First Year Project: Extinction prediciton

2023-02-07

We start by importing the dataset directly from the csv file and saving it to the data variable:

```
head(data <- read.csv('Factors Affecting Extinction.csv', header=T))
```

```
##          Species Time Pairs Size Status
## 1    Sparrowhawk 3.03  1.00    L      R
## 2        Buzzard 5.46  2.00    L      R
## 3        Kestrel 4.10  1.21    L      R
## 4      Peregrine 1.68  1.13    L      R
## 5 Grey_partridge 8.85  5.17    L      R
## 6          Quail 1.49  1.00    L      M
```

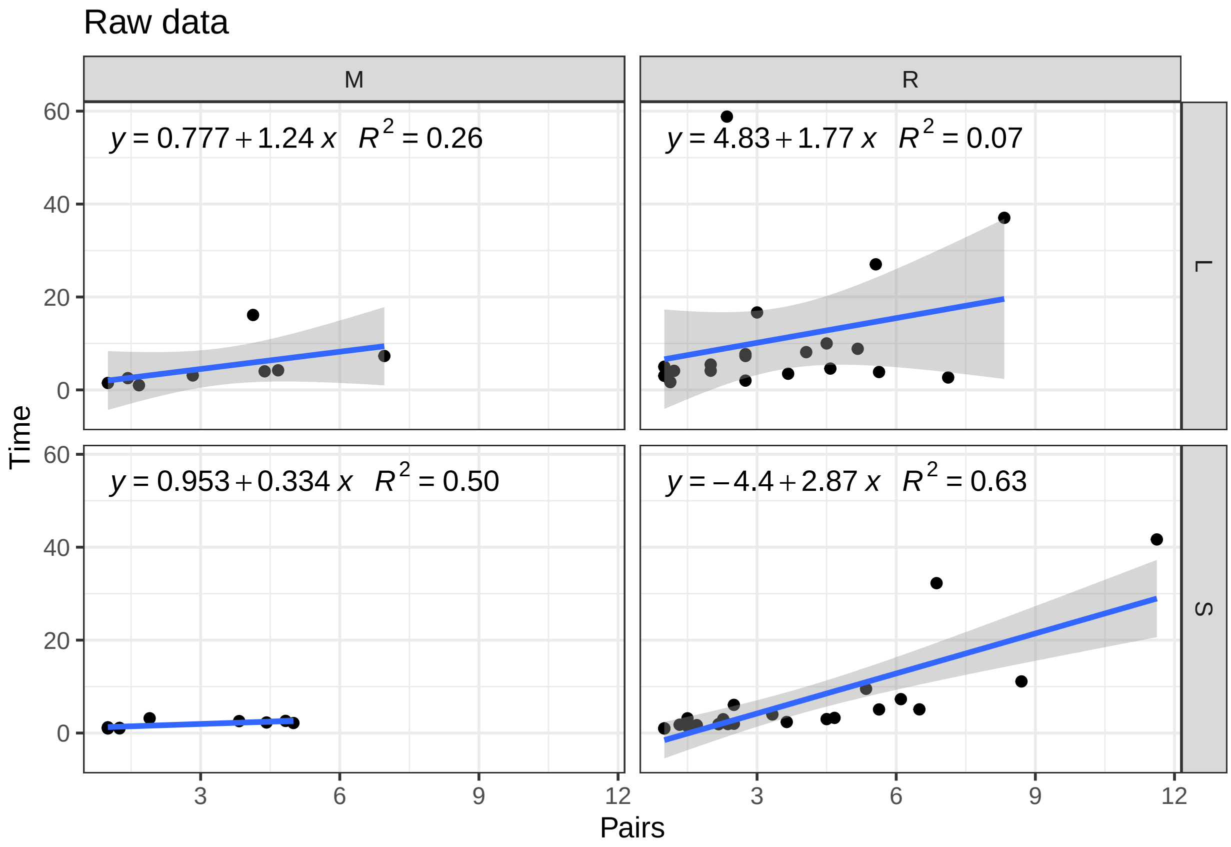
(1) Initial Plotting

There are four different combinations of Size and Status, LR, LM, SR and SM. If we want to find the correlation between *extinction time* as a function of *pairs*, we can make a regression line with *pairs* as the predictor value and *extinction time* as the predicted value.

```
ggplot(data, aes(x = Pairs, y = Time)) +
  geom_point() +
  facet_grid(Size ~ Status) +
  theme(legend.position = "top") +
  geom_smooth(method = "lm", formula = y ~ x) +

  stat_poly_eq(formula = y ~ x,
  aes(label = paste(after_stat(eq.label), after_stat(rr.label), sep = "~~~")),
  parse = TRUE) +

  labs(title = "Raw data") +
  theme_bw()
```

