Accelerometer Linear Regression

Spencer Gray & Michael Stinnett

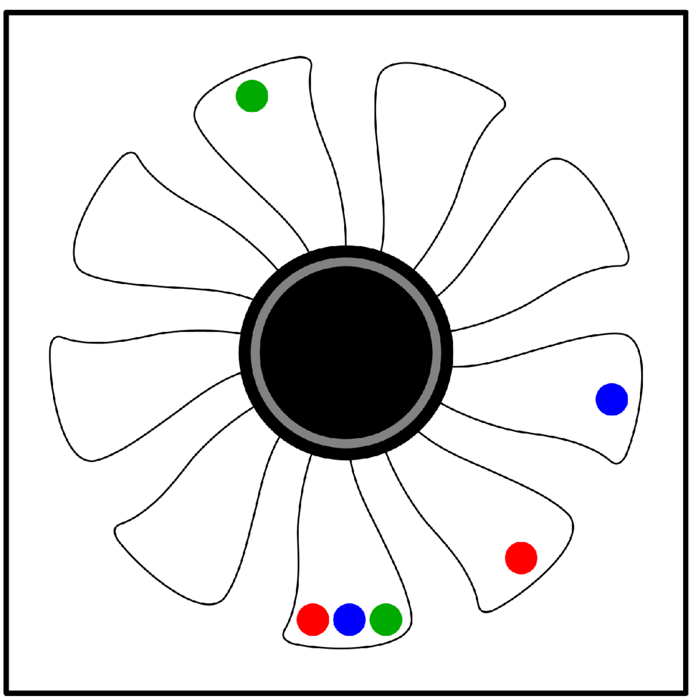
2022-09-25

```
library(ggplot2)
```

Dataset Source: https://archive.ics.uci.edu/ml/datasets/Accelerometer# (https://archive.ics.uci.edu/ml/datasets/Accelerometer#)

This dataset is a recording of movement in the x y and z directions in response to variously placed weights on a fan. There are 153,000 recordings with 5 variables. There are 3 different configurations (Wconfigid) which determines the arrangement of weights on the fan blade. Pctid represents the power applied to the motor. In summary, there are 3 setups each slowly increasing the motor power and recording the disturbance.

Red is configuration 1
Blue is configuration 2
Green is configuration 3



Fan Configuration, Credit: https://www.mdpi.com/1424-8220/19/19/4342/htm (https://www.mdpi.com/1424-8220/19/19/4342/htm)

Predicting variations in Z movement based off of power going to the motor (pctid).

```
accelerometer <- read.csv("accelerometer.csv")
dim(accelerometer)
```

```
## [1] 153000      5
```

```
head(accelerometer)
```

```
##   wconfid pctid     x      y      z
## 1       1    20 1.004  0.090 -0.125
## 2       1    20 1.004 -0.043 -0.125
## 3       1    20 0.969  0.090 -0.121
## 4       1    20 0.973 -0.012 -0.137
## 5       1    20 1.000 -0.016 -0.121
## 6       1    20 0.961  0.082 -0.121
```

```
mean1 <- mean(accelerometer$z)
med1 <- median(accelerometer$z)
range1 <- range(accelerometer$z)
var1 <- var(accelerometer$z)
sd1 <- sd(accelerometer$z)

print(paste('mean: ', mean1))
```

```
## [1] "mean:  -0.117769163398693"
```

```
print(paste('median: ', med1))
```

```
## [1] "median:  -0.125"
```

```
print(paste('range: ', range1))
```

```
## [1] "range:  -5.867" "range:  6.086"
```

