CPU Performance

About

Title

Relative CPU Performance Data, described in terms of its cycle time, memory size, etc.

Data Source

- Feldmesser,Jacob. (1987). Computer Hardware. UCI Machine Learning Repository.
- https://archive-beta.ics.uci.edu/dataset/29/computer+hardware

Relevant Information

The estimated relative performance values were estimated by the authors using a linear regression method.

Dataset

Load CPU Perfomance dataset

Data Dictionary

Column	Description
vendor name	30 different vendor
Model Name	many unique symbols
MYCT	machine cycle time in nanoseconds (integer)
MMIN	minimum main memory in kilobytes (integer)
MMAX	maximum main memory in kilobytes (integer)
CACH	cache memory in kilobytes (integer)

Column	Description
CHMIN	minimum channels in units (integer)
CHMAX	maximum channels in units (integer)
PRP	published relative performance (integer)
ERP	estimated relative performance from the original article (integer)

Class Distribution: the class value (PRP) is continuously valued.

PRP Value Range	Number of Instances in Range
0-20	31
21-100	121
101-200	27
201-300	13
301-400	7
401-500	4
501-600	2
above 600	4

```
   Vendor   Model MYCT MMIN  MMAX CACH CHMIN CHMAX PRP ERP
1 adviser   32/60  125  256  6000  256    16   128 198 199
2  amdahl  470v/7   29 8000 32000   32     8    32 269 253
3  amdahl 470v/7a   29 8000 32000   32     8    32 220 253
4  amdahl 470v/7b   29 8000 32000   32     8    32 172 253
5  amdahl 470v/7c   29 8000 16000   32     8    16 132 132
6  amdahl  470v/b   26 8000 32000   64     8    32 318 290
```

Summary Statistics

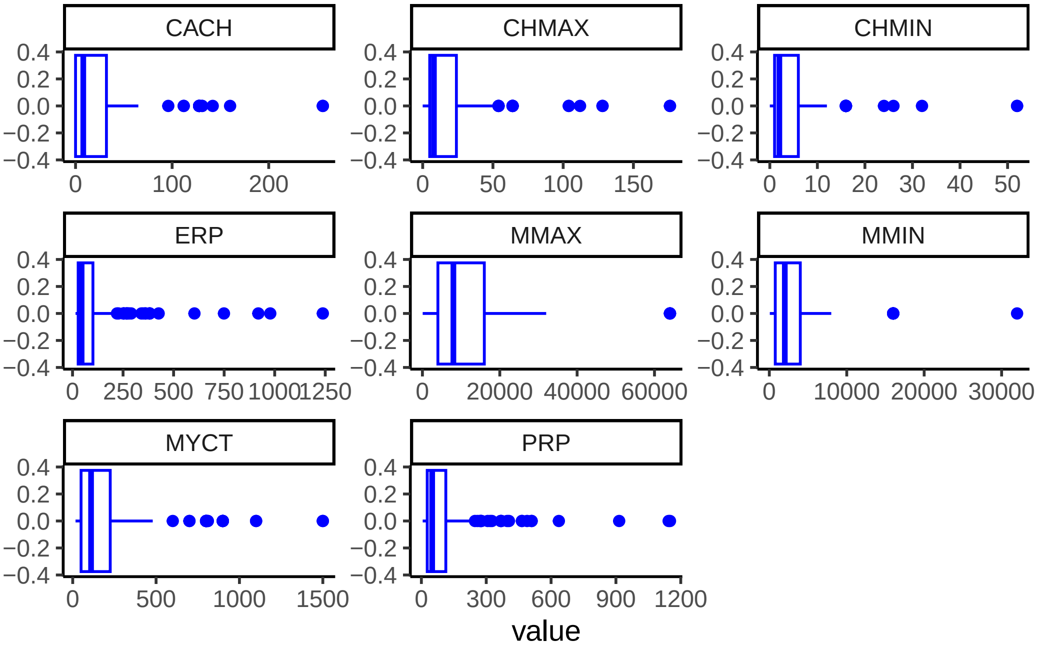
```
    Vendor              Model                MYCT             MMIN      
 Length:209         Length:209         Min.   :  17.0   Min.   :   64  
 Class :character   Class :character   1st Qu.:  50.0   1st Qu.:  768  
 Mode  :character   Mode  :character   Median : 110.0   Median : 2000  
                                       Mean   : 203.8   Mean   : 2868  
                                       3rd Qu.: 225.0   3rd Qu.: 4000  
                                       Max.   :1500.0   Max.   :32000  
      MMAX            CACH           CHMIN            CHMAX       
 Min.   :   64   Min.   :  0.00   Min.   : 0.000   Min.   :  0.00  
 1st Qu.: 4000   1st Qu.:  0.00   1st Qu.: 1.000   1st Qu.:  5.00  
 Median : 8000   Median :  8.00   Median : 2.000   Median :  8.00  
 Mean   :11796   Mean   : 25.21   Mean   : 4.699   Mean   : 18.27  
 3rd Qu.:16000   3rd Qu.: 32.00   3rd Qu.: 6.000   3rd Qu.: 24.00  
 Max.   :64000   Max.   :256.00   Max.   :52.000   Max.   :176.00  
      PRP              ERP         
 Min.   :   6.0   Min.   :  15.00  
 1st Qu.:  27.0   1st Qu.:  28.00  
 Median :  50.0   Median :  45.00  
 Mean   : 105.6   Mean   :  99.33  
 3rd Qu.: 113.0   3rd Qu.: 101.00  
 Max.   :1150.0   Max.   :1238.00  
```

Glimpse of Data

```
Rows: 209
Columns: 10
$ Vendor <chr> "adviser", "amdahl", "amdahl", "amdahl", "amdahl", "amdahl", "a~
$ Model  <chr> "32/60", "470v/7", "470v/7a", "470v/7b", "470v/7c", "470v/b", "~
$ MYCT   <int> 125, 29, 29, 29, 29, 26, 23, 23, 23, 23, 400, 400, 60, 50, 350,~
$ MMIN   <int> 256, 8000, 8000, 8000, 8000, 8000, 16000, 16000, 16000, 32000, ~
$ MMAX   <int> 6000, 32000, 32000, 32000, 16000, 32000, 32000, 32000, 64000, 6~
$ CACH   <int> 256, 32, 32, 32, 32, 64, 64, 64, 64, 128, 0, 4, 65, 65, 0, 0, 8~
$ CHMIN  <int> 16, 8, 8, 8, 8, 8, 16, 16, 16, 32, 1, 1, 1, 1, 1, 4, 4, 7, 5, 8~
$ CHMAX  <int> 128, 32, 32, 32, 16, 32, 32, 32, 32, 64, 2, 6, 8, 8, 4, 32, 15,~
$ PRP    <int> 198, 269, 220, 172, 132, 318, 367, 489, 636, 1144, 38, 40, 92, ~
$ ERP    <int> 199, 253, 253, 253, 132, 290, 381, 381, 749, 1238, 23, 24, 70, ~
```

