|Test stat|)	
invMYCT	1.2438	0.215024	
CACH	-3.5284	0.000517	***
MAVG	1.8259	0.069341	.
CHAVG	-0.0268	0.978612	
Tukey test	1.4641	0.143155	

Figure 7: Tests for curvature of model 2

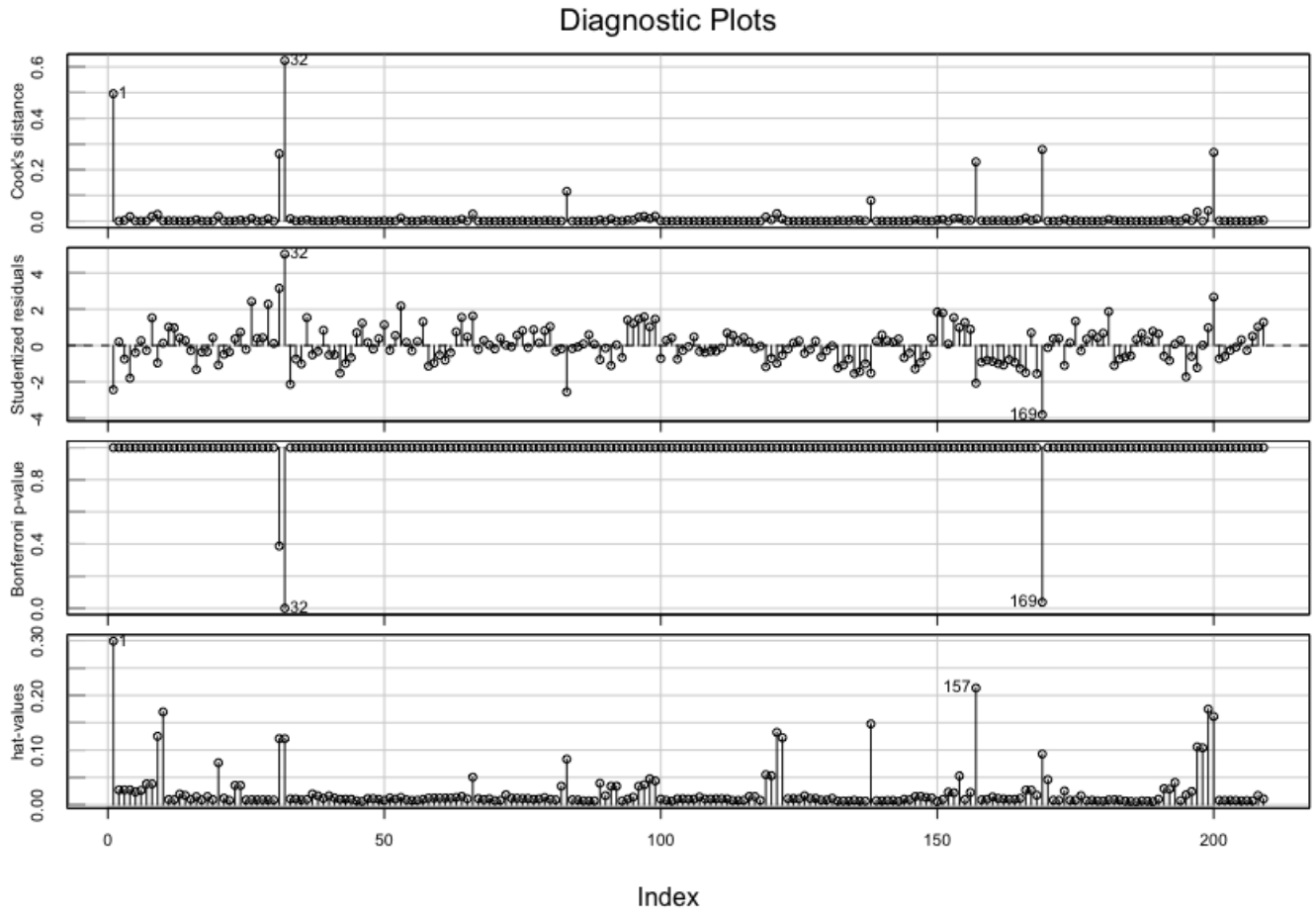

Figure 8: Influence index plots

```
                k.predictors    k.adjr     k.aic
15 invMYCT CACH MAVG CHAVG 0.9054292  814.7196
11          CACH MAVG CHAVG 0.8993607  826.7402
12       invMYCT CACH MAVG 0.8959805  833.6447
5                 CACH MAVG 0.8923707  839.7916
13      invMYCT MAVG CHAVG 0.8659703  886.6225
6                 MAVG CHAVG 0.8509871  907.7877
7              invMYCT MAVG 0.8313997  933.5987
1                     MAVG 0.8211462  944.9497
14      invMYCT CACH CHAVG 0.7618145 1006.7952
8               invMYCT CACH 0.6954100 1057.2088
9             invMYCT CHAVG 0.6920945 1059.4715
10               CACH CHAVG 0.6127299 1107.4013
2                     CACH 0.5327541 1145.6497
3                  invMYCT 0.5001364 1159.7529
4                    CHAVG 0.4078923 1195.1478
```

