Does Size Matter? (Estimation of Banana Weight with a Regression Modeling Approach)

Scott Graham, Kaisa Roggeveen

February 13, 2018

Summary

Introduction

The purpose of this study was to determine the most effective regression model to predict the weight of a banana using external measurements. This study also demonstrated multiple techniques for developing regression models. These models were then examined to demonstrate their effectiveness at creating regression models.

Data Collection

First a small sample set bananas were purchased from the Real Canadian Superstore. The weight, length, diameter and circumference were then calculated using a scale and a ruler.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
0.0004513	0.4005	0.6352	0.5719	0.7617	0.9812

In order to determine the minimum sample size needed, random sample sizes of 10 were generated using radius and length as the predictors. The correlation of the random sample sizes were calculated and a matrix of the correlations were generated. The value of the squared population multiple correlation coefficients with two predictor variables was then calculated and determined to be approximately 0.5719. From this the minimum sample size required was then determined from the table from Gregory T. Knofczynski's Sample Size When Using Multiple Linear Regression for Prediction, the minimum sample size was determined to be between 15 and 35, therefore the minimum number of bananas required was finalized at 24 bananas.

Analysis

To begin analysis a model using all predictor variables was created. In this case the density of the banana is assumed to be a constant.

Let:

$$W = \text{Weight (g)},\ L = \text{Length (mm)},\ R = \text{Radius (mm)}$$

Then:

$$\log(W) = \beta_0 + \beta_1 \log(L) + \beta_2 \log(R) + \beta_3 \log(C) \implies W = e^{\beta_0} \times L^{\beta_1} \times R^{\beta_2} \times C^{\beta_3}$$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	10.06	26.32	0.3822	0.7063
Length_log	0.123	0.1275	0.9652	0.346

	Estimate	Std. Error	t value	Pr(>|t|)
Radius_log	7.526	14.09	0.5341	0.5992
Circumference_log	-5.788	14.16	-0.4088	0.687

Table 3: Fitting linear model: Weight_log ~ Length_log + Radius_log + Circumference_log This returned values with insignificant p-values

Observations	Residual Std. Error	R^2	Adjusted R^2
24	0.09248	0.3318	0.2316

In the second model the predictor variable, circumference, was removed. This is because $C = 2\pi R$.

$$\log(W) = \beta_0 + \beta_1 \log(L) + \beta_2 \log(R)$$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-0.6702	1.913	-0.3503	0.7296
Length_log	0.1223	0.1249	0.9787	0.3389
Radius_log	1.77	0.5596	3.163	0.004684

Table 5: Fitting linear model: Weight_log ~ Length_log + Radius_log

Observations	Residual Std. Error	R^2	Adjusted R^2
24	0.09062	0.3262	0.2621

The third model considered the predictor, length.

$$\log(W) = \beta_0 + \beta_1 \log(L)$$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	4.99	0.8037	6.209	3e-06
Length_log	0.04917	0.1457	0.3374	0.739

Table 7: Fitting linear model: Weight_log ~ Length_log

Observations	Residual Std. Error	R^2	Adjusted R^2
24	0.1076	0.005146	-0.04007

The fourth model considered only one predictor, radius.

$$\log(W) = \beta_0 + \beta_2 \log(R)$$

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	0.3046	1.632	0.1867	0.8536
Radius_log	1.669	0.5494	3.038	0.006043

Table 9: Fitting linear model: Weight_log ~ Radius_log

Observations	Residual Std. Error	R^2	Adjusted R^2
24	0.09054	0.2955	0.2635

Table 10: foo

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
21	0.1725	NA	NA	NA	NA
20	0.171	1	0.001429	0.1671	0.687

