**Variables**

**Y = Average Speed (response variable) measured in miles/hour**

**X1 = Average Heart Rate (numerical predictor) measured in beats/min**

**X2 = Average Cadence (numerical predictor) measured in steps/min**

**X3 = Distance (numerical predictor) measured in miles**

**X4 = Runner (categorical predictor) where 0=Adam and 1=Leah**

**X5 = Time of Day (categorical predictor) where 0=Night and 1=Morning, Afternoon or Evening**

**Base Model**

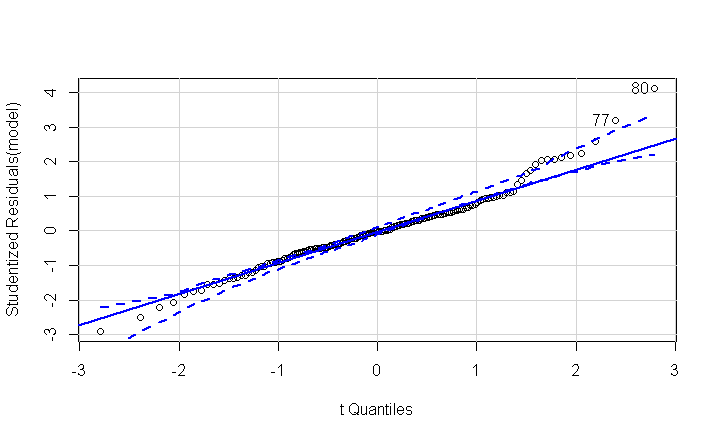
Call:

lm(formula = avgspeed ~ avghr + avgcadence + distance + runner +

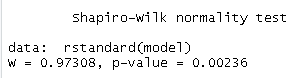
time2 + avghr:avgcadence + avgcadence:distance, data = running)

**Y = 4.945 – 0.058\*X1 – 0.008\*X2 + 0.319\*X3 – 0.808\*X4 – 0.002\*X5 + 0.001\*X1\*X2 – 0.003\*X2\*X3**

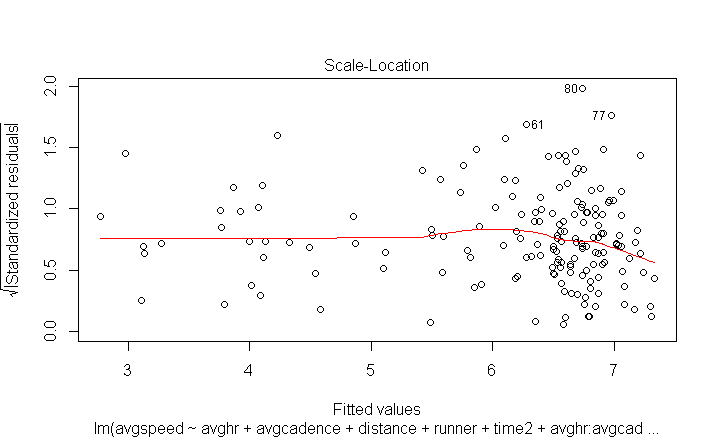
Adjusted R-squared: 0.9125

**Normality**

Normality is clearly violated in the base model due to high influence points. This is evident in the QQPlot and Shapiro-Wilk test.



**Homoscedasticity**

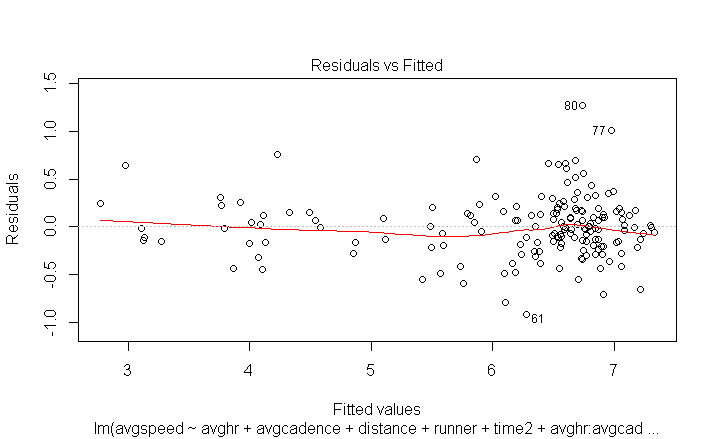


There is no evidence to disprove homoscedasticity. There is no obvious fan shape in the scale-location plot. We also see a very large p-value in the Breusch-Pagan test.