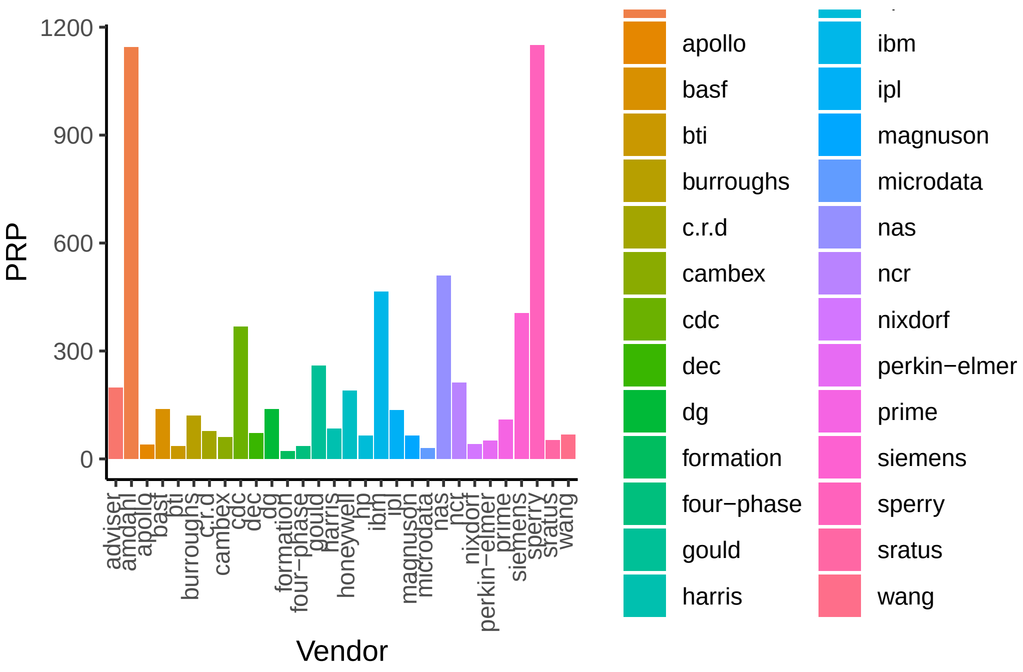
Visual Analysis

Histograms



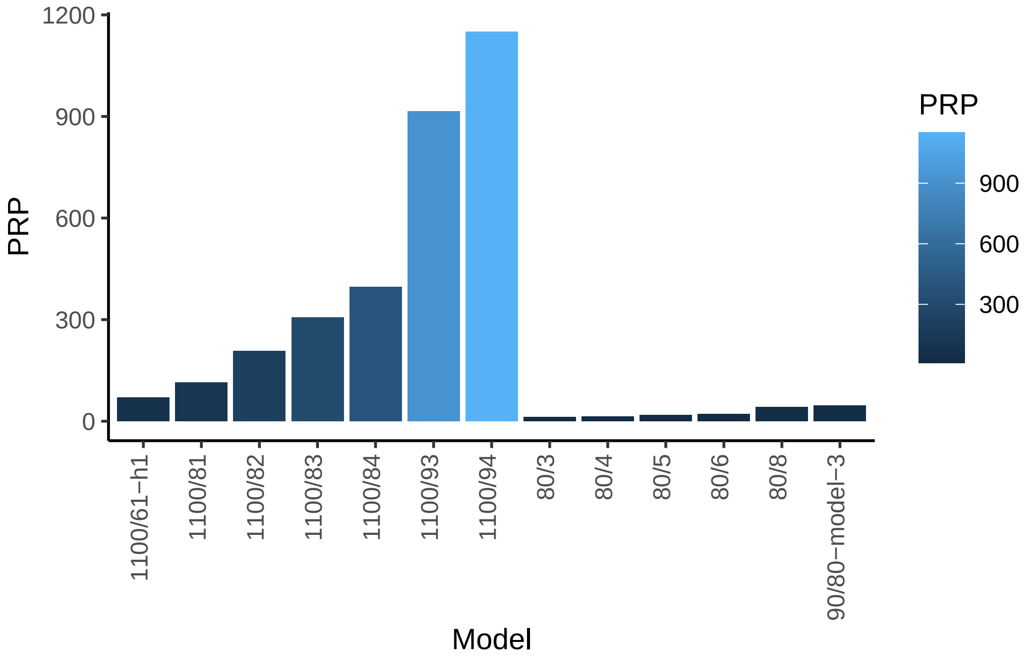
All of the features have some outliers.

Performance per Vendor



Amdahl and Sperry have the highest performance

Performance per Model of Sperry



The better performing Sperry is model #1100/94

Correlations

Drop Vendor, Model, and ERP from dataset

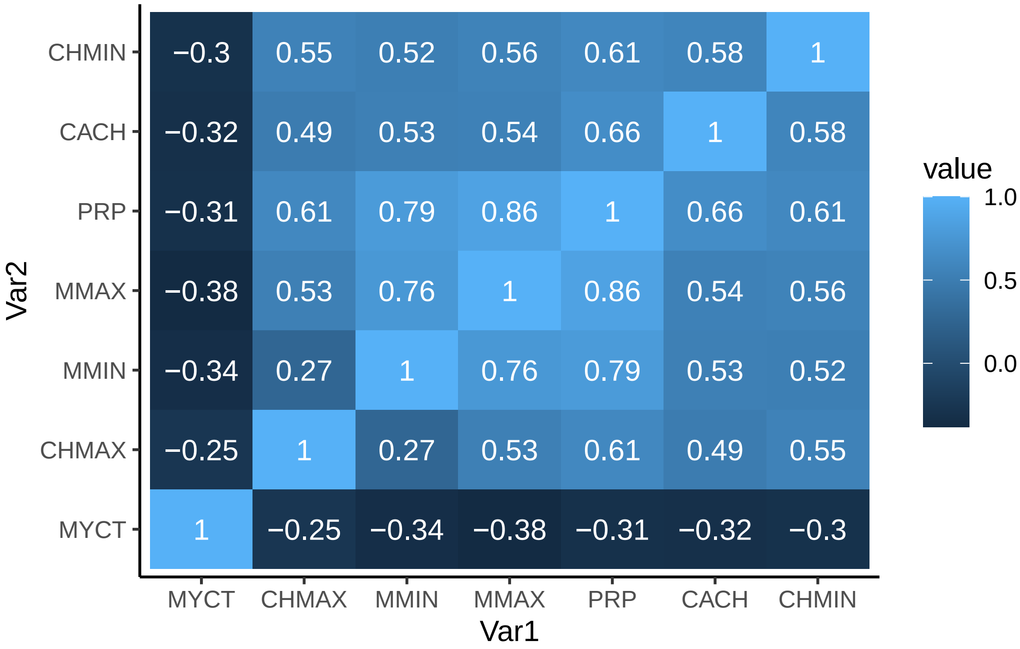
```
  MYCT MMIN  MMAX CACH CHMIN CHMAX PRP
1  125  256  6000  256    16   128 198
2   29 8000 32000   32     8    32 269
3   29 8000 32000   32     8    32 220
4   29 8000 32000   32     8    32 172
5   29 8000 16000   32     8    16 132
6   26 8000 32000   64     8    32 318
```

