Abalone dataset: Predicting the Ring Age in Years

Filip Reierson

25/09/2021

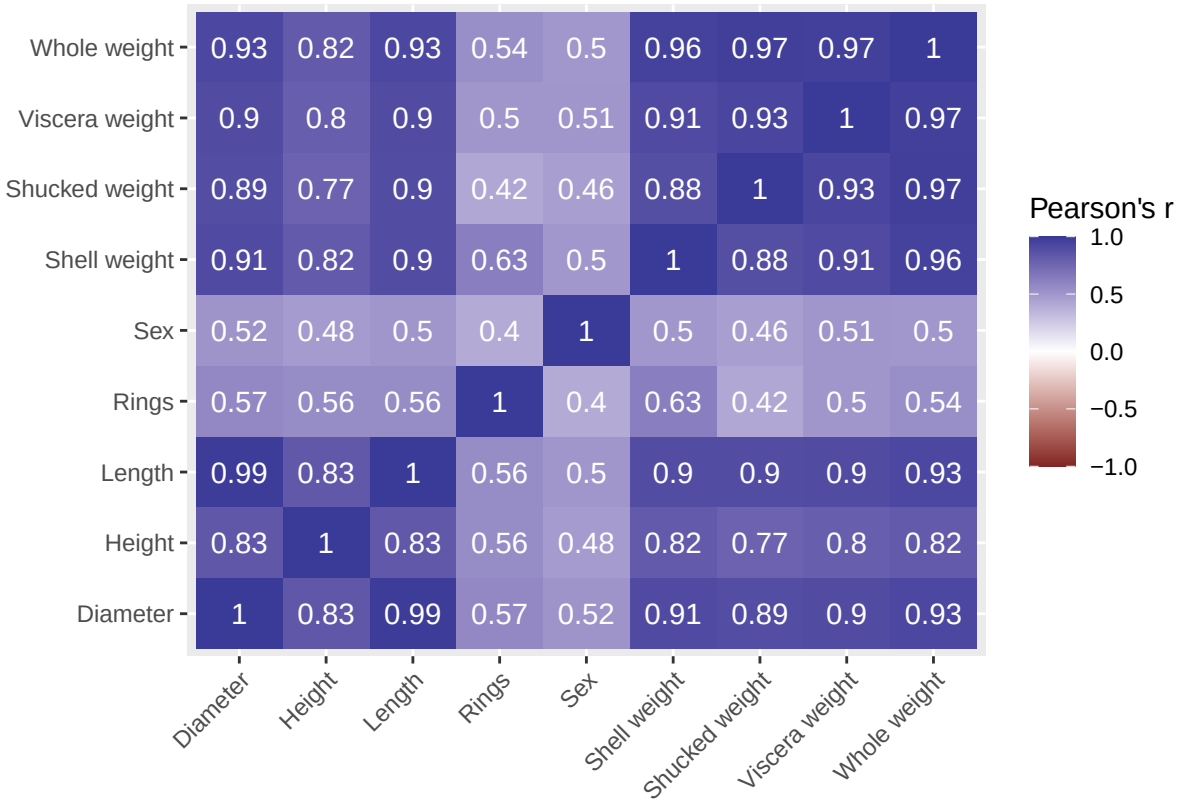
Contents

Data processing

The nine attributes of the Abalone dataset.

Name	Data Type	Meas.	Description
Sex	nominal	–	M, F, and I (infant)
Length	continuous	mm	Longest shell measurement
Diameter	continuous	mm	perpendicular to length
Height	continuous	mm	with meat in shell
Whole weight	continuous	grams	whole abalone
Shucked weight	continuous	grams	weight of meat
Viscera weight	continuous	grams	gut weight (after bleeding)
Shell weight	continuous	grams	after being dried
Rings	integer	–	+1.5 gives the age in years