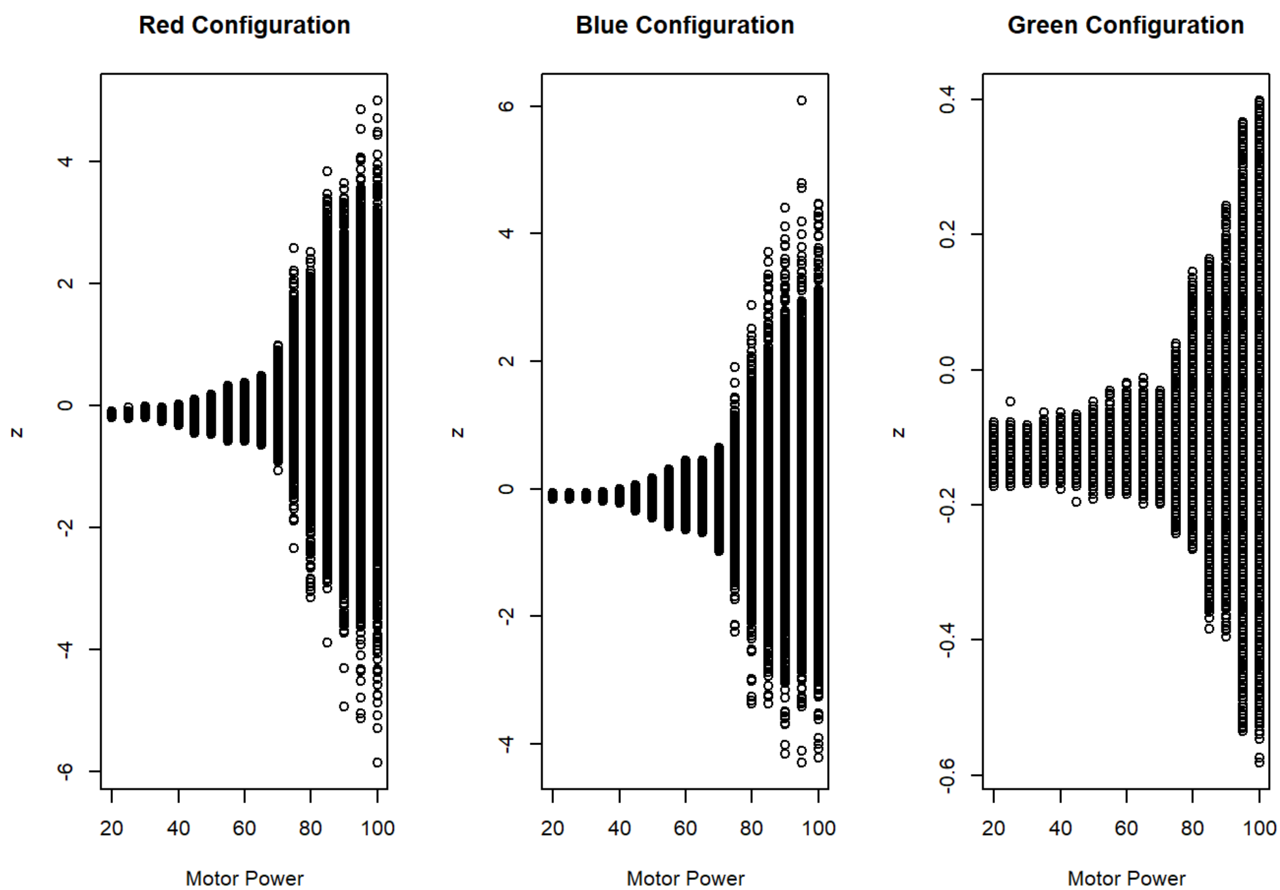
```
print(paste('var: ', var1))
```

```
## [1] "var:  0.267297088538572"
```

```
print(paste('sd: ', sd1))
```

```
## [1] "sd:  0.517007822511973"
```

```
par(mfrow=c(1,3))
with(accelerometer[accelerometer$wconfid<2,], plot(pctid, z, xlab="Motor Power", main="Red Configu
ration"))
with(accelerometer[accelerometer$wconfid>1 & accelerometer$wconfid < 3,], plot(pctid, z,xlab="Moto
r Power", main="Blue Configuration"))
with(accelerometer[accelerometer$wconfid>2,], plot(pctid, z,xlab="Motor Power", main="Green Config
uration"))
```