Run correlation

```
      MYCT  MMIN  MMAX  CACH CHMIN CHMAX   PRP
MYCT  1.00 -0.34 -0.38 -0.32 -0.30 -0.25 -0.31
MMIN -0.34  1.00  0.76  0.53  0.52  0.27  0.79
MMAX -0.38  0.76  1.00  0.54  0.56  0.53  0.86
```

```
CACH  -0.32  0.53  0.54  1.00  0.58  0.49  0.66
CHMIN -0.30  0.52  0.56  0.58  1.00  0.55  0.61
CHMAX -0.25  0.27  0.53  0.49  0.55  1.00  0.61
```

Correlation heatmap

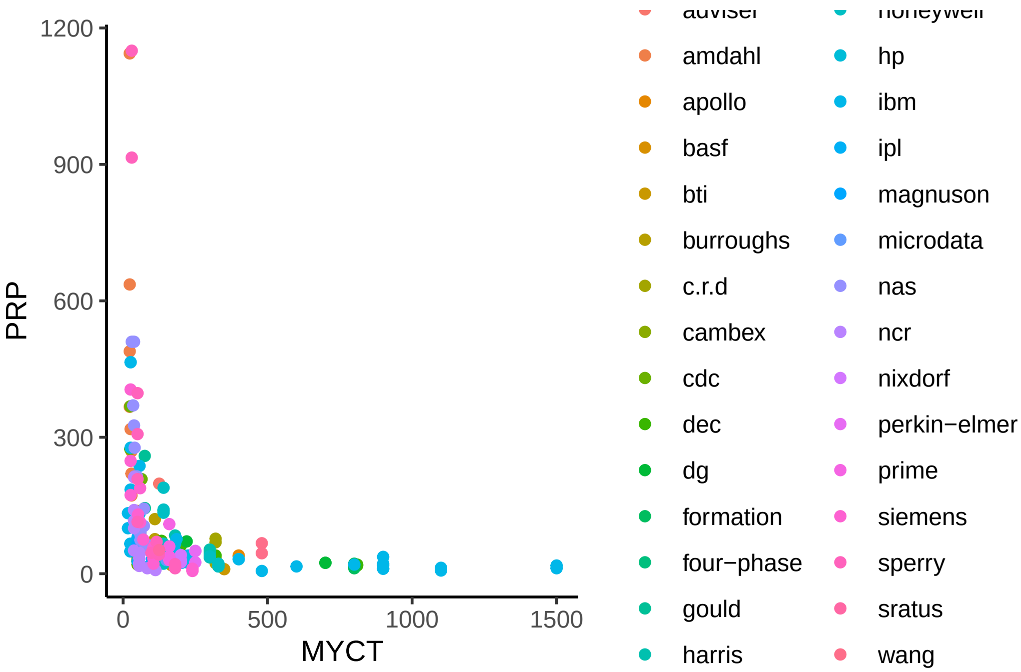


From the correlation matrix we can see:

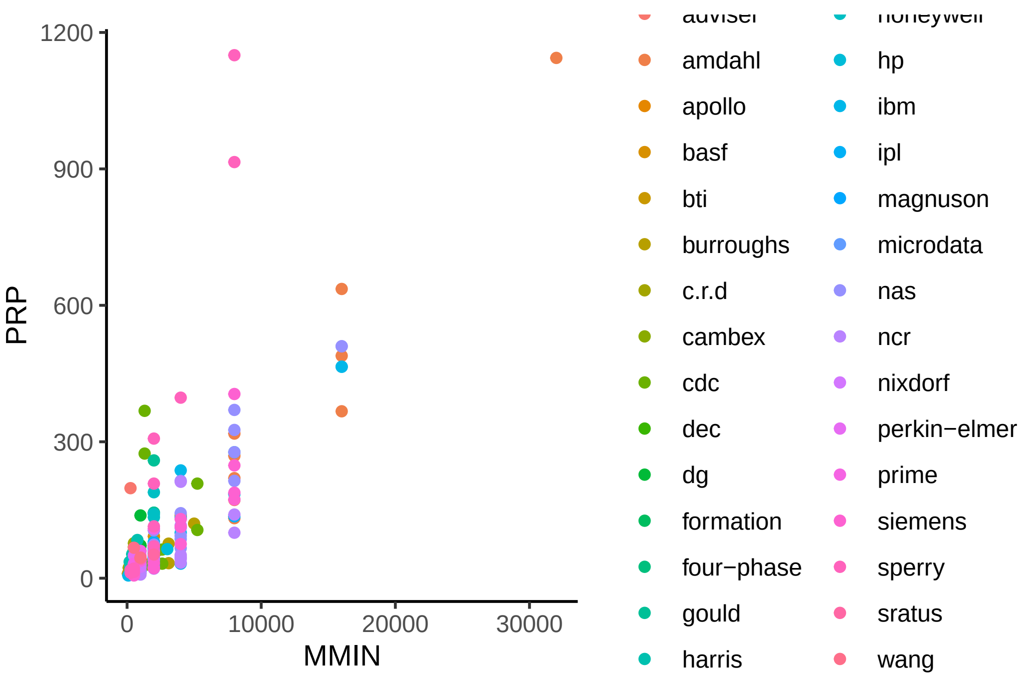
- PRP and MMAX are highly correlated
- PRP and CACH are highly correlated
- PRP and CHMAX are highly correlated
- PRP and MMIN are highly correlated
- PRP and CHMIN are highly correlated