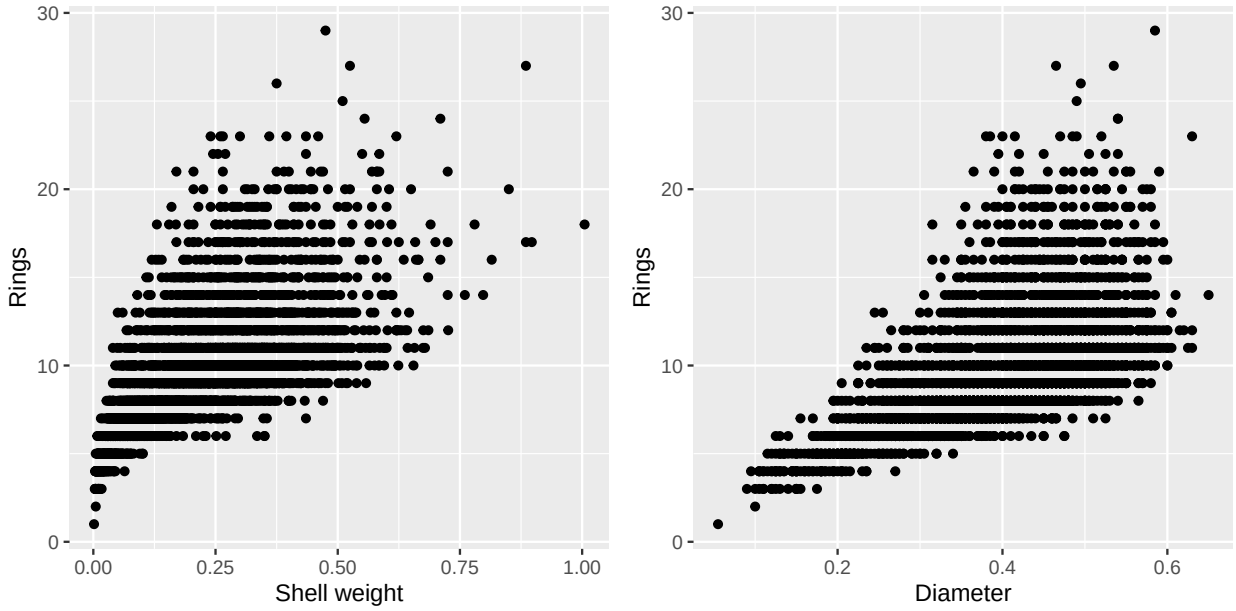
We begin by replacing Sex values I, M, and F by -1, 0, and 1 respectively.



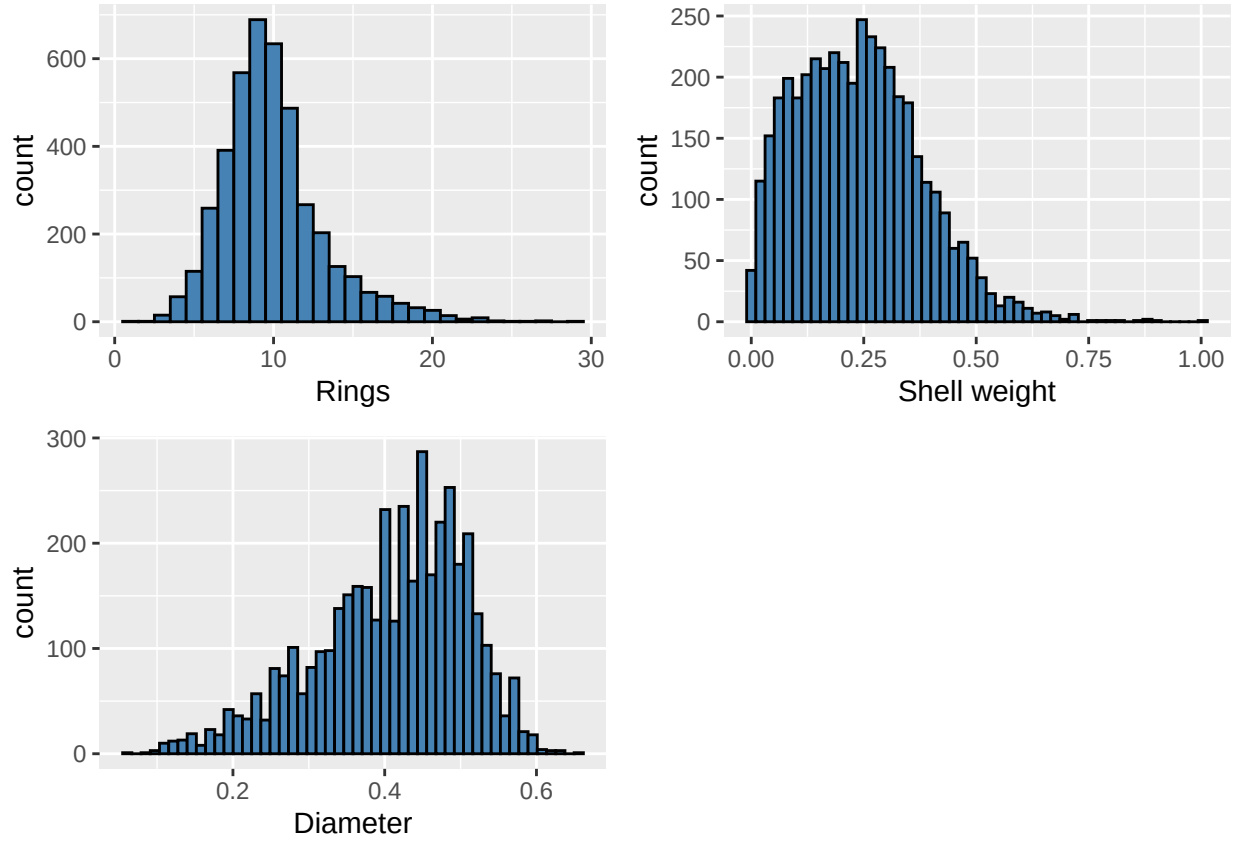
On the correlation map we can observe that correlation between rings and other features ranges from 0.4 to 0.63 all in the positive direction. The sex correlation can't be meaningfully interpreted here as it is a nominal variable. One hot encoding, would be appropriate, but is outside the scope. We can also see that the various measures of weight are strongly correlated, as we might expect. We can also see that Length and Diameter is very highly correlated, and as a result would make coefficient interpretation problematic if they are both included in the model. We can also read off the correlation plot which features are most correlated with ring-age.