Table 11: Analysis of Variance Table

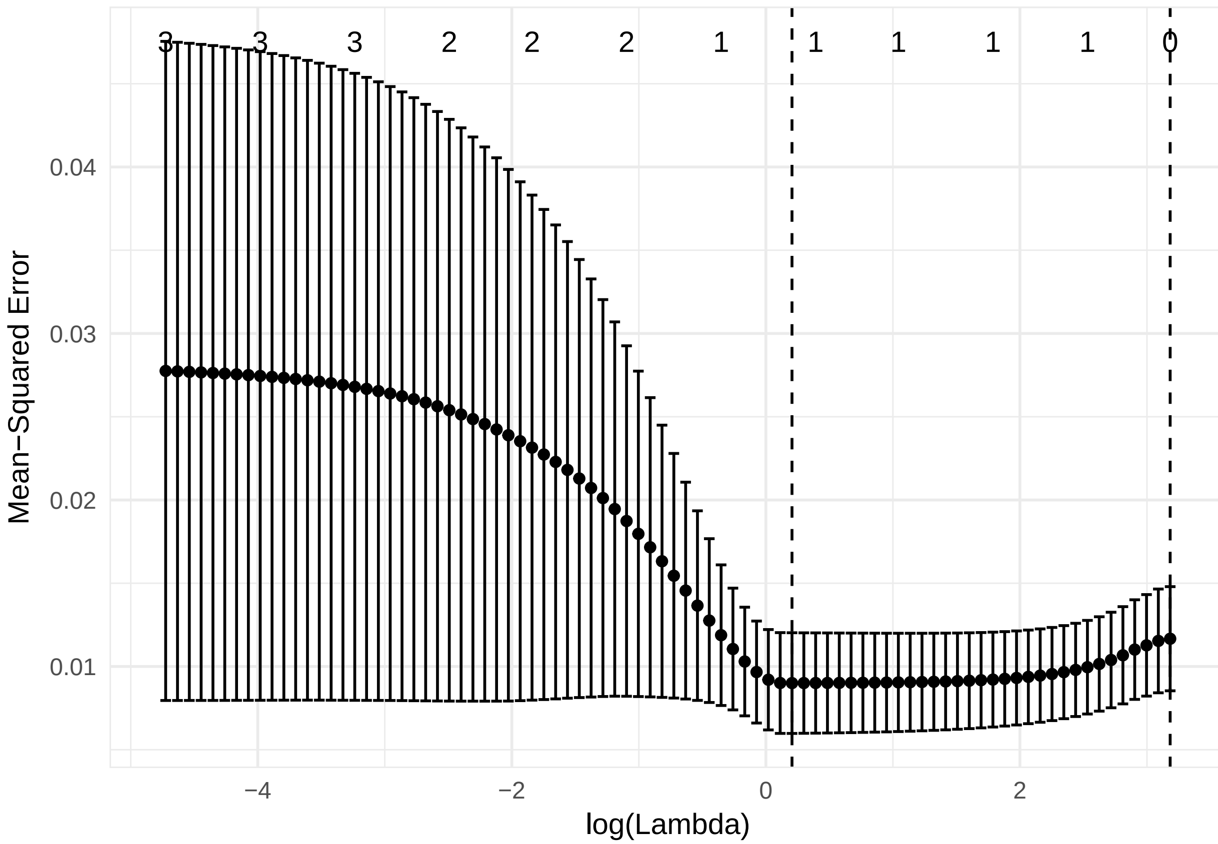
Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
22	0.2547	NA	NA	NA	NA
20	0.171	2	0.08362	4.889	0.01868

Table 12: Analysis of Variance Table

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
22	0.2547	NA	NA	NA	NA
21	0.1725	1	0.08219	10.01	0.004684

```
## Warning: Option grouped=FALSE enforced in cv.glmnet, since < 3 observations
## per fold
```

```
## Warning: Option grouped=FALSE enforced in cv.glmnet, since < 3 observations
## per fold
```



	1
(Intercept)	0.55713
Length	0.00000
Radius	0.00000
Circumference	0.00000
Length_log	0.00000
Radius_log	1.58392
Circumference_log	0.00000