Scatterplot of PRP vs features

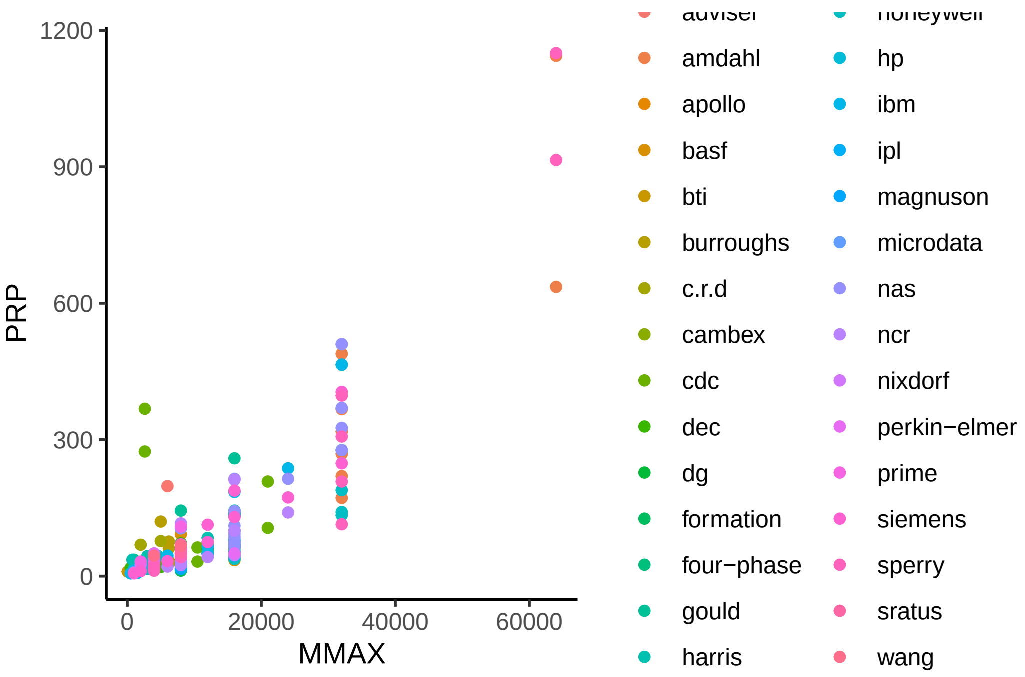
PRP vs MYCT



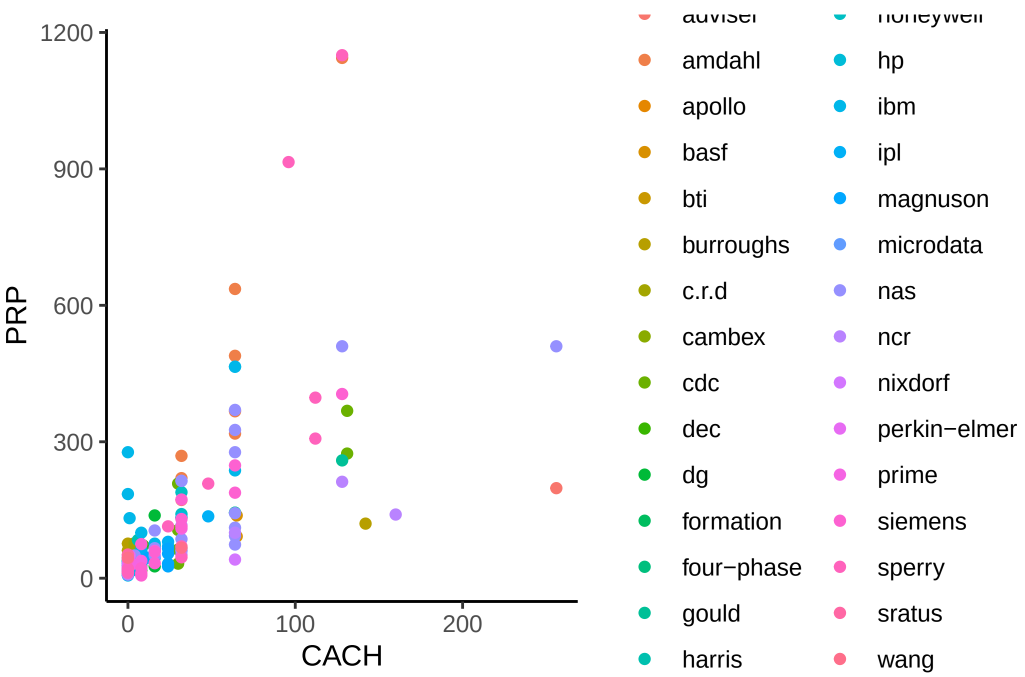
PRP vs MYCT



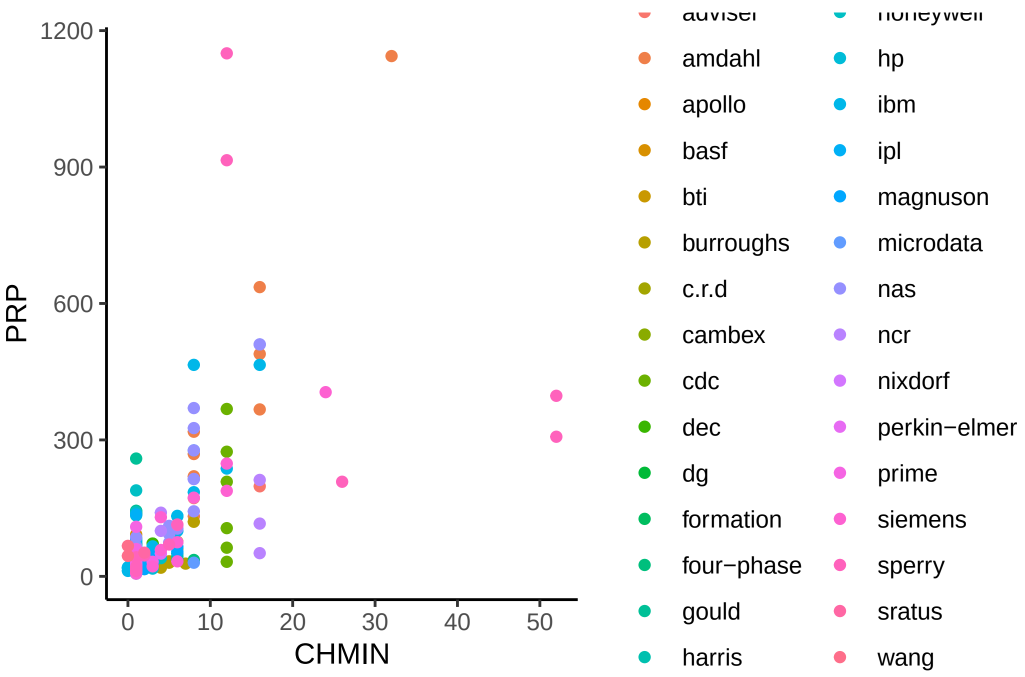
PRP vs MMAX



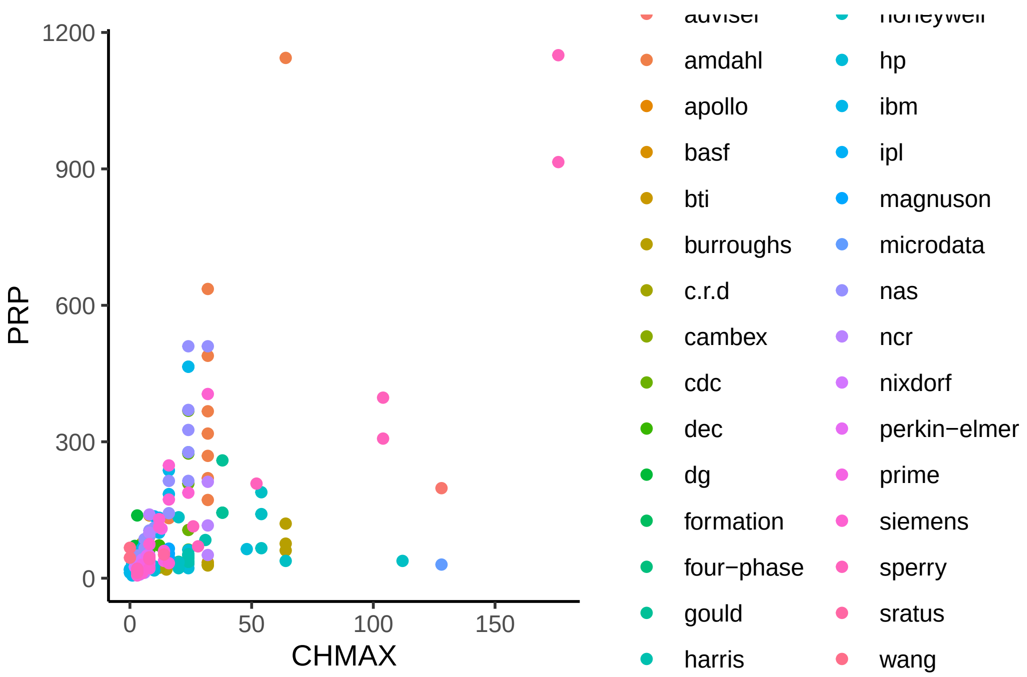
PRP vs CACH



PRP vs CHMIN



PRP vs CHMAX



Model

Split data into training and test datasets

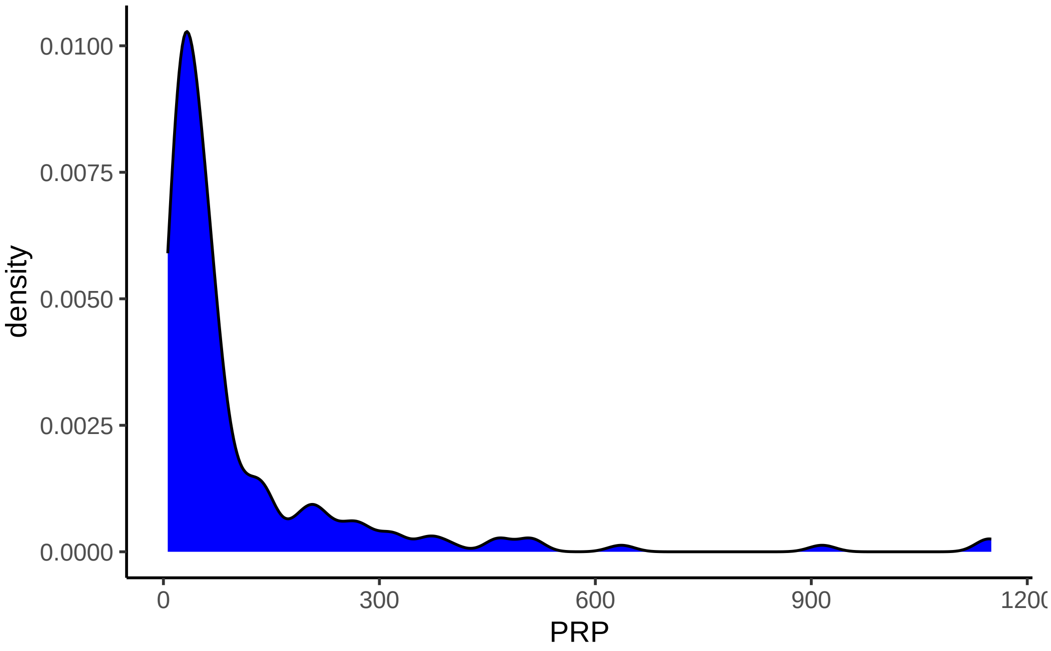
```
train:  167 7
```

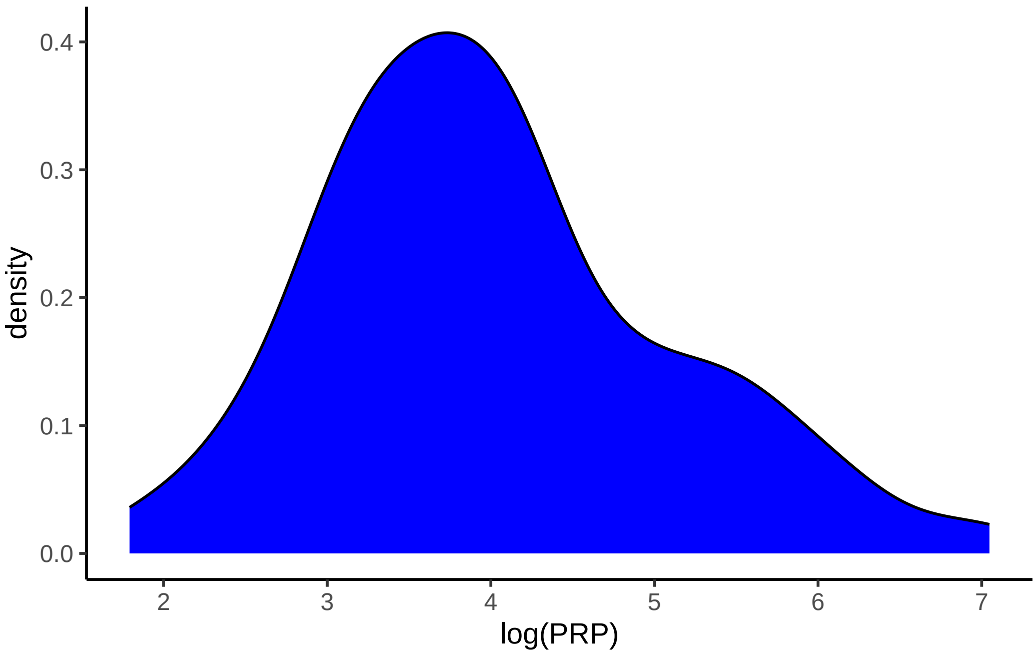
```
test:  42 7
```

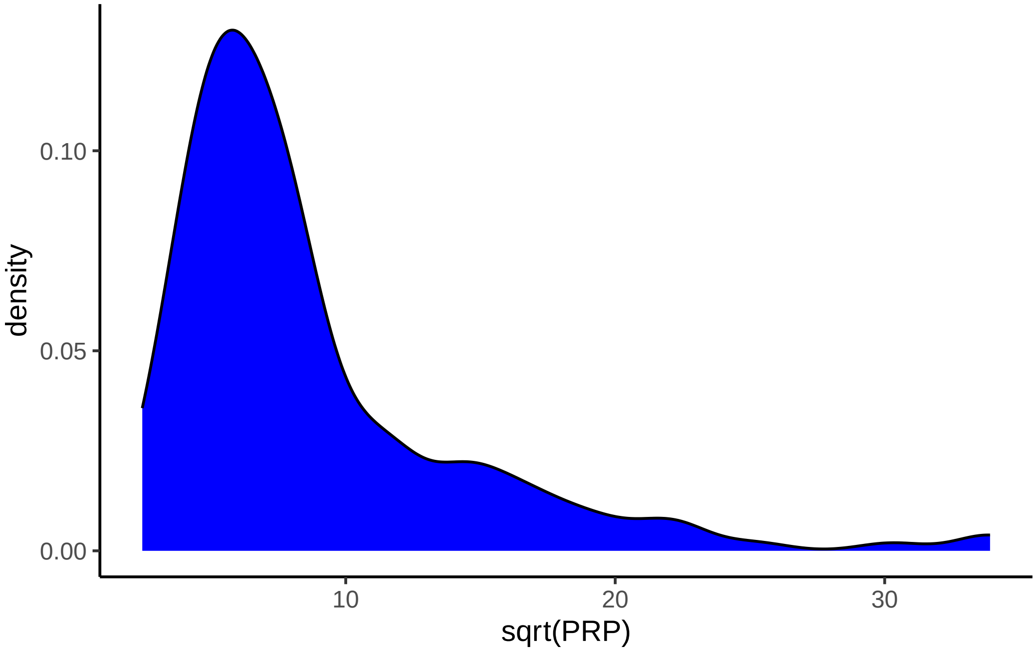
Training dataset

```
  MYCT  MMIN  MMAX CACH CHMIN CHMAX PRP
1  125   256  6000  256    16   128 198
2   29  8000 32000   32     8    32 269
3   29  8000 32000   32     8    32 220
6   26  8000 32000   64     8    32 318
7   23 16000 32000   64    16    32 367
9   23 16000 64000   64    16    32 636
```

Check distribution of PRP response variable