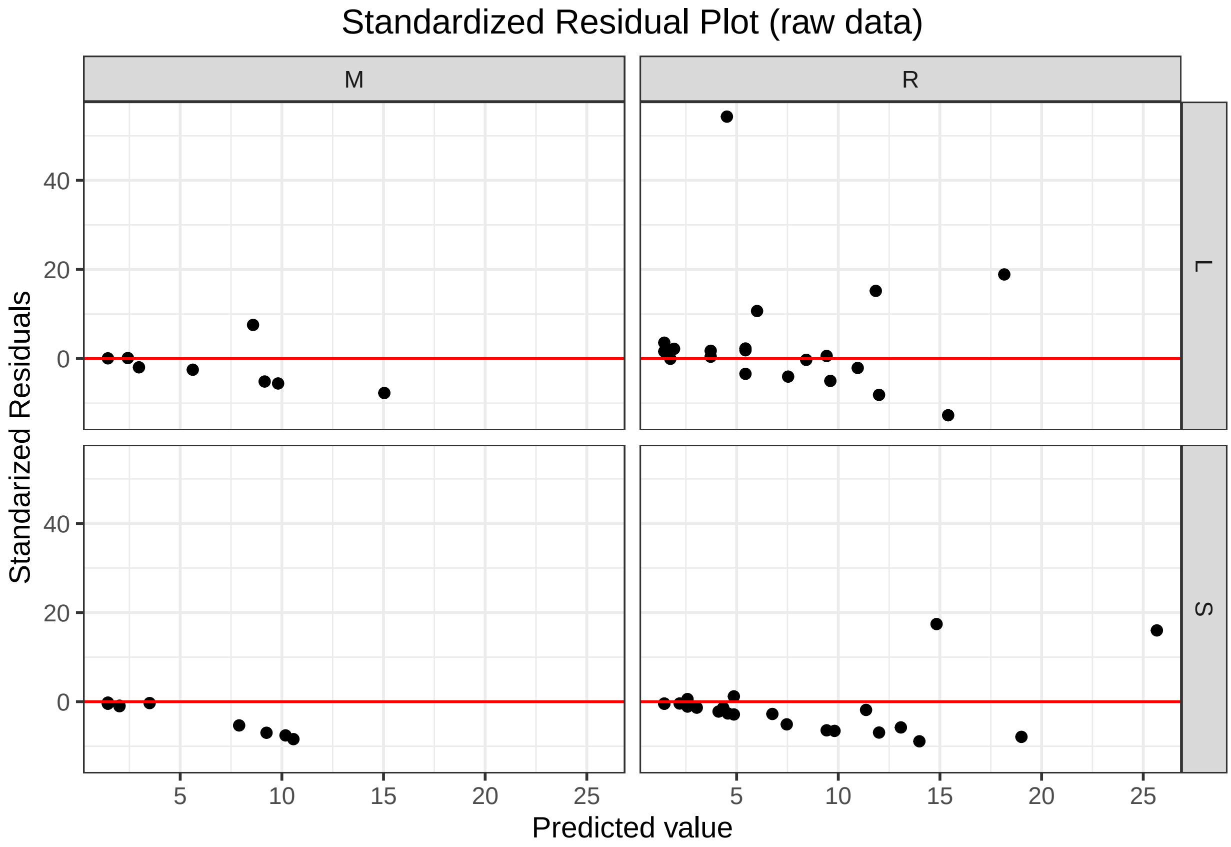


(2) Residual plot of raw (not transformed) data

This is the ONLY correct residual plot

```r
model <- lm(Time ~ Pairs, data = data)
y_hat <- predict(model, newdata = data)

ggplot(data,
  mapping = aes(x = y_hat,
              y = resid(lm(Time ~ Pairs, data = data)))) +
  geom_point() +
  facet_grid(Size ~ Status) +
  geom_hline(yintercept = 0, color = "red") +
  xlab("Predicted value") +
  ylab("Standarized Residuals") +
  labs(title = "Standardized Residual Plot (raw data)") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(legend.position = "bottom")
```