Linear Regression Model

```
set.seed(1234)
i <- sample(1:nrow(accelerometer), nrow(accelerometer)*0.8, replace = FALSE)
train <- accelerometer[i,]
test <- accelerometer[-i,]
```

Statistical Analysis of Linear Model (1 predictor)

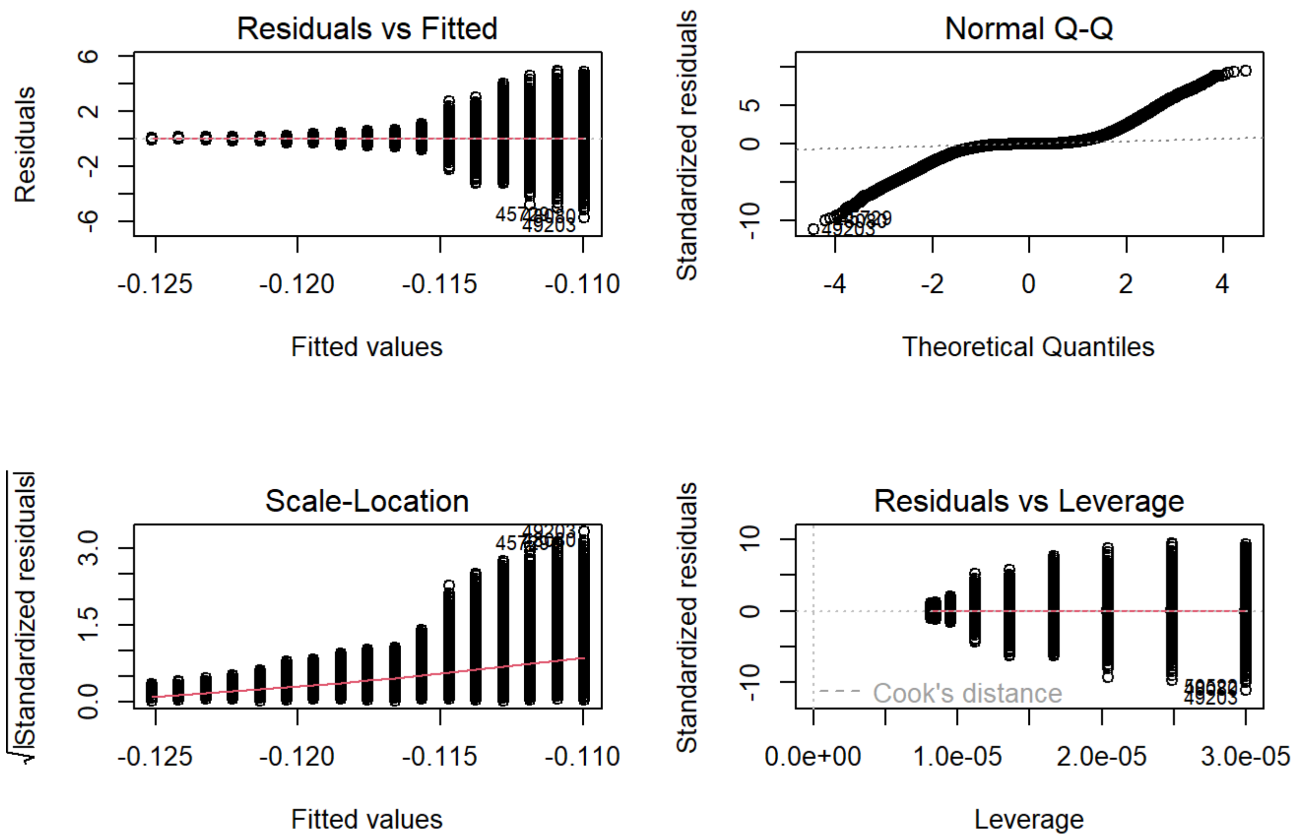
The relationship in the Residuals vs Fitted plot appears to show a 'linear' relationship. Moreso a line centered and parallel with the x-axis. However this can be caused by a symmetrical oscillation centered around the x-axis. We will most likely need more predictors and tweaks to provide an accurate approximation.
The error (residuals) in the Q-Q plot appear to deviate strongly, follow the line, and then deviate again. This could be