The features most correlated with ring-age are,

Feature	r
Shell weight	0.63
Diameter	0.57

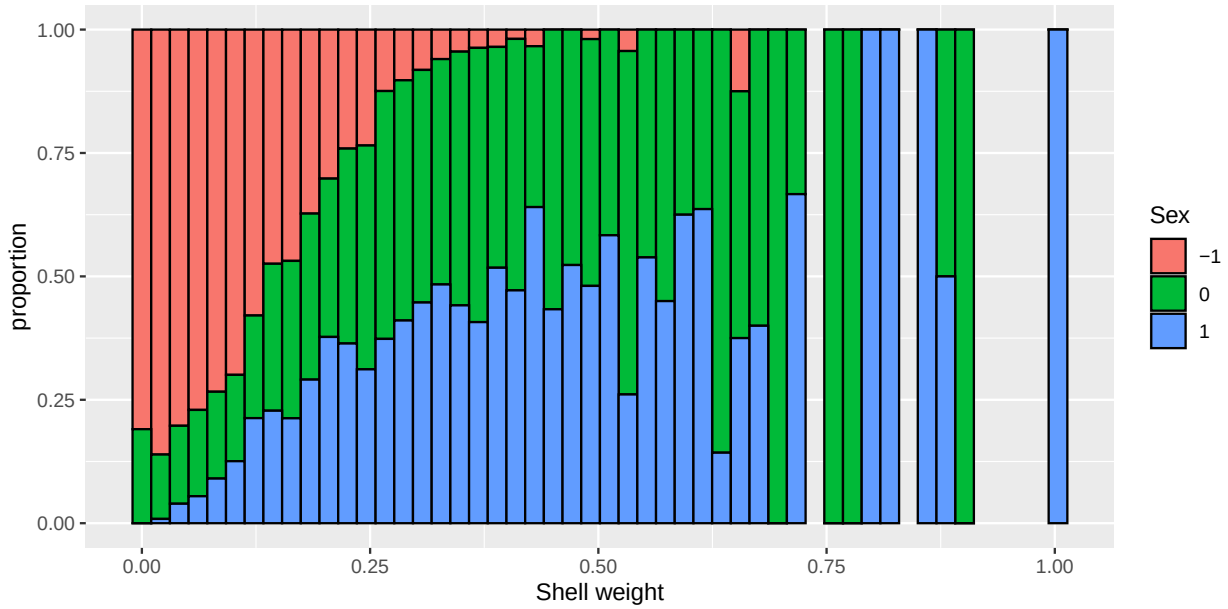


In the above plot we can see that shell weight doesn't appear to have a strictly linear relationship with rings, but there is clearly a strong association. Diameter appears to have a more linear looking relationship for low values, but has a concave up shape for later values. Both relationships appear to have fairly large heteroscedasticity in the form of fanning, suggesting estimations of ring age may worsen as number of rings increases.