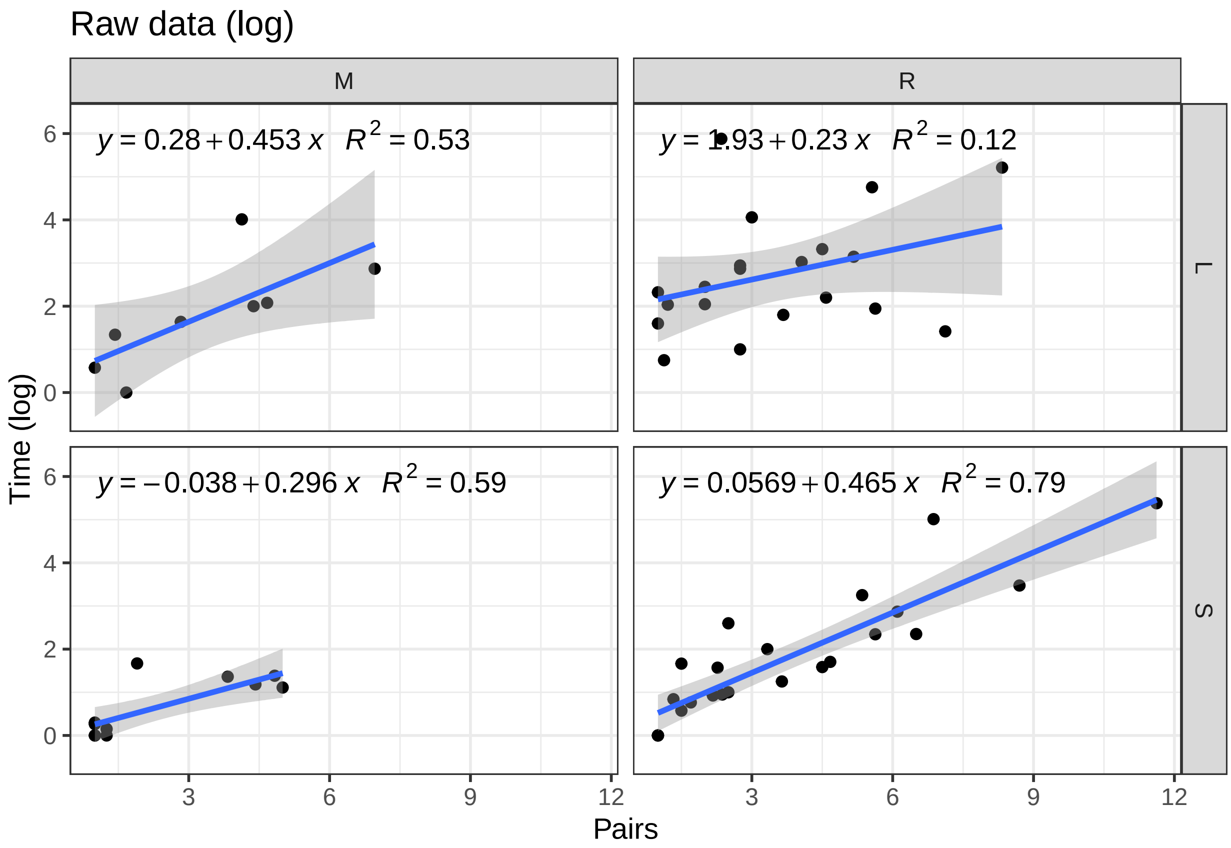
(3) Transformations of time to log_time, sqrt_time, inverse_time

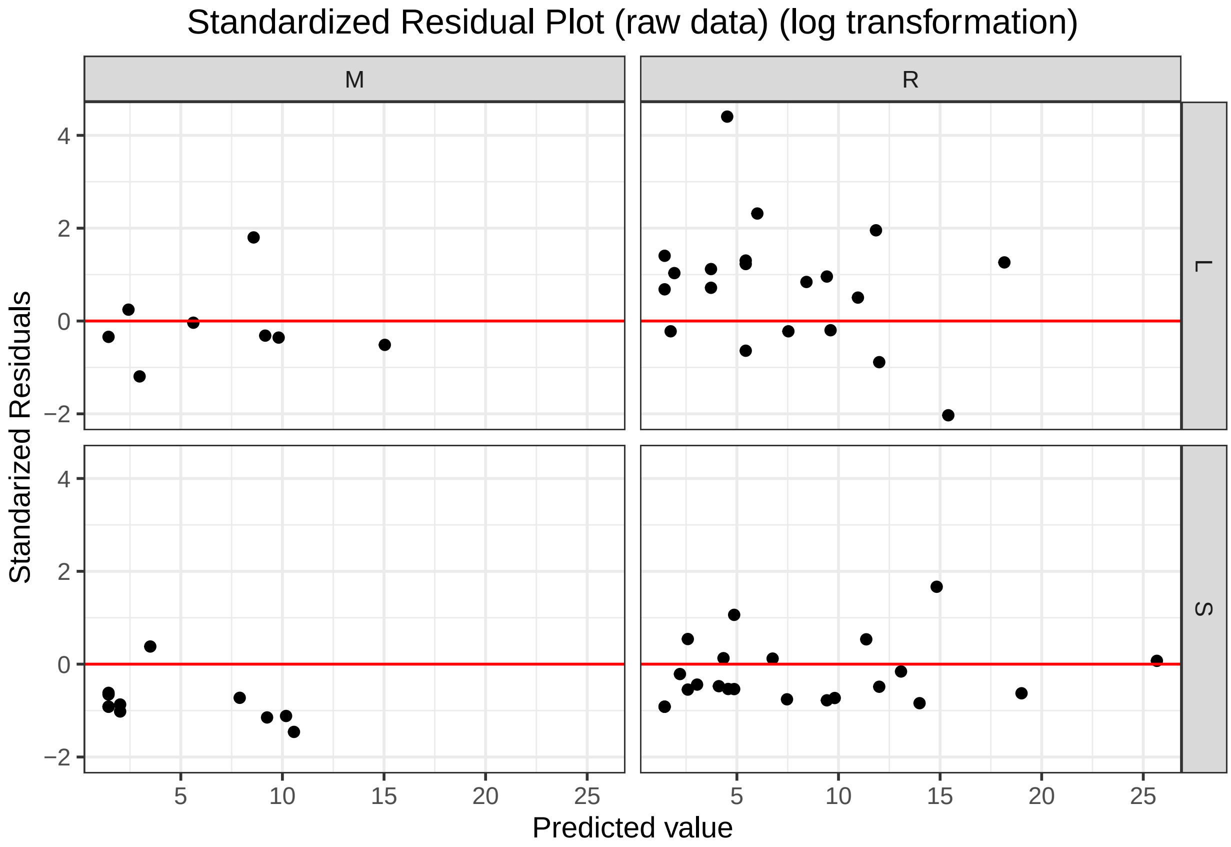
We now try different transformations on the data to see if we can get a better fitted regression line. We try 3 different transformation: log("time"), sqrt("time") and 1/("time").