caused by the three separate configurations producing higher or lower variance. This is further revealed in the scale-location plot revealing that our z-axis variance increases as we increase our motor power. Residuals vs Leverage plot does not seem to share any substantial information on strong outliers

```
lm1 <- lm(z ~ pctid, data=train)
summary(lm1)
```

```
##
## Call:
## lm(formula = z ~ pctid, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7570 -0.0543 -0.0018  0.0515  4.9039
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.289e-01  3.911e-03 -32.968  < 2e-16 ***
## pctid        1.898e-04  6.038e-05   3.143  0.00167 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5175 on 122398 degrees of freedom
## Multiple R-squared:  8.072e-05,  Adjusted R-squared:  7.255e-05
## F-statistic: 9.881 on 1 and 122398 DF,  p-value: 0.00167
```

```
par(mfrow=c(2,2))
plot(lm1)
```

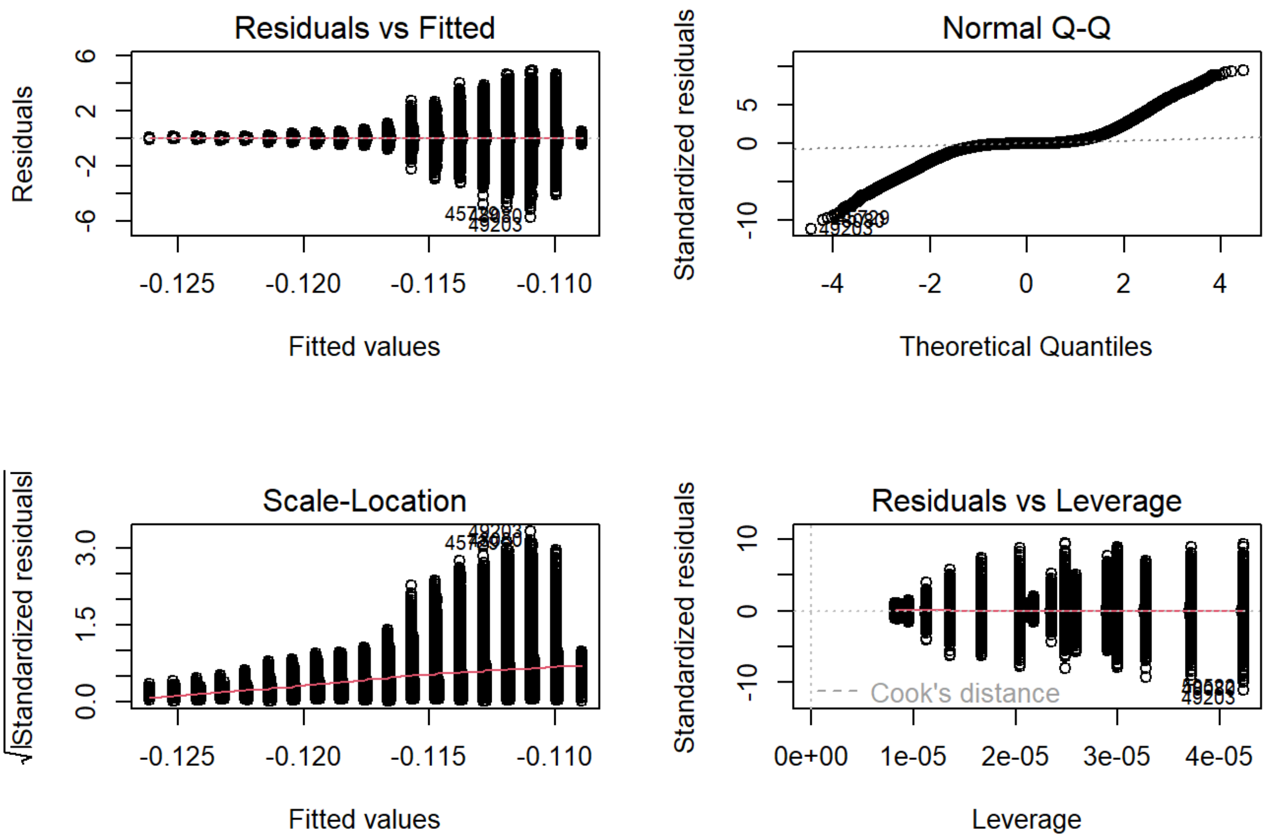


Statistical Analysis of Linear Model (2 predictors)

```
lm2 <- lm(z ~ pctid+wconfid, data=train)
summary(lm2)
```

```
##
## Call:
## lm(formula = z ~ pctid + wconfid, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.7560 -0.0544 -0.0018  0.0516  4.9039
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.310e-01  5.333e-03 -24.557  < 2e-16 ***
## pctid        1.898e-04  6.038e-05   3.144  0.00167 **
## wconfid      1.016e-03  1.811e-03   0.561  0.57493
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5175 on 122397 degrees of freedom
## Multiple R-squared:  8.329e-05,	Adjusted R-squared:  6.695e-05
## F-statistic: 5.098 on 2 and 122397 DF,  p-value: 0.006112
```

```
par(mfrow=c(2,2))
plot(lm2)
```



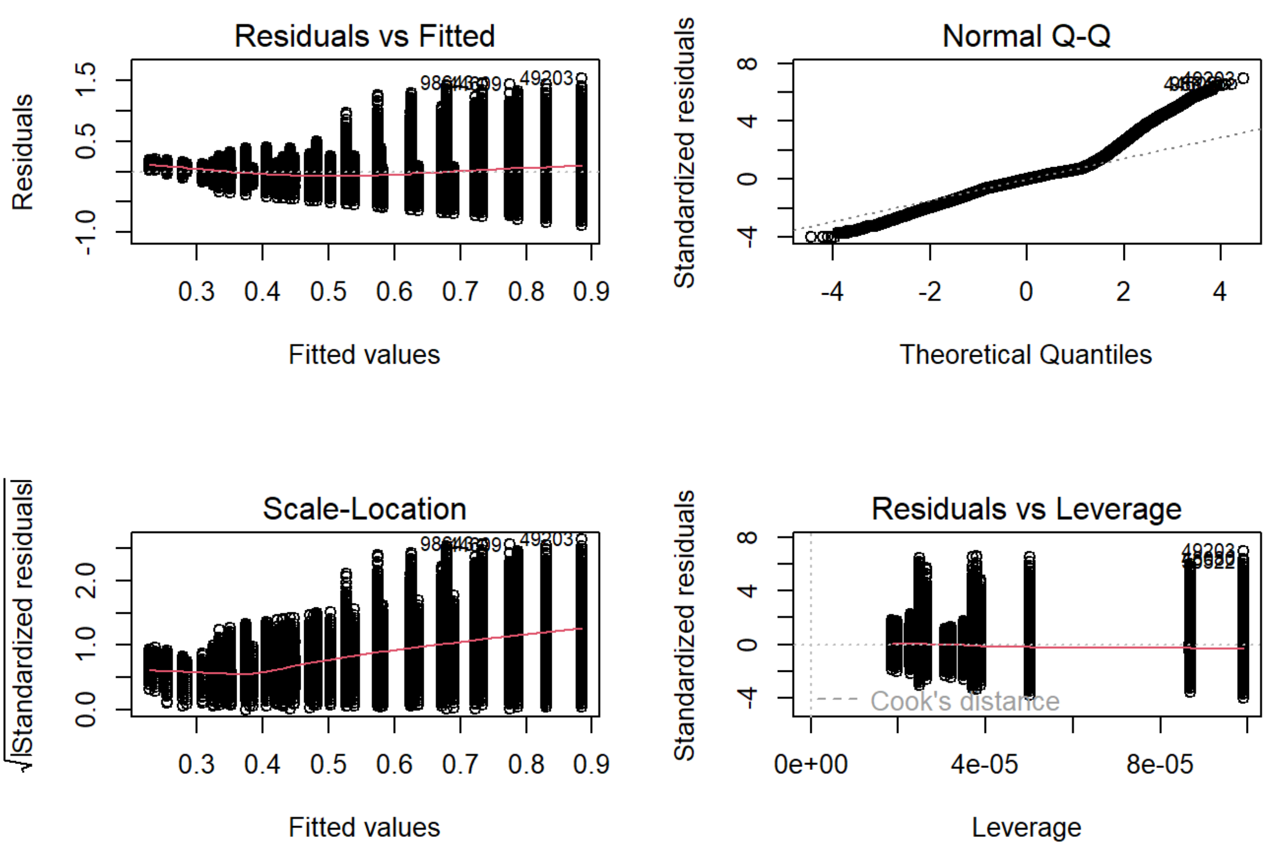
Statistical Analysis using Other Methods