Rings appear to be fairly symmetrically distributed, perhaps with a small right skew. Shell weight is right skewed, and may be bimodal, although it is not entirely clear from this plot. Diameter is highly left skewed.

We can confirm shell weight is bimodal with the following stacked histogram.



A better model may use sex in the model, perhaps with an interaction effect, however, this is outside the report's scope.

With that said I will move on to modelling.

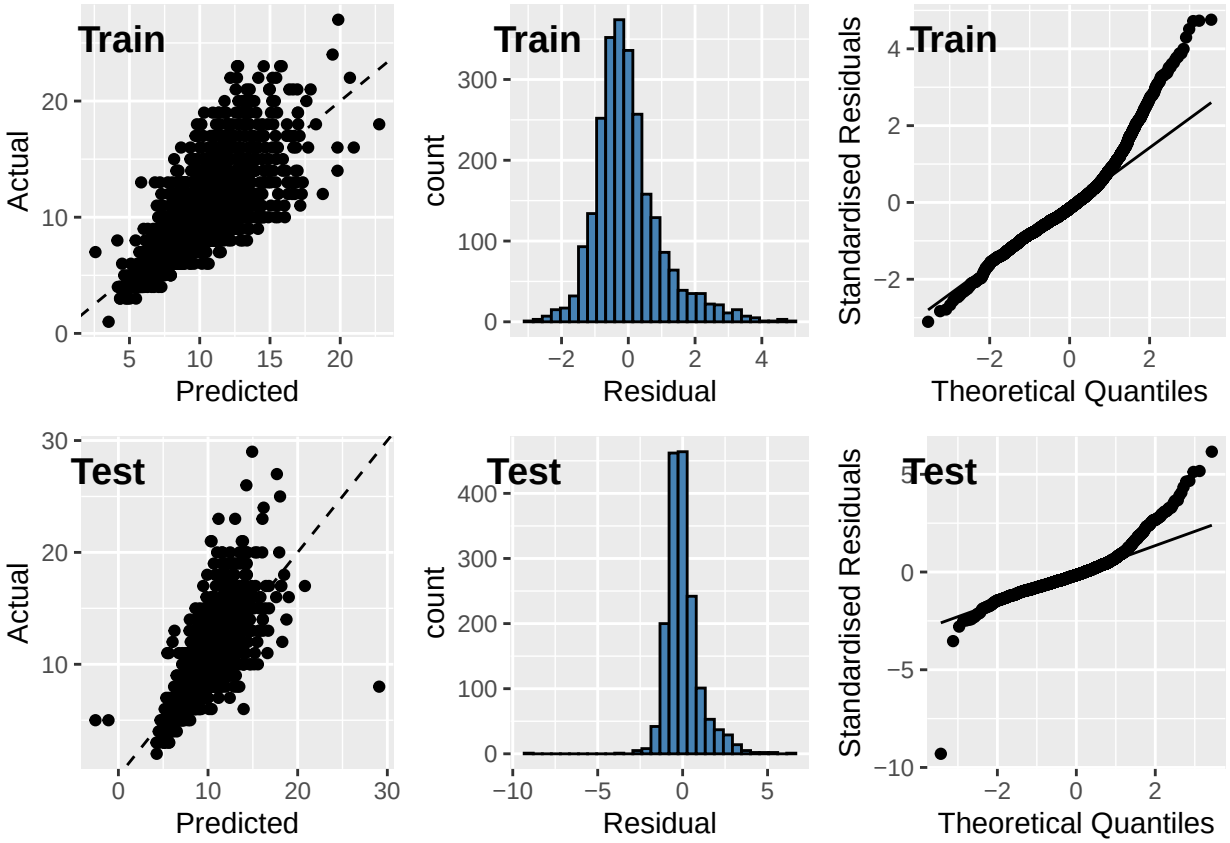
Modelling

Full linear model non-normalised

Fitting the full model gives coefficients:

term	estimate	std.error	statistic	p.value
(Intercept)	3.2675	0.3528	9.2613	0.0000
Sex	0.4381	0.0643	6.8144	0.0000
Length	1.2865	2.2613	0.5689	0.5695
Diameter	6.8514	2.7927	2.4533	0.0142
Height	21.0714	2.6815	7.8582	0.0000
`Whole weight`	8.6150	0.9320	9.2431	0.0000
`Shucked weight`	-19.3679	1.0639	-18.2050	0.0000
`Viscera weight`	-10.4631	1.6958	-6.1700	0.0000
`Shell weight`	8.1397	1.4181	5.7396	0.0000

We can see that Diameter and Length explain much of the same effect as expected from the correlation plot. Also note that Sex is not one hot encoded, so interpretation is difficult. Length, height, and diameter are all associated with ring-years. Different measures of weight have different associations with ring-years when considering all features. Whole weight and shell weight are positively associated with ring-years, while shucked weight and viscera weight are negatively associated with ring-years.