The log transformation of the PRP response variable is closer to normal so we will use that

Log PRP

```
  MYCT  MMIN  MMAX CACH CHMIN CHMAX      PRP
1  125   256  6000  256    16   128 5.288267
2   29  8000 32000   32     8    32 5.594711
3   29  8000 32000   32     8    32 5.393628
6   26  8000 32000   64     8    32 5.762051
7   23 16000 32000   64    16    32 5.905362
9   23 16000 64000   64    16    32 6.455199
```

Regression model 1 - All features

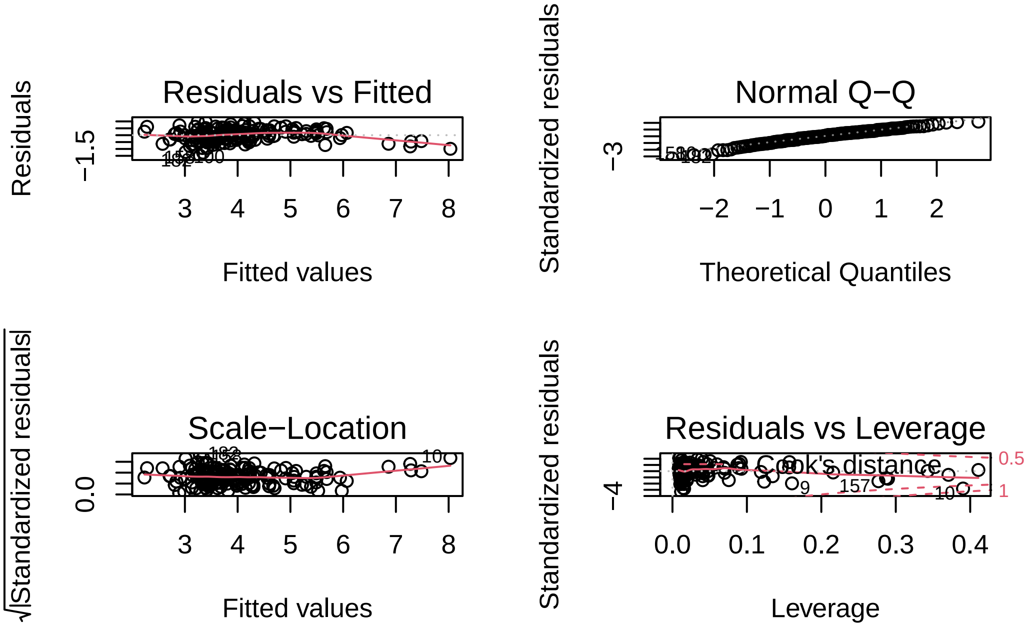
```

Call:
lm(formula = PRP ~ ., data = train_df)

Coefficients:
(Intercept)         MYCT         MMIN         MMAX         CACH        CHMIN
  3.361e+00   -7.937e-04    1.702e-05    4.948e-05    6.133e-03    6.244e-03
      CHMAX
```

```
 -1.009e-04
```

Summary Statistics

```

Call:
lm(formula = PRP ~ ., data = train_df)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.49193 -0.25878  0.04092  0.30446  0.98896 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.361e+00  6.718e-02  50.029  < 2e-16 ***
MYCT        -7.937e-04  1.445e-04  -5.491 1.54e-07 ***
MMIN         1.702e-05  1.560e-05   1.091    0.277    
MMAX         4.948e-05  5.661e-06   8.741 3.00e-15 ***
CACH         6.133e-03  1.200e-03   5.111 9.04e-07 ***
CHMIN        6.244e-03  6.740e-03   0.926    0.356    
CHMAX       -1.009e-04  1.791e-03  -0.056    0.955    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.454 on 160 degrees of freedom
Multiple R-squared:  0.8261,	Adjusted R-squared:  0.8196 
F-statistic: 126.7 on 6 and 160 DF,  p-value: < 2.2e-16
```

Visualize model



The adjusted R-squared is .8196, meaning the independent variables explain 82% of the variance of the CPU performance.