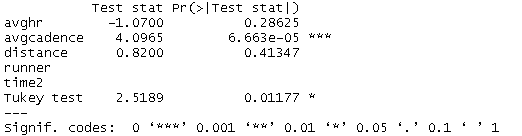


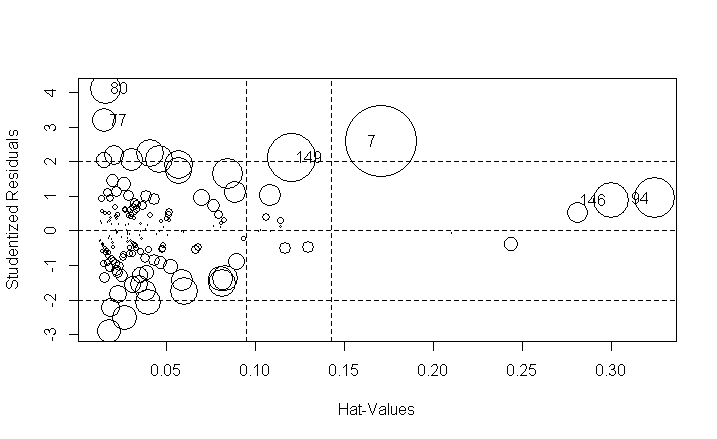
**Linearity**

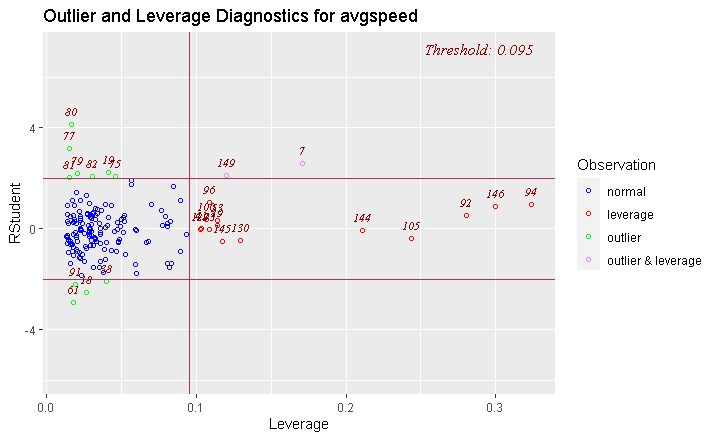
The linearity assumption seems OK here. There is only a very slight decreasing trend in the residuals vs fitted values plot.



Interestingly, we see evidence that adding a quadratic term for avgcadence may be statistically significant while investigating the “residualPlots” command:



**Influential Points**

There are a good number of influential points in our model. We will investigate the effects of removing some of these influential observations with the subset command.

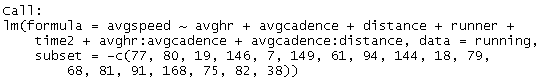
**Game Plan**

The base model looks solid aside from the normality violation. The adjusted r-squared for the base model is 0.9125 which is difficult to improve upon, but we will investigate:

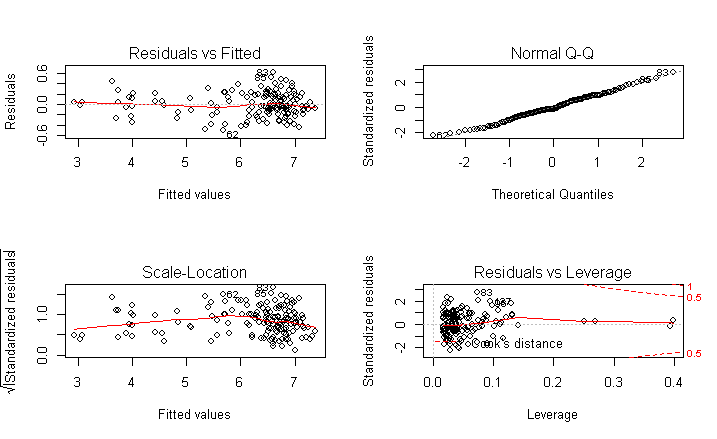
* A model where we remove influential points
* A model where we add a quadratic predictor and remove a categorical predictor
* A model where we use weighted least-squares to correct assumption violations
* A model where we use a logarithmic transformation on the response variable
* Do we need other types of transformations?

**Addressing the Assumption Violations**

We begin our investigation by looking at a model where we remove several influential observations:



**Model1**

**Y = 2.899 – 0.053\*X1 + 0.006\*X2 + 0.468\*X3 – 0.757\*X4 + 0.005\*X5 + 0.0004\*X1\*X2 – 0.004\*X2\*X3**

Adjusted R-squared: 0.9502

Obviously this model is going to show significant improvement because we are essentially cherry picking the best data points to use. This is not the preferred was of obtaining a good model, but it does do a good job of addressing the normality violation.

Next, we used the results of the “residualPlots” command which suggested we should add a quadratic predictor to our model. I didn’t preserve the code (because it took 15 iterations) but I ran the “drop1” command to arrive at the optimal model using the quadratic predictor, which unfortunately involved dropping a non-significant categorical predictor (time of day) from the model:

Call:

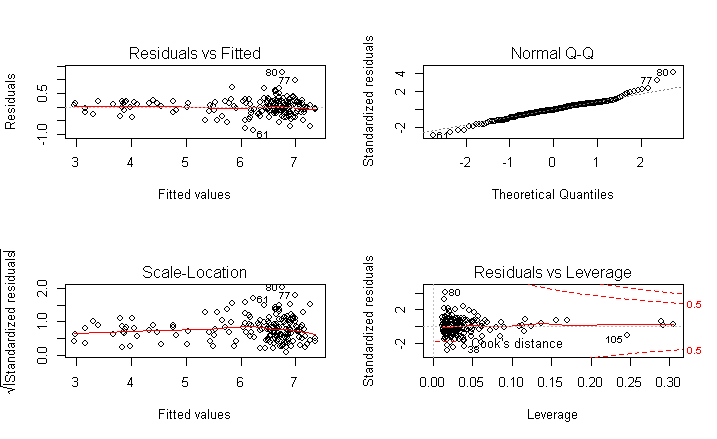
lm(formula = avgspeed ~ avghr + avgcadence + I(avgcadence^2) +

distance + runner + I(avgcadence^2):distance, data = running)

**Model2**

**Y = 8.344 + 0.017\*X1 – 0.123\*X2 + 0.001\*X22 + 0.122\*X3 – 0.878\*X4 – 0.000008\*X22\*X3**

Adjusted R-squared: 0.9211

The first thing I noticed here was the coefficient of the quadratic term was so close to zero that it was almost negligibly quadratic (0\*X2 = 0). This could be related to the units used to represent the response variable, however. Representing speed with feet per second instead of miles per hour may improve coefficients.

Looking closely at the plots we also start to notice that perhaps we now have a homoscedasticity violation.