The actual vs predicted plot indicate that a linear model isn't great, but explains some of the variation in ring-years. The qq-plot indicate that normality of residuals is not an appropriate assumption. However, since this is a predictive exercise rather than explanatory it is not problematic as long as the model has good predictive power on the test set. However, looking at the test set we see that at least one residual is way out and in general the estimates are poor for higher quantiles.

Set	R squared	RMSE
Train	0.538	2.174
Test	0.514	2.276

The R^2 value indicates that 51.4% of the ring-year's variability in the test set was explained by the model.

Full model normalised

term	estimate	std.error	statistic	p.value
(Intercept)	3.7456	0.2956	12.6721	0.0000
Sex	0.4381	0.0643	6.8144	0.0000
Length	0.9520	1.6734	0.5689	0.5695
Diameter	4.0766	1.6617	2.4533	0.0142
Height	23.8107	3.0300	7.8582	0.0000
`Whole weight`	24.3244	2.6316	9.2431	0.0000
`Shucked weight`	-28.8000	1.5820	-18.2050	0.0000
`Viscera weight`	-7.9467	1.2880	-6.1700	0.0000

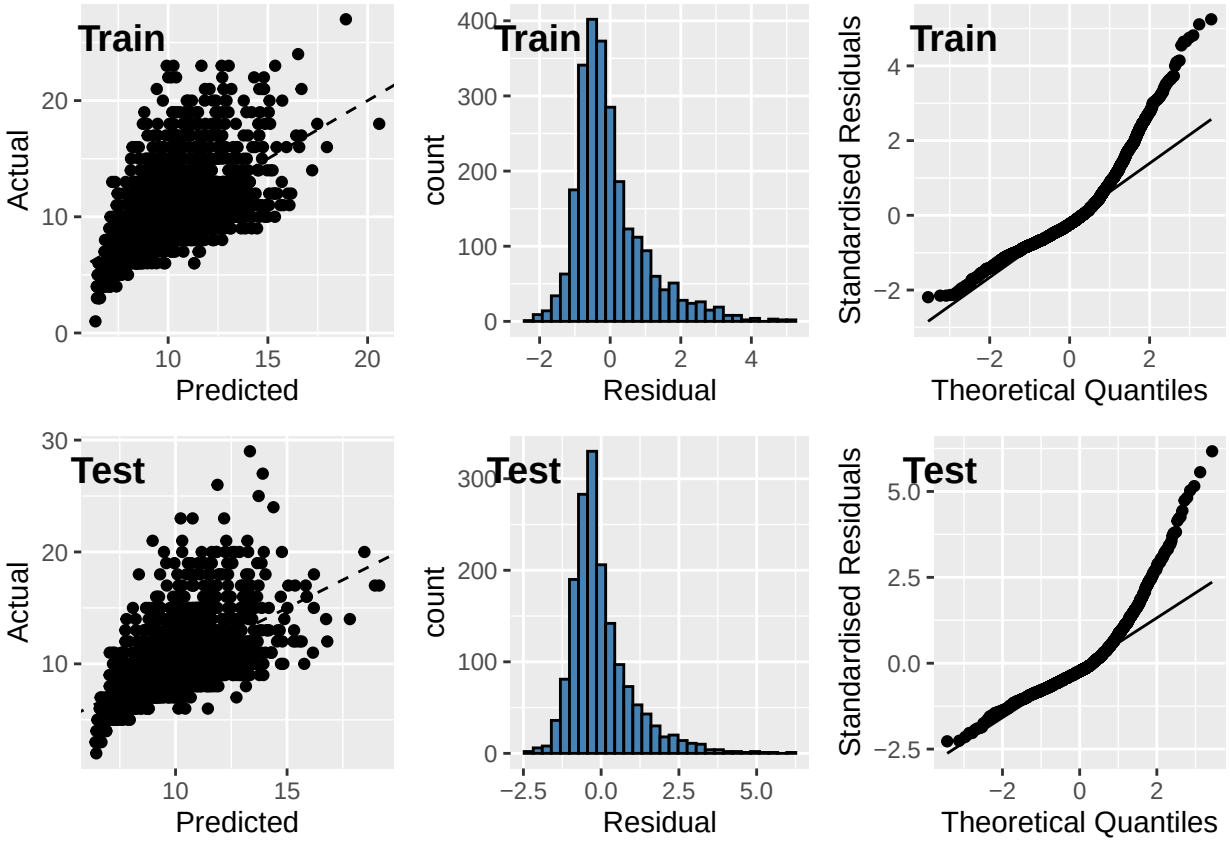
term	estimate	std.error	statistic	p.value
`Shell weight`	8.1682	1.4231	5.7396	0.0000