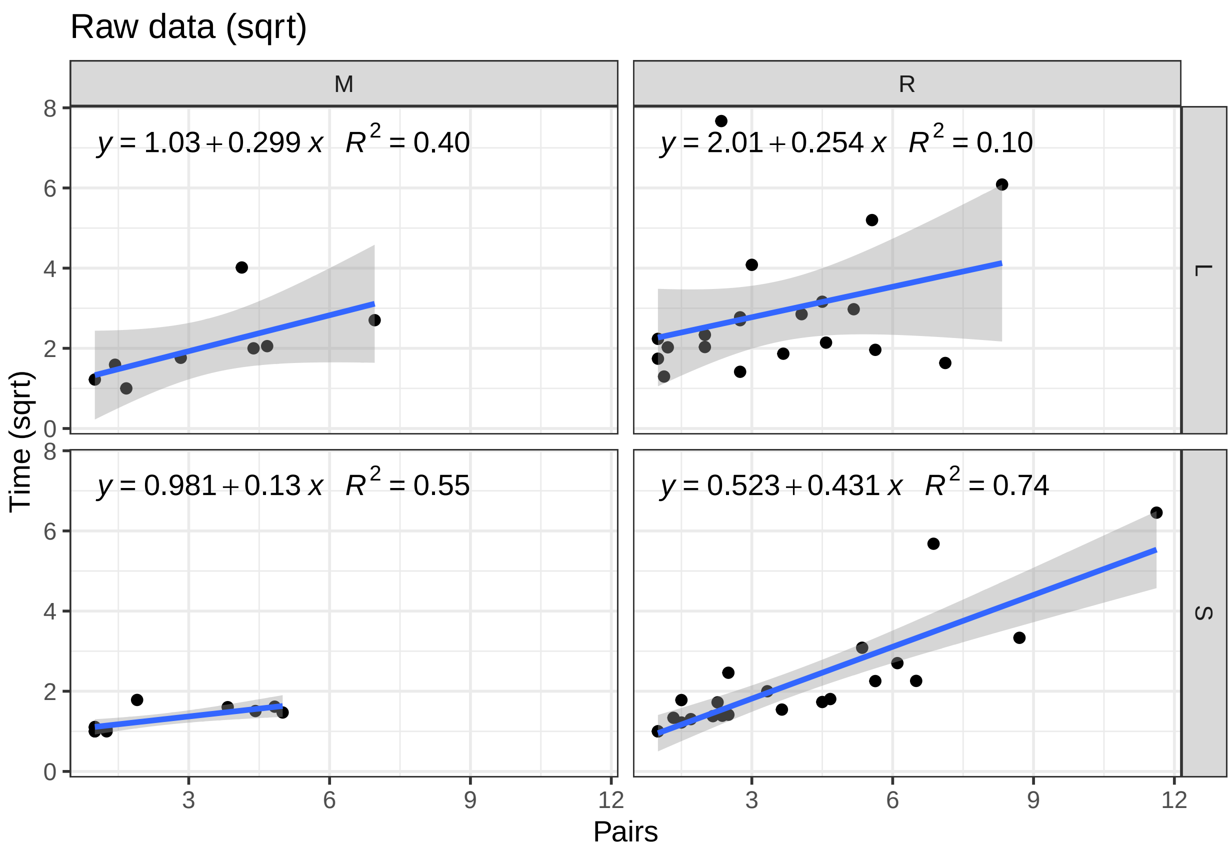
```
data$log_time <- log2(data$Time)
data$sqrt_time <- sqrt(data$Time)
data$inverse_time <- 1/data$Time
head(data)
```