Non-constant Variance Score Test

Variance formula: ~ fitted.values

Chisquare = 4.261077, Df = 1, p = 0.038995

This does indeed seem to be the case. We try using weights to correct the violation:

Call:

lm(formula = avgspeed ~ avghr + avgcadence + I(avgcadence^2) +

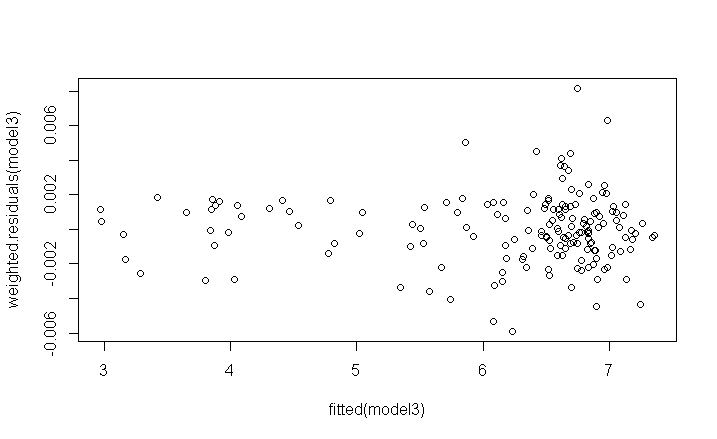
distance + runner + I(avgcadence^2):distance, data = running,

weights = 1/avgcadence^2)

**Model 3**

**Y = 8.366 + 0.018\*X1 – 0.135\*X2 + 0.0007\*X22 + 0.113\*X3 - .870\*X4 – 0.000007\*X22\*X3**

Adjusted R-squared: 0.9427

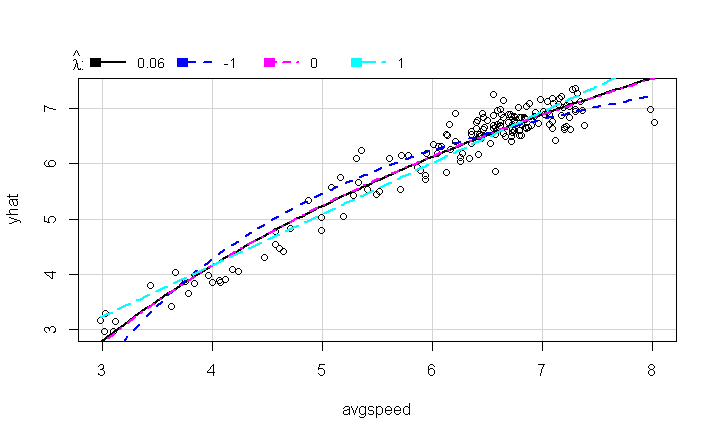


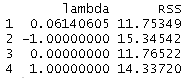
Adding weights does improve (albeit very slightly) the fan shape. An improvement (again, slight) is more apparent in the Breush-Pagan test:

Non-constant Variance Score Test

Variance formula: ~ fitted.values

Chisquare = 1.234065, Df = 1, p = 0.26662

Next we used the “invResPlot” command to investigate the need for transforming our response variable. The optimal lambda value appeared to be “0” (although there was not much difference in the lambdas for -1 and 1) thus we looked at using a logarithmic transformation:



Call:

lm(formula = log(avgspeed) ~ avghr + avgcadence + I(avgcadence^2) +

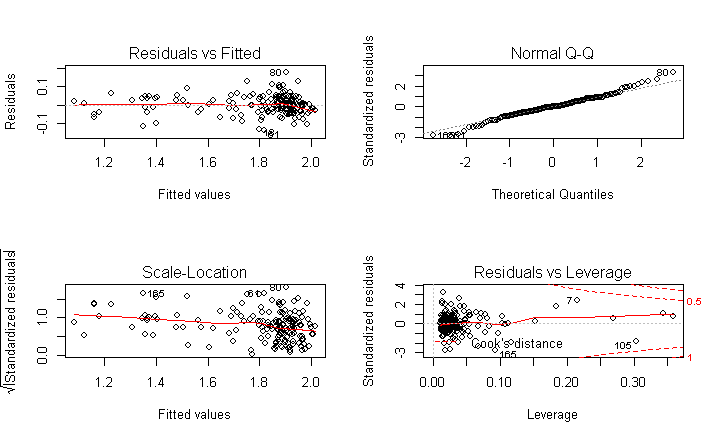
distance + runner + I(avgcadence^2):distance, data = running,

weights = 1/avgcadence^2)