To get a more accurate estimate I am selecting one sample of tests, fan configuration 1. Then taking the absolute value of the z movements (to remove symmetry causing net zero slope) and doing a polynomial regression.

```
lm3 <- with(train[train$wconfid<2,], lm(sqrt(abs(z)) ~ pctid + I(pctid^2) + I(pctid^3) + wconfid,
 data = train))
summary(lm3)
```

```
##
## Call:
## lm(formula = sqrt(abs(z)) ~ pctid + I(pctid^2) + I(pctid^3) +
##     wconfid, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.88436 -0.11438 -0.00127  0.10228  1.53783
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.142e-01  9.455e-03   86.11   <2e-16 ***
## pctid       -1.730e-02  5.620e-04  -30.79   <2e-16 ***
## I(pctid^2)   2.873e-04  1.012e-05   28.39   <2e-16 ***
## I(pctid^3)  -9.760e-07  5.585e-08  -17.47   <2e-16 ***
## wconfid     -9.693e-02  7.748e-04 -125.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2214 on 122395 degrees of freedom
## Multiple R-squared:  0.3692, Adjusted R-squared:  0.3692
## F-statistic: 1.791e+04 on 4 and 122395 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(lm3)
```



As seen in both linear model 1 and linear model 2, they both got similar results. A misrepresentation of the data and a prediction that doesn't even come close to the test data. The statistical summary listed below shows a covariance less than 0.01, but with linear model 3 we got a covariance of 0.6. The first step was to separate the three different categories. For model 3 I restricted the test and training data to fan configuration 1. Then removed x-axis symmetry with abs() before using sqrt() to reduce the exponential behavior. A polynomial fit with a degree of 3 was used for best fit, compensating for any quadratic behavior.

Statistical Summary

```
pred1 <- predict(lm1, newdata = test)
cor1 <- cor(pred1, test$z)
mse1 <- mean((pred1 - test$z)^2)
rmse1 <- sqrt(mse1)

pred2 <- predict(lm2, newdata = test)
cor2 <- cor(pred2, test$z)
mse2 <- mean((pred2 - test$z)^2)
rmse2 <- sqrt(mse2)

pred3 <- with(test[test$wconfid<2,], predict(lm3, newdata = test))
cor3 <- cor(pred3, sqrt(abs(test$z)))
mse3 <- mean((pred3 - sqrt(abs(test$z)))^2)
rmse3 <- sqrt(mse3)

print(paste('correlation: ', cor1, cor2, cor3))
```

```
## [1] "correlation:  0.0088989796081442 0.00959082756295296 0.611563022552147"
```

```
print(paste('mse: ', mse1, mse2, mse3))
```

```
## [1] "mse:  0.265270527756122 0.26526716143052 0.0486342589125535"
```

```
print(paste('rmse: ', rmse1, rmse2, rmse3))
```

```
## [1] "rmse:  0.515044199808252 0.515040931801076 0.220531763953752"
```

Summary

The first two models included all of the different fan configurations, resulting in inconsistent results. The biggest factor that separated the third model is to categorize data properly before training a model.