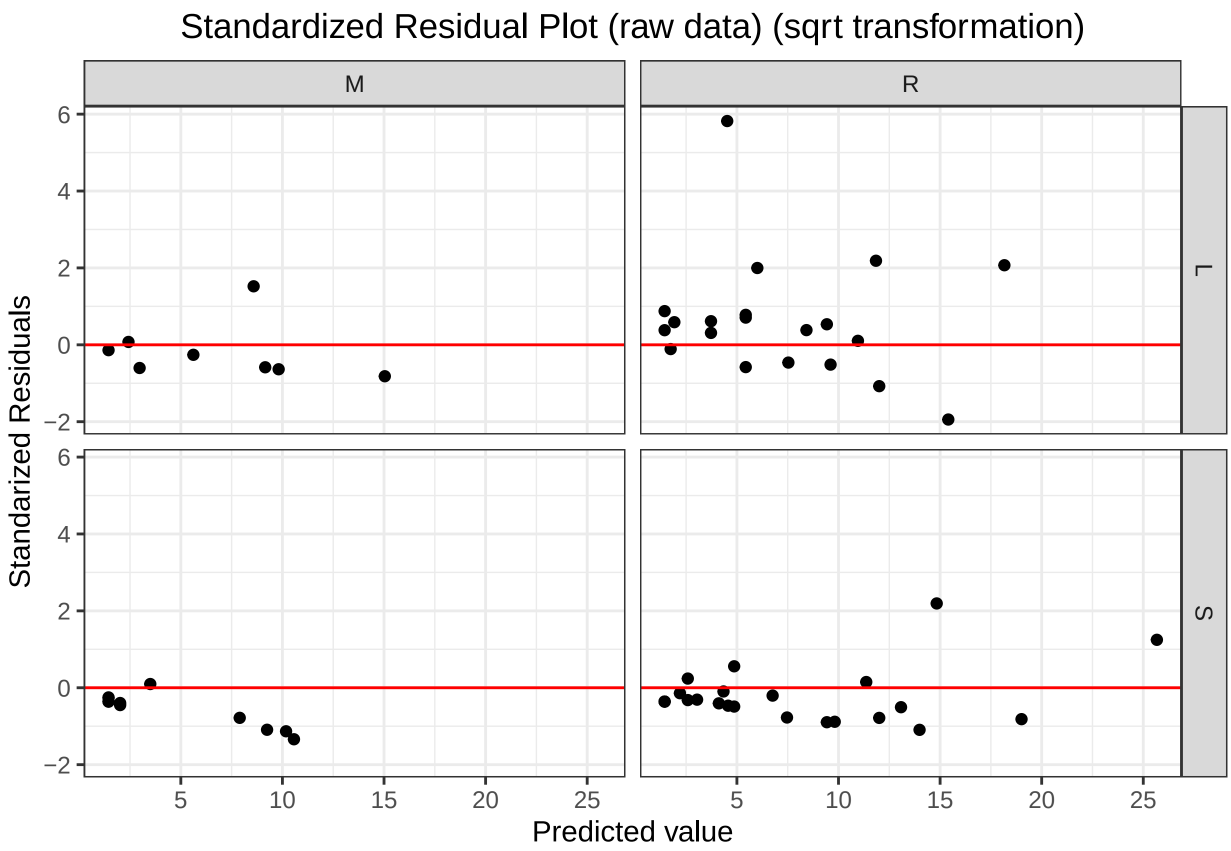
```
##          Species Time Pairs Size Status  log_time sqrt_time inverse_time
## 1    Sparrowhawk 3.03  1.00    L      R 1.5993178  1.740690    0.3300330
## 2        Buzzard 5.46  2.00    L      R 2.4489010  2.336664    0.1831502
## 3        Kestrel 4.10  1.21    L      R 2.0356239  2.024846    0.2439024
## 4      Peregrine 1.68  1.13    L      R 0.7484612  1.296148    0.5952381
## 5 Grey_partridge 8.85  5.17    L      R 3.1456775  2.974895    0.1129944
## 6          Quail 1.49  1.00    L      M 0.5753123  1.220656    0.6711409
```

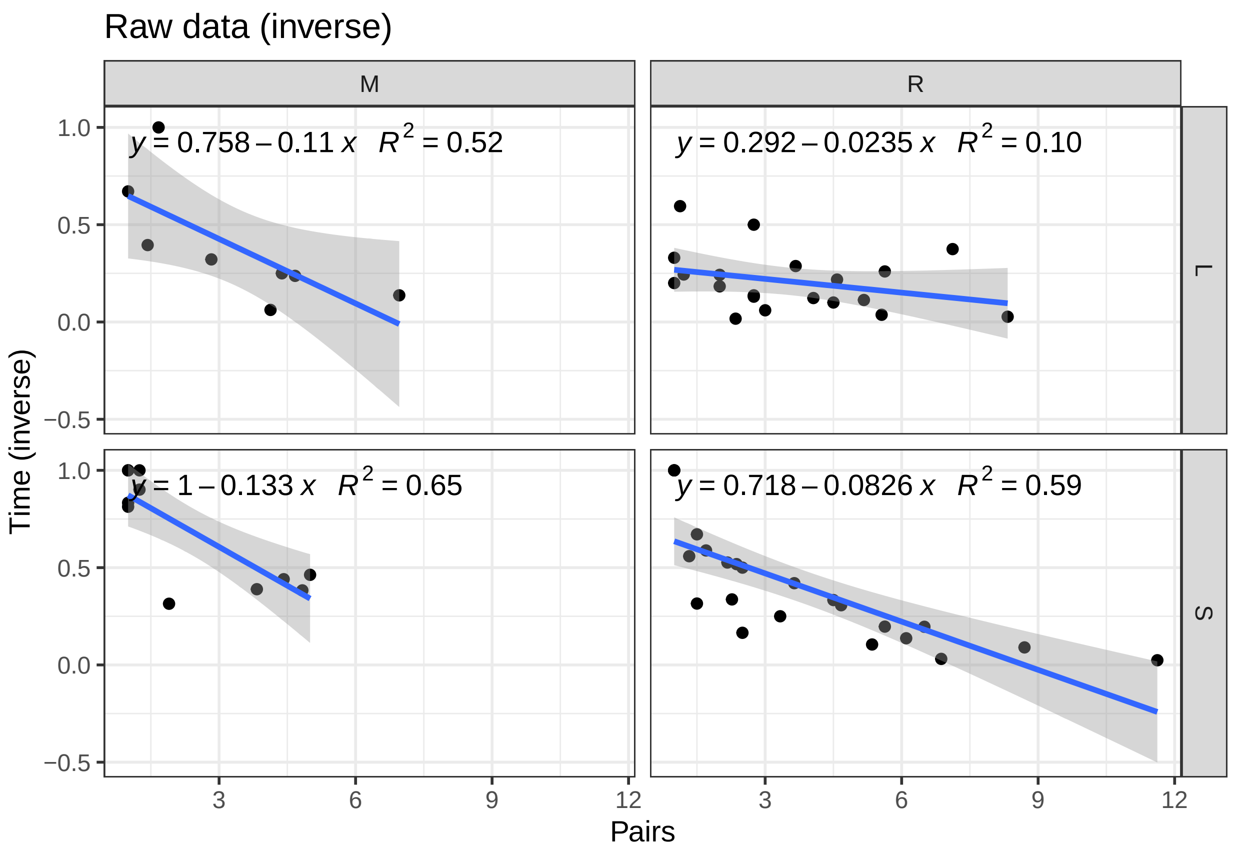
Plotting of log_time:

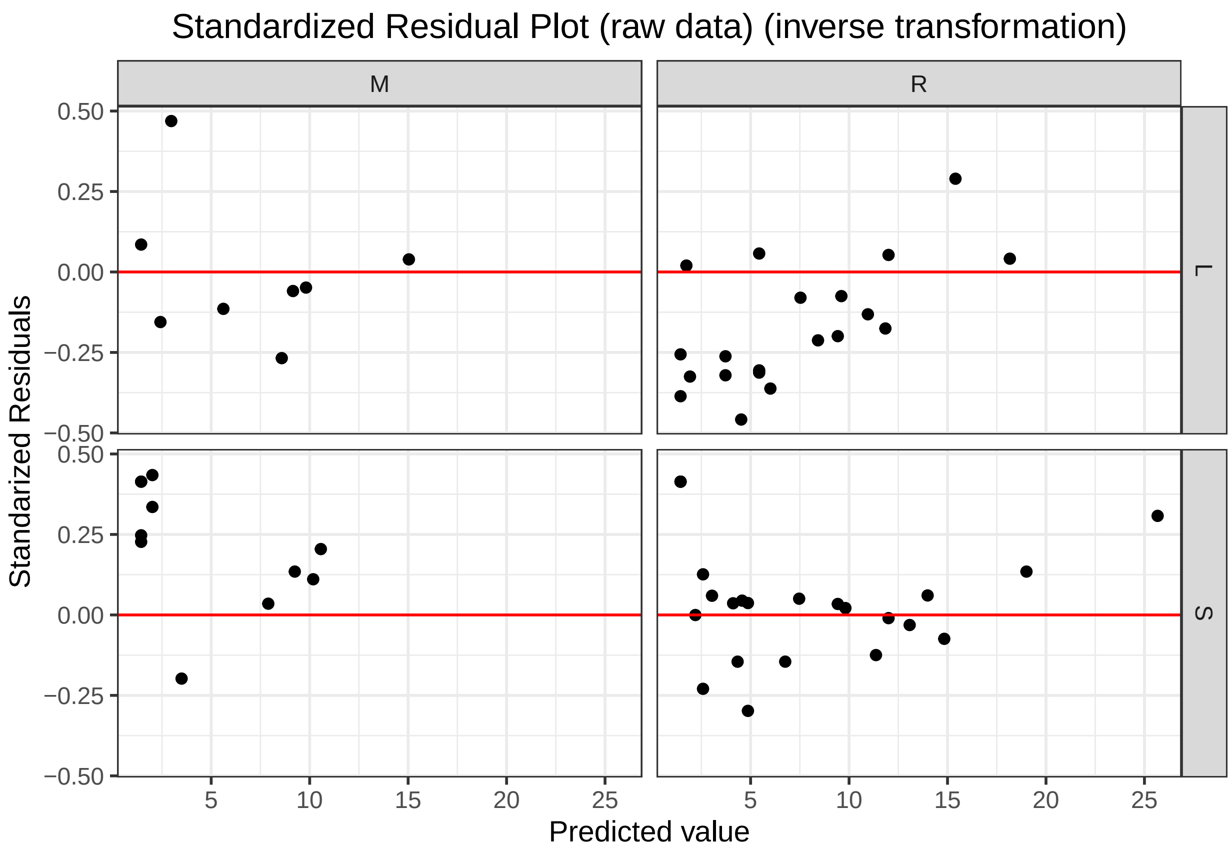



Standardized Residual Plot (raw data) (log transformation)
M
R
L
S
Standarized Residuals
4
2
0
−2
5
10
15
20
25
Predicted value

Raw data (sqrt)
M
R
L
S
$y = 1.03 + 0.299\,x \quad R^2 = 0.40$
$y = 2.01 + 0.254\,x \quad R^2 = 0.10$
$y = 0.981 + 0.13\,x \quad R^2 = 0.55$
$y = 0.523 + 0.431\,x \quad R^2 = 0.74$
Time (sqrt)
0
2
4
6
8
Pairs
3
6
9
12


Standardized Residual Plot (raw data) (sqrt transformation)
M
R
L
S
Standarized Residuals
Predicted value
6
4
2
0
−2
5
10
15
20
25