Figure 9: All-possible regressions test

```
lm(formula = sqrt(PRP) ~ invMYCT + CACH + MAVG + CHAVG, data = cpu)

Residuals:
    Min      1Q  Median      3Q     Max
-5.8803 -1.1084 -0.1063  0.8502  7.4677

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 3.359e+00  1.818e-01  18.481  < 2e-16 ***
invMYCT     5.159e+01  1.371e+01   3.762  0.00022 ***
CACH        3.507e-02  3.770e-03   9.302  < 2e-16 ***
MAVG        4.443e-04  2.514e-05  17.672  < 2e-16 ***
CHAVG       4.588e-02  9.899e-03   4.635 6.37e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.671 on 204 degrees of freedom
Multiple R-squared:  0.9072,    Adjusted R-squared:  0.9054
F-statistic: 498.9 on 4 and 204 DF,  p-value: < 2.2e-16
```

Figure 10: Regression model outputs

```
                    2.5 %       97.5 %
(Intercept) 3.001043e+00 3.717837e+00
invMYCT     2.455234e+01 7.862239e+01
CACH        2.763374e-02 4.249839e-02
MAVG        3.947745e-04 4.939245e-04
CHAVG       2.636418e-02 6.540094e-02
```

Figure 11: 95% confidence interval for parameter estimates

```
Analysis of Variance Table

Response: sqrt(PRP)
           Df  Sum Sq Mean Sq  F value    Pr(>F)
invMYCT     1 3087.03 3087.03 1105.291 < 2.2e-16 ***
CACH        1 1202.77 1202.77  430.643 < 2.2e-16 ***
MAVG        1 1223.30 1223.30  437.995 < 2.2e-16 ***
CHAVG       1   60.00   60.00   21.482 6.366e-06 ***
Residuals 204  569.76    2.79
```