Three variables (MYCT, MMAX, CACH) show very low p-values (less than 0.05) and are significant

The residuals vs fitted plot show the trend line close to zero except after around 5.5

The Q_Q plot shows us that the features are normal except for the ends

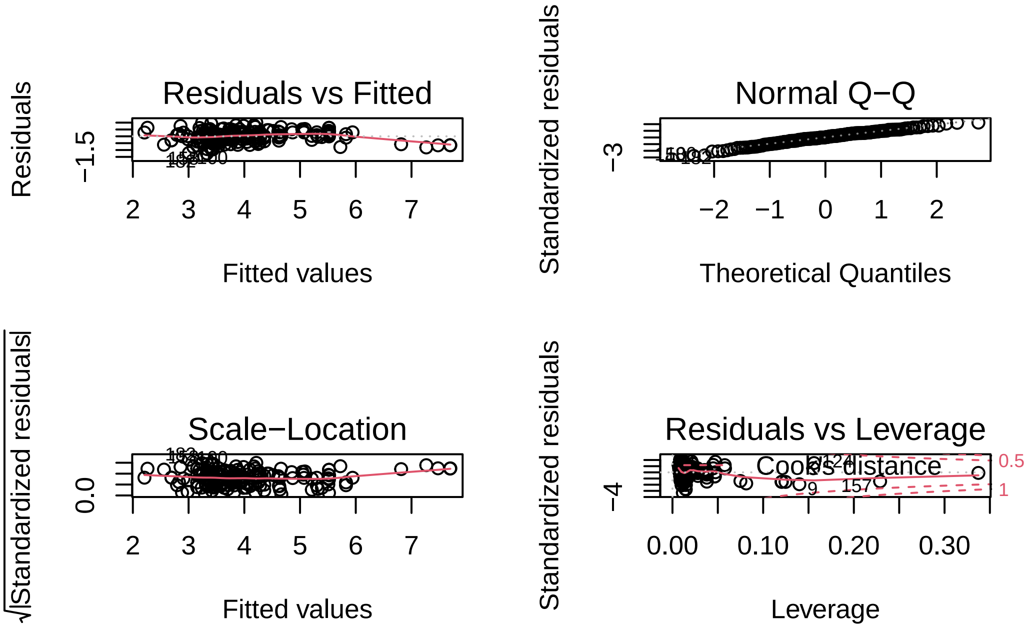
Regression Model 2 - features MYCT, MMAX, CACH only

```
Call:
lm(formula = PRP ~ MYCT + MMAX + CACH, data = train_df)

Coefficients:
(Intercept)         MYCT         MMAX         CACH  
  3.365e+00   -8.074e-04    5.447e-05    6.761e-03  
```

Summary Statistics

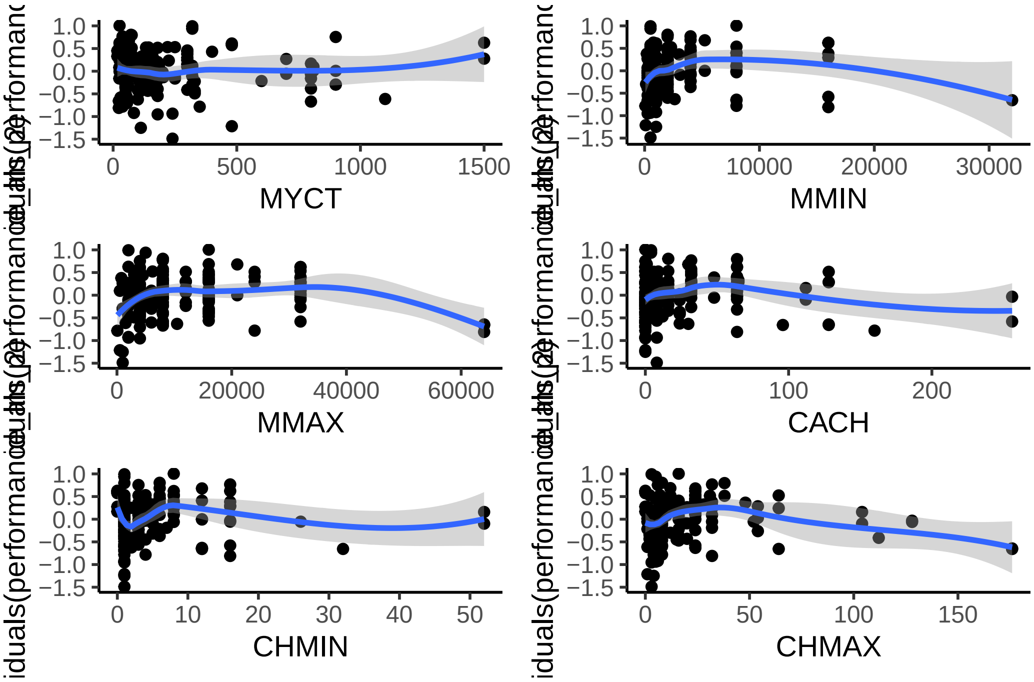
```
Call:
lm(formula = PRP ~ MYCT + MMAX + CACH, data = train_df)

Residuals:
     Min       1Q   Median       3Q      Max 
-1.48775 -0.27856  0.01263  0.29954  1.00502 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  3.365e+00  6.671e-02  50.436  < 2e-16 ***
MYCT        -8.074e-04  1.441e-04  -5.605 8.73e-08 ***
MMAX         5.448e-05  3.695e-06  14.741  < 2e-16 ***
CACH         6.761e-03  1.074e-03   6.293 2.76e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4533 on 163 degrees of freedom
Multiple R-squared:  0.8234,    Adjusted R-squared:  0.8201 
F-statistic: 253.3 on 3 and 163 DF,  p-value: < 2.2e-16
```

Visualize model



The F-statistic is much higher than in model 1 and all features are significant. The R2 is a little higher than in model 1.