**Model 4**

**Y = 1.021 + 0.004\*X1 - 0.009\*X2 + 0.00007\*X22 + 0.014\*X3 – 0.155\*X4 – 0.000001\*X22\*X3**

Adjusted R-squared: 0.9578



The main downside here is the transformation worsens the homoscedasticity assumption that we just corrected in the previous model:

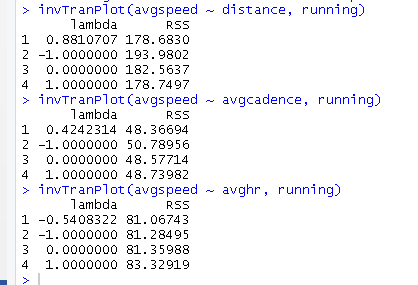
Non-constant Variance Score Test

Variance formula: ~ fitted.values

Chisquare = 5.854747, Df = 1, p = 0.015535

If we were only looking at adjusted r-squared this model seems to be the best. However, the sacrifice of our assumptions is not inspiring.

Finally, we investigate the need to use transformations on the other predictors:

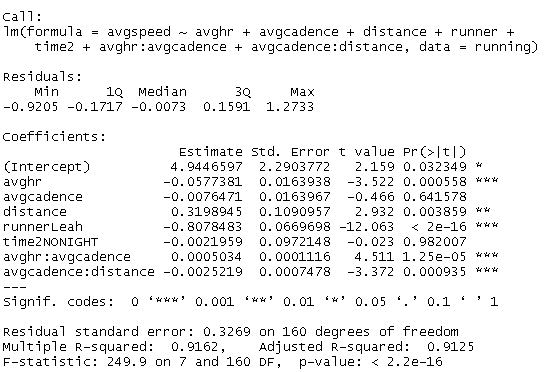


There is nothing here to suggest any great improvements from adding transformations to the other predictors.

**Final Model Summary**

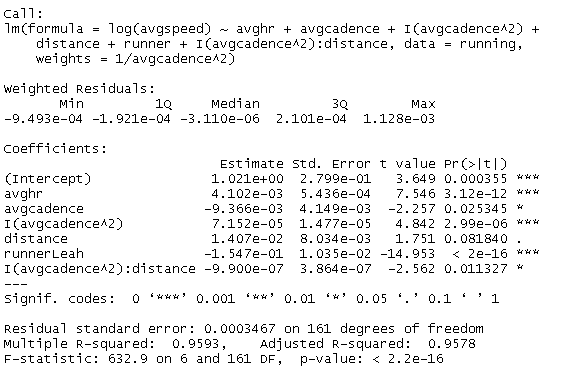
**Base Model:**

**Y = 4.945 – 0.058\*X1 – 0.008\*X2 + 0.319\*X3 – 0.808\*X4 – 0.002\*X5 + 0.001\*X1\*X2 – 0.003\*X2\*X3**



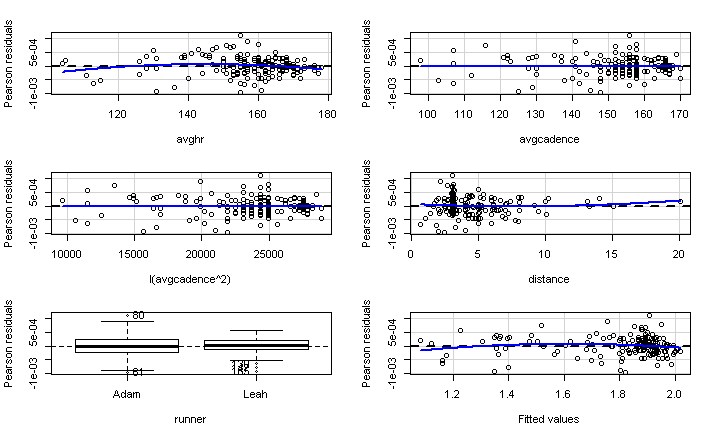
**Model 4:**

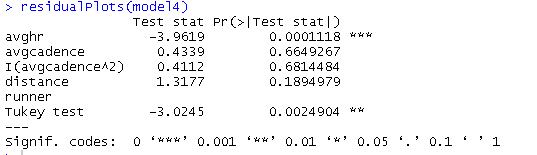
**Y = 1.021 + 0.004\*X1 - 0.009\*X2 + 0.00007\*X22 + 0.014\*X3 – 0.155\*X4 – 0.000001\*X22\*X3**