Raw data (inverse)
M
R
L
S
$y = 0.758 - 0.11\,x \quad R^2 = 0.52$
$y = 0.292 - 0.0235\,x \quad R^2 = 0.10$
$y = 1 - 0.133\,x \quad R^2 = 0.65$
$y = 0.718 - 0.0826\,x \quad R^2 = 0.59$
1.0
0.5
0.0
−0.5
3
6
9
12
Time (inverse)
Pairs



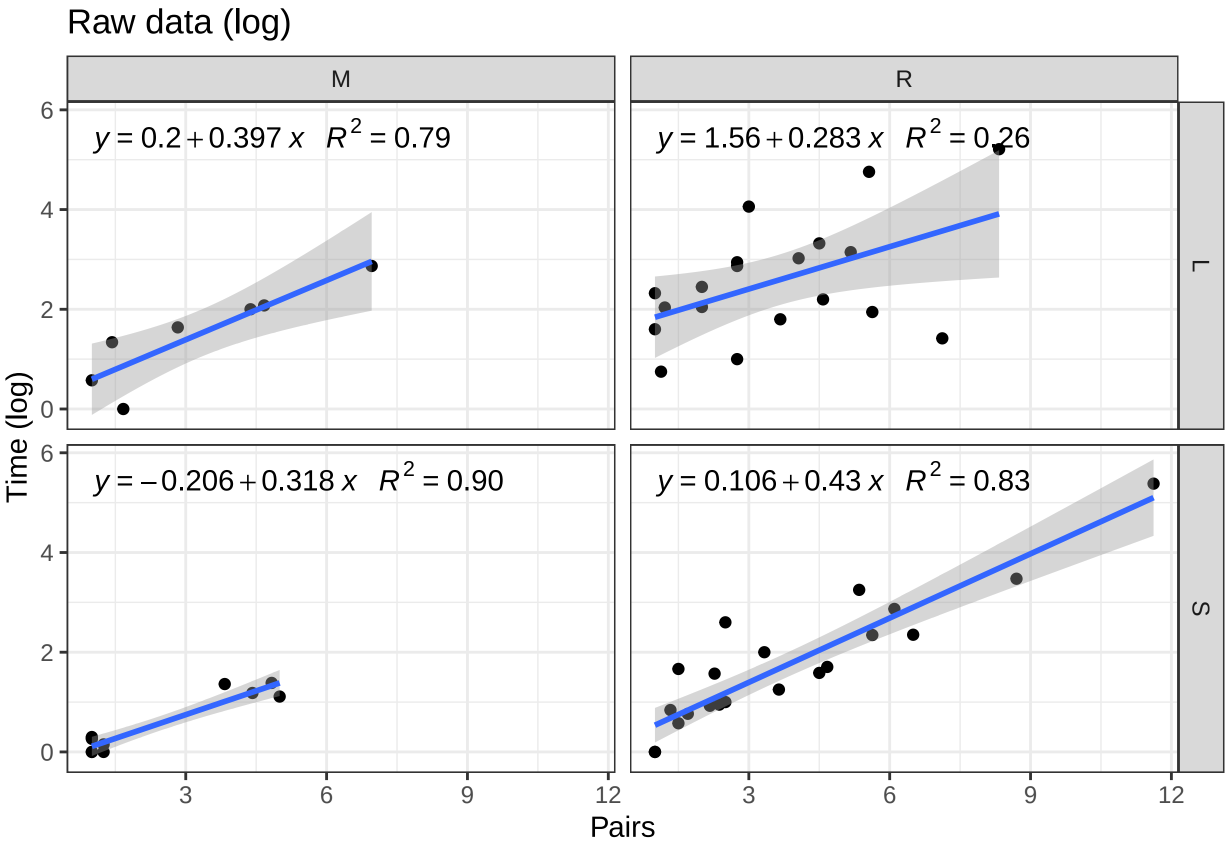
As we can see, the downward trend of the points is no longer apparent. # (4) Removing outliers and comparing
We now proceed to remove outliers to see to what degree they affect the final result. We have opted to only graph and compare the logarithmic graph, as it has the best fit. We start by removing the maximum from the four plots:

```
# Find the max and min values for each of the four plots
get_outliers <- function(data) {
  # Finds residual outliers and returns a vector in the form of (logical_max, logical_min) where logica
  residuals = resid(lm(log2(Time) ~ Pairs, data = data))
  outliers <- c(
    max(residuals),
    min(residuals)
  )
  logicals <- data.frame(
    max = residuals == outliers[1],
    min = residuals == outliers[2]
  )
  return(logicals)
}
LR_mx <- get_outliers(data[
  data$Size == "L" &
  data$Status == "R",
])
LM_mx <- get_outliers(data[
  data$Size == "L" &
  data$Status == "M",
])
SR_mx <- get_outliers(data[
  data$Size == "S" &
  data$Status == "R",
])
SM_mx <- get_outliers(data[
  data$Size == "S" &
  data$Status == "M",
])

# Plot the graph by removing the four Max datapoints
# Get IDs of rows outlier rows
to_remove_ids = as.numeric(c(
  row.names(data[data$Size == "L" & data$Status == "R",][LR_mx$max,]),
  row.names(data[data$Size == "L" & data$Status == "M",][LM_mx$max,]),
  row.names(data[data$Size == "S" & data$Status == "R",][SR_mx$max,]),
  row.names(data[data$Size == "S" & data$Status == "M",][SM_mx$max,])
))

# Filter rows by removing those with the bad IDs
max_filt_data <- data[-to_remove_ids,]

# Plot time
new_Y <- log2(max_filt_data$Time)
print(transform_plots("regres", max_filt_data, new_Y, "log"))
```