Check predictor vs residual plot



ANOVA Test - Model 2

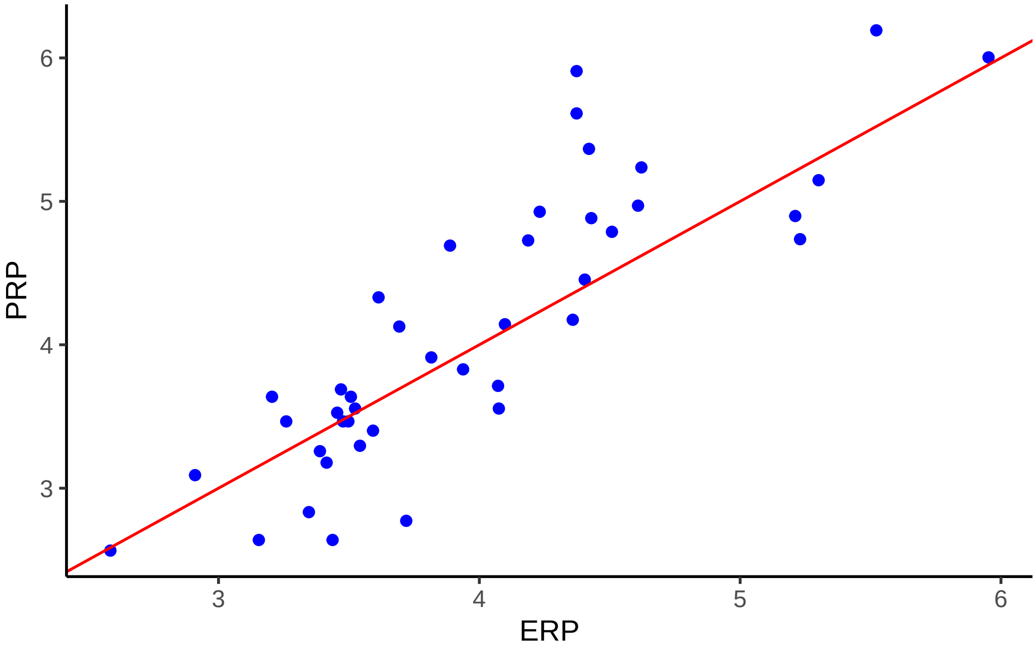
```
Analysis of Variance Table

Response: PRP
           Df Sum Sq Mean Sq F value    Pr(>F)    
MYCT        1 52.791  52.791 256.913 < 2.2e-16 ***
MMAX        1 95.199  95.199 463.291 < 2.2e-16 ***
CACH        1  8.138   8.138  39.604  2.76e-09 ***
Residuals 163 33.494   0.205                      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Predict PRP with model 2

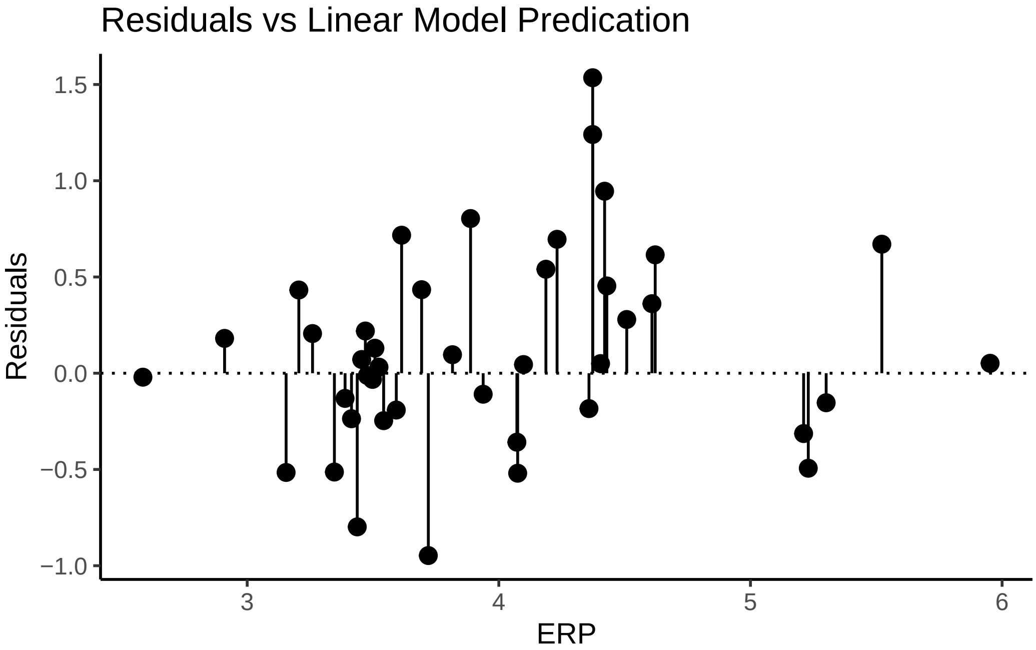
```
        ERP      PRP MYCT  MMIN  MMAX CACH CHMIN CHMAX
4  5.300864 5.147494   29  8000 32000   32     8    32
5  4.429265 4.882802   29  8000 16000   32     8    16
8  5.522059 6.192362   23 16000 32000   64    16    32
```

```
11 3.205190 3.637586  400  1000  3000    0     1     2
16 4.074847 3.555348  200   512 16000    0     4    32
20 4.508345 4.787492  110  5000  5000  142     8    64
```

Plot predicted PRP vs PRP

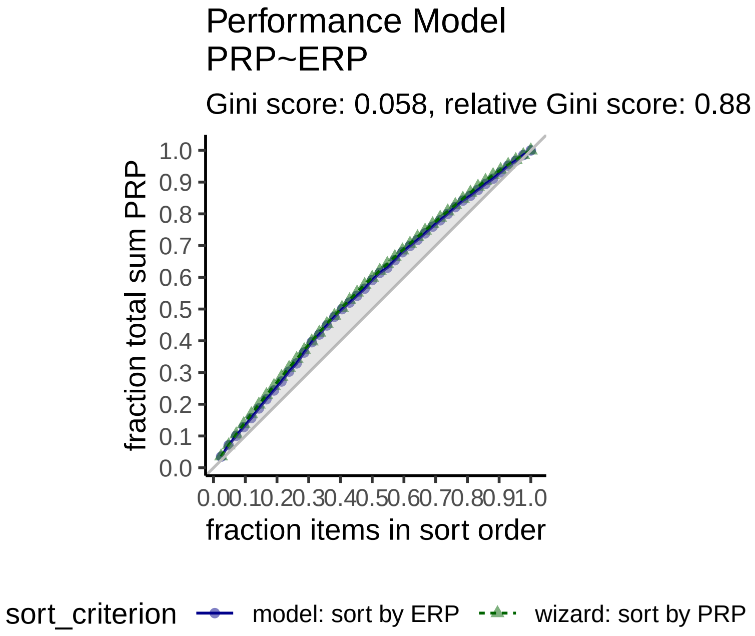


Residuals vs Prediction



The plot shows the prediction errors vary from the PRP

Gain Curve plot



The Gini score of 0.88 shows that the model correctly sorts high performance from lower ones.

Performance on Test data

```
RMSE:  0.5243939


Std Deviation:  0.9720138


r2:  0.7283943
```

The RMSE is lower than the Std deviation so the model predicts the PRP well. The R2 is 73% which shows that the model predicts pretty well

Cross Validation

Split data

```
List of 3
 $ :List of 2
  ..$ train: int [1:140] 2 3 4 5 7 9 10 11 12 13 ...
  ..$ app  : int [1:69] 57 161 74 25 85 170 189 145 104 93 ...
 $ :List of 2
  ..$ train: int [1:139] 1 2 4 6 8 10 13 16 17 18 ...
  ..$ app  : int [1:70] 192 148 66 133 136 45 159 105 173 184 ...
 $ :List of 2
  ..$ train: int [1:139] 1 3 5 6 7 8 9 11 12 14 ...
  ..$ app  : int [1:70] 165 163 4 117 55 146 134 176 53 63 ...
 - attr(*, "splitmethod")= chr "kwaycross"
```

Run Crossfold

```
RMSE on full model : 71.13728


RMSE of the cross-validation predictions:  81.09387
```