Figure 12: ANOVA table for regression model

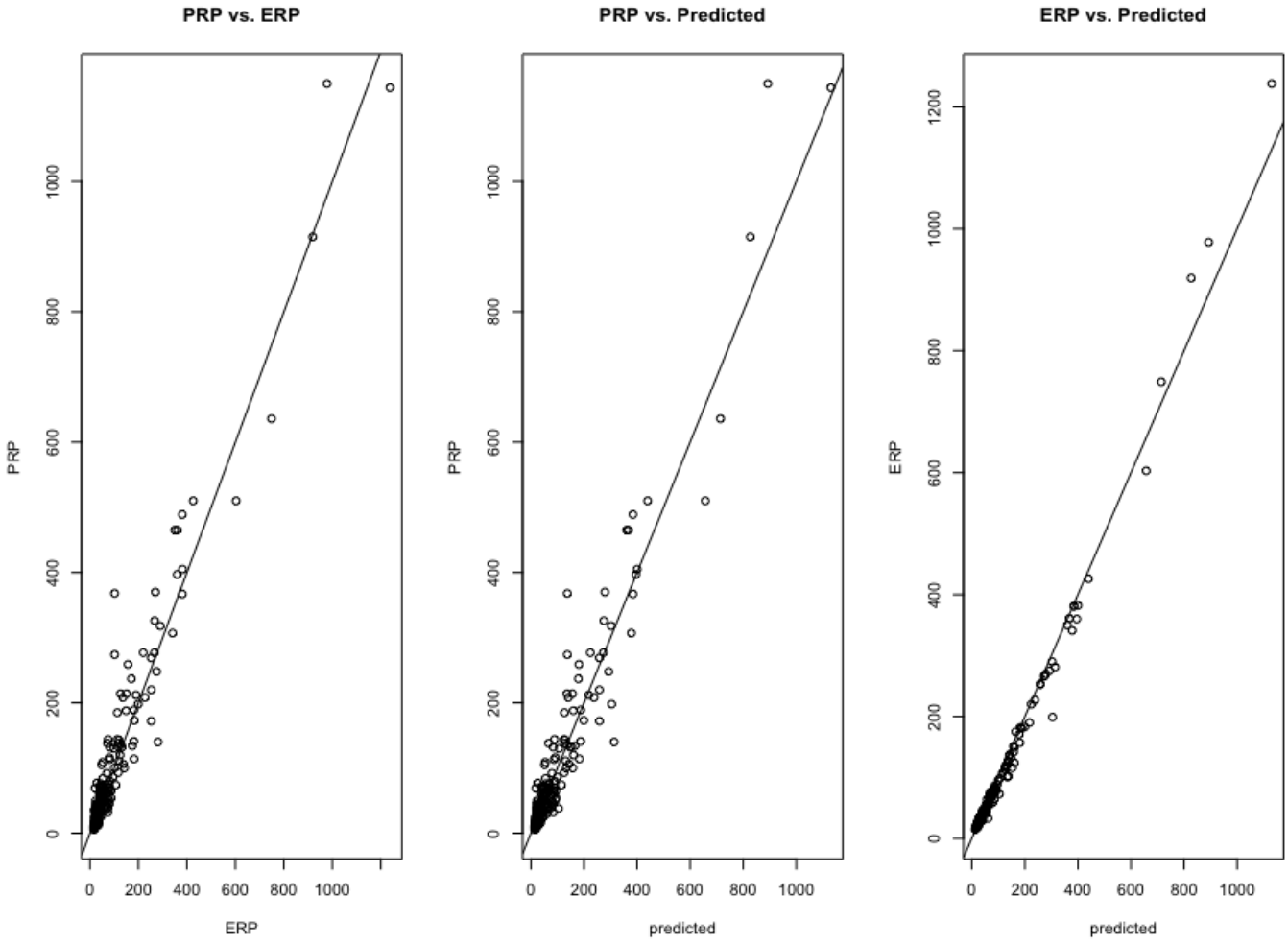

Figure 13: A comparison between the effectiveness of

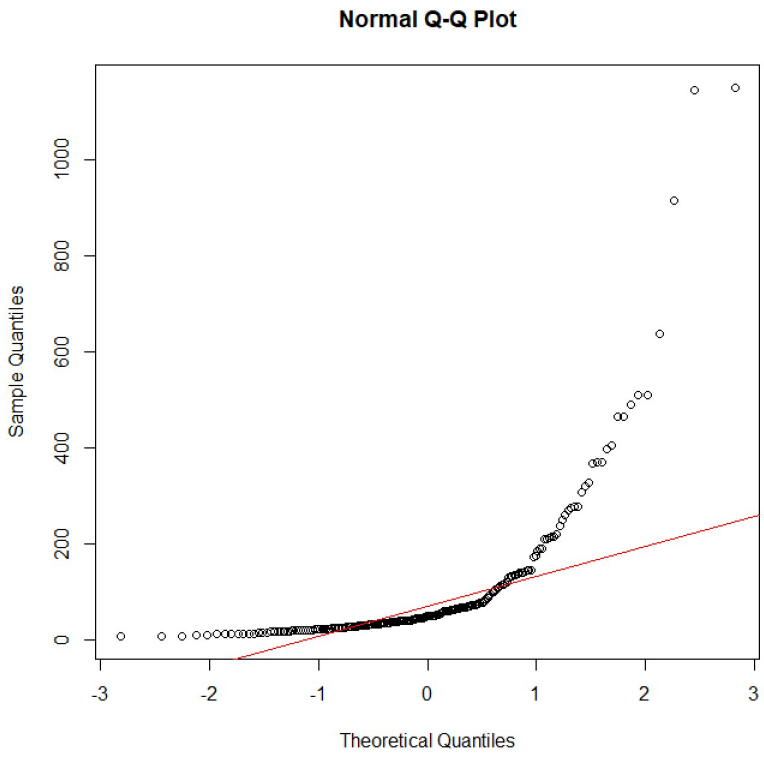

Figure 14: Normal Q-Q plot to assess normality of response, PRP

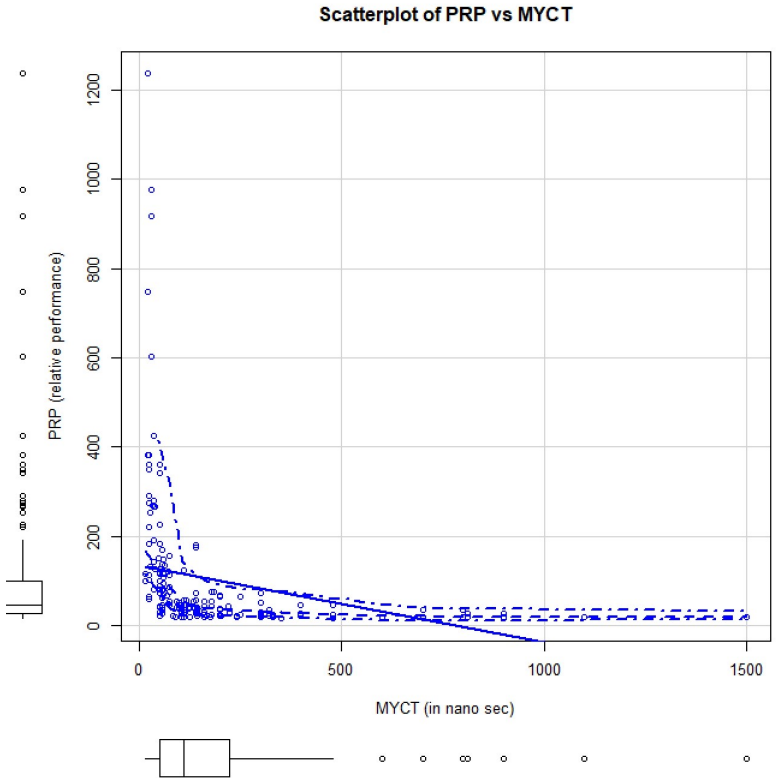

Figure 15: Scatterplot of PRP and MYCT showing hyperbolic nature