Set	R squared	RMSE
Train	0.538	2.174
Test	0.514	2.276

We observe that while the normalisation changes the coefficients of our model it does not affect its ability to predict. This is because ordinary least squares is scale invariant, so a linear transformation doesn't impact it.

Linear model two features

Recall we found that shell weight and diameter had the highest correlation with rings, so we will develop a model using only those features as predictors.



term	estimate	std.error	statistic	p.value
(Intercept)	6.2849	0.3106	20.2338	0.000
`Shell weight`	13.7151	0.8291	16.5421	0.000
Diameter	0.9149	1.1641	0.7860	0.432

Set	R squared	RMSE
Train	0.394	2.490
Test	0.395	2.541

While the metrics indicate a worse fit than the full model it is worth noting that this model appears to generalise better to the test set. Looking at the test set qq-plots we don't see the same deviation from normality for lower quantiles as we saw in the full model. However, Diameter doesn't add meaningfully to the model as indicated by the p-value, so a more parsimonious model wouldn't include diameter.

term	estimate	std.error	statistic	p.value
(Intercept)	6.3558	0.2515	25.2691	0.000
`Shell weight`	13.7631	0.8320	16.5421	0.000
Diameter	0.5444	0.6926	0.7860	0.432

Set	R squared	RMSE
Train	0.394	2.490
Test	0.395	2.538

As before, normalising doesn't affect our metrics.

Sensitivity analysis

The follow table shows aggregate statistics from the full model being fitted 30 times with randomised splits.

Set	Statistic	Mean	SD
Test	R squared	0.525	0.018
Test	RMSE	2.240	0.051
Train	R squared	0.533	0.010
Train	RMSE	2.191	0.030

And the following shows the aggregate statistics from the full model being fitted 30 times in the same manner, but with the normalised inputs.

Set	Statistic	Mean	SD
Test	R squared	0.525	0.018
Test	RMSE	2.240	0.051
Train	R squared	0.533	0.010
Train	RMSE	2.191	0.030

Again we observe that the metrics are identical between normalised and non-normalised, due to the invariance properties of the estimates. We would observe a different result if we didn't use the same seeds for the experiments (1 to 30).

The following table was calculated by running the two feature model on 30 random 60/40 splits.

Set	Statistic	Mean	SD
Test	R squared	0.396	0.015
Test	RMSE	2.526	0.050
Train	R squared	0.393	0.010
Train	RMSE	2.499	0.033

Here we see that the standard deviation of both R^2 and RMSE are marginally lower for the test set, while standard deviation of R squared in the training set is almost identical in both models. The training set RMSE is marginally higher in the two feature model. The two feature model also maintains its R squared on the test set while the full model's R squared is marginally lower on the test data, this suggests the simpler model generalises slightly better. The full model explains about 13% more of the variation in ring-years than the two feature model.

The follow table shows aggregate statistics from the two feature model using normalised inputs being fitted 30 times with randomised splits.

Set	Statistic	Mean	SD
Test	R squared	0.396	0.015
Test	RMSE	2.526	0.050
Train	R squared	0.393	0.010
Train	RMSE	2.499	0.033

Again we observe the invariance property.

Conclusion

Neither model appears appears to be ideal. What I would look at for future analyses:

1. Using one hot encoding to deal with Sex (since there are 3 categories)
2. Interaction effects.
3. Transformations that better deal with the non-linear residuals.
4. Hierarchical clustering by variable to help select better features for the model.

References

Other than the R packages imported I did not use any existing code. R packages used: tidyverse, knitr, cowplot, and broom.