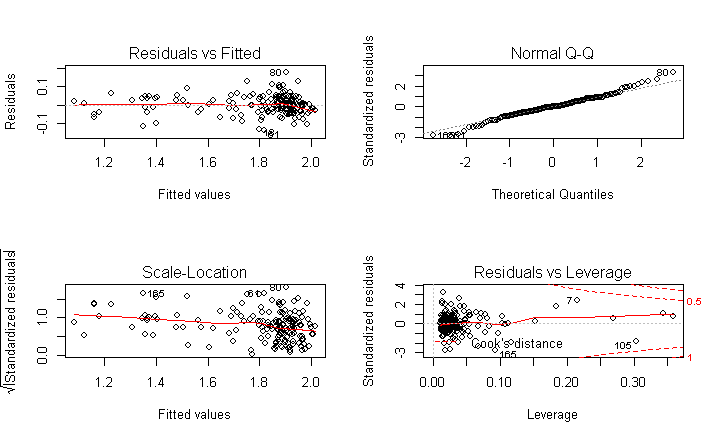
There is some improvement made by the alterations we made to the base model. All of our predictors are now significant at the 10% level or better. We also see improvement in the adjusted r-squared.

We didn’t see any interesting sign changes that couldn’t be explained. Average heart rate, for example, was positive in the base model without the inclusion of an interaction term, so we weren’t surprised to see it return to positive when the interaction term was removed.



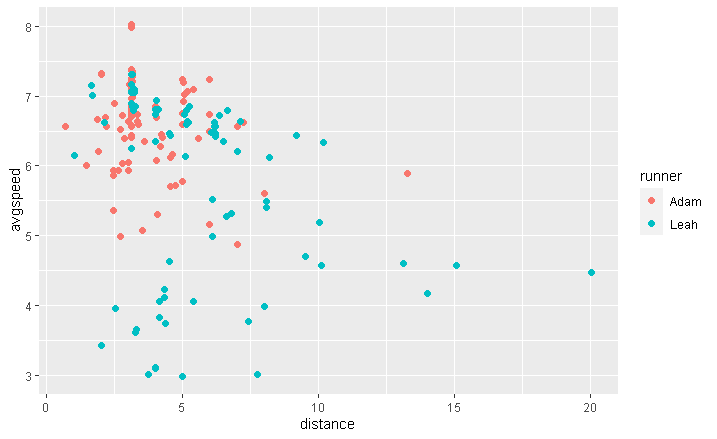


We did find it interesting that we may see an improvement by also including a quadratic term for avghr, but we did not investigate this inclusion. Perhaps this may improve the homoscedasticity assumption that is still violated (although I think it’s unlikely)



This final model does suggest a slight improvement in the normality, likely due to the logarithmic transformation decreasing the values, but without removing observations it is difficult to correct this entirely.

Fascinatingly, we did notice some evidence of outliers using the ggplot package and investigating the relationships between our predictors and response variable.

In the relationship between distance and average speed, we can observe a decreasing linear trend for Leah but there are a large handful of points in the bottom left corner that don’t seem to fit the trend at all. We hypothesize some of these observations may have been walks recorded by the watch instead of jogs.

While for the other predictors, the trends are a bit more obvious. Below we see the relationship between average heart rate and average speed. There is an obvious increasing trend, possibly quadratic for Leah. The trend is less obvious for Adam, but still generally increasing.



Finally for average cadence, we can observe an obvious positive trend for both runners. The trend appears to be possibly quadratic or exponential for Leah.