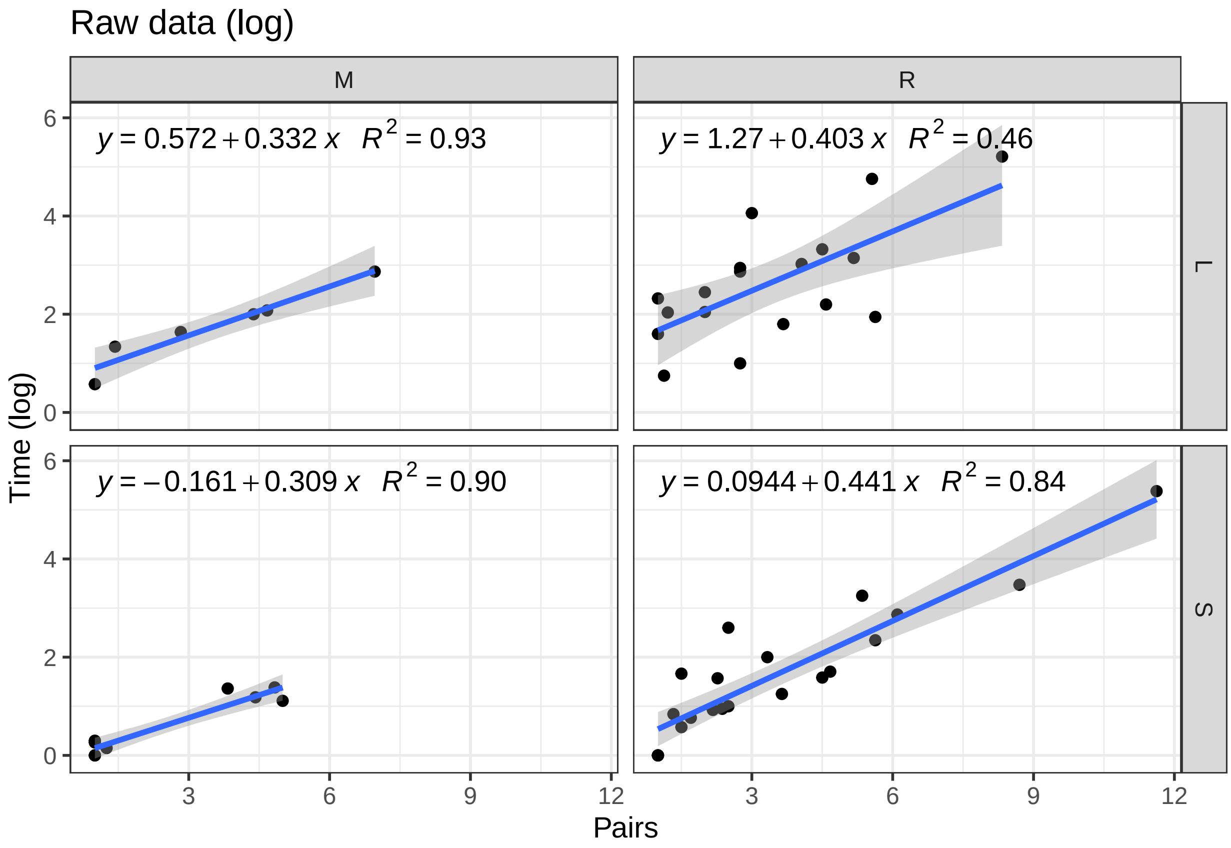


Removing the max outliers from each plot significantly improved our R^2 value. If we also remove the min outliers, it will probably become an even better fit:

```r
# Plot the graph by removing the four Max datapoints
# Get IDs of rows outlier rows (max only)
to_remove_ids = as.numeric(c(
  row.names(data[data$Size == "L" & data$Status == "R",][LR_mx$min | LR_mx$max,]),
  row.names(data[data$Size == "L" & data$Status == "M",][LM_mx$min | LM_mx$max,]),
  row.names(data[data$Size == "S" & data$Status == "R",][SR_mx$min | SR_mx$max,]),
  row.names(data[data$Size == "S" & data$Status == "M",][SM_mx$min | SM_mx$max,])
))

# Filter rows by removing those with the bad IDs (max only)
max_filt_data <- data[-to_remove_ids,]

# Plot time
new_Y <- log2(max_filt_data$Time) # Need to redefine this because the row count has changed
print(transform_plots("regres", max_filt_data, new_Y, "log"))
```



Just removing the two extremes from the plots managed to increase one of our R^2 values from 0.53 to 0.93, an incredible improvement. Despite the fact that removing outliers improved the fit tremendously, it is a bad idea to do so. Data should only be removed if it has been either measured or written down incorrectly. Removing real data points is cherry-picking, and outliers can be a valuable insight into the data that has been collected. The mere fact that they can change line fittings this way is a testament to their influence - and thus their importance.
q