	1
(Intercept)	5.261471
Length	0.000000
Radius	0.000000
Circumference	0.000000
Length_log	0.000000
Radius_log	0.000000
Circumference_log	0.000000

Recommendations

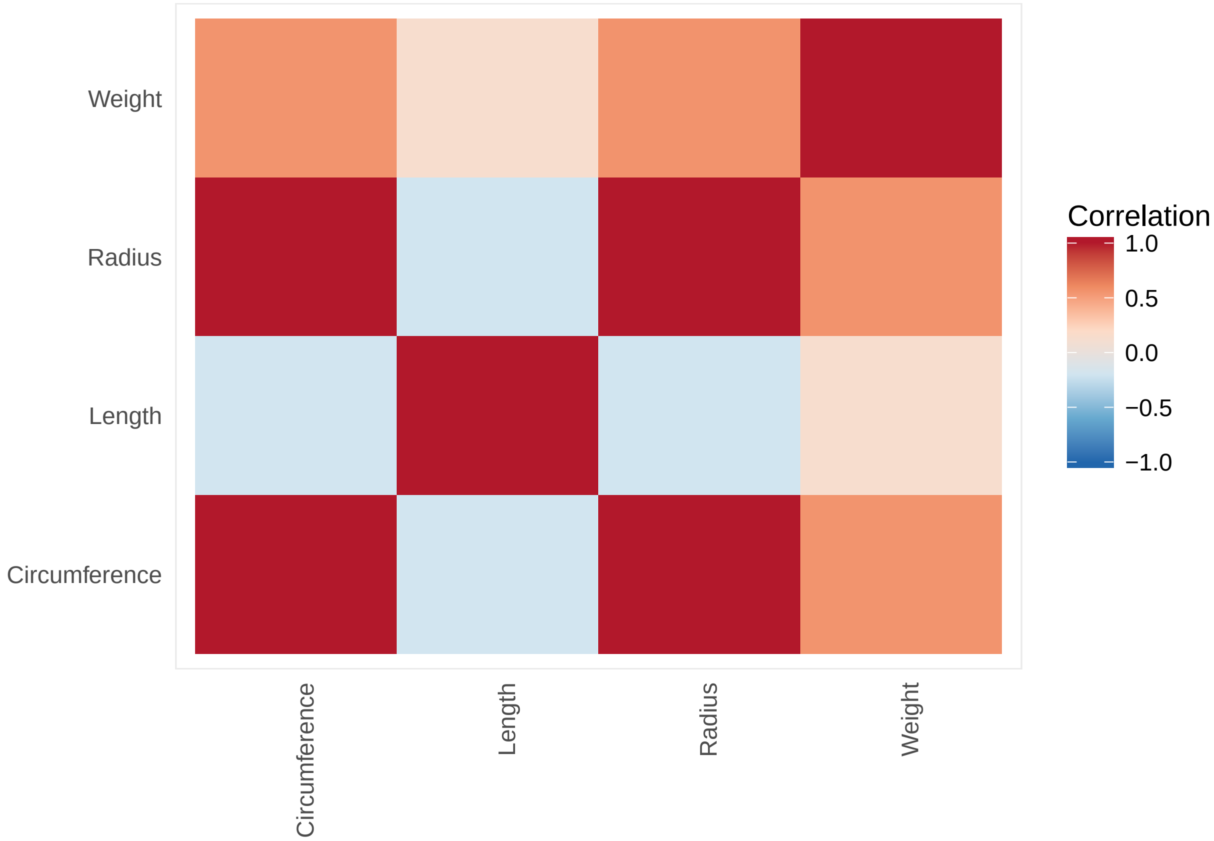
It can be determiend that the best way to predict the weight of a banana is by measuring the radius of the banana. The model that is then used for banana weight prediction is the following:

$$\log(W) = \beta_0 + \beta_1 \log(R)$$

$$\log(W) = 0.3046 + 1.669 \log(R)$$

The predictor variable radius, was more significantly correlated to the weight in comparison to circumference and length.

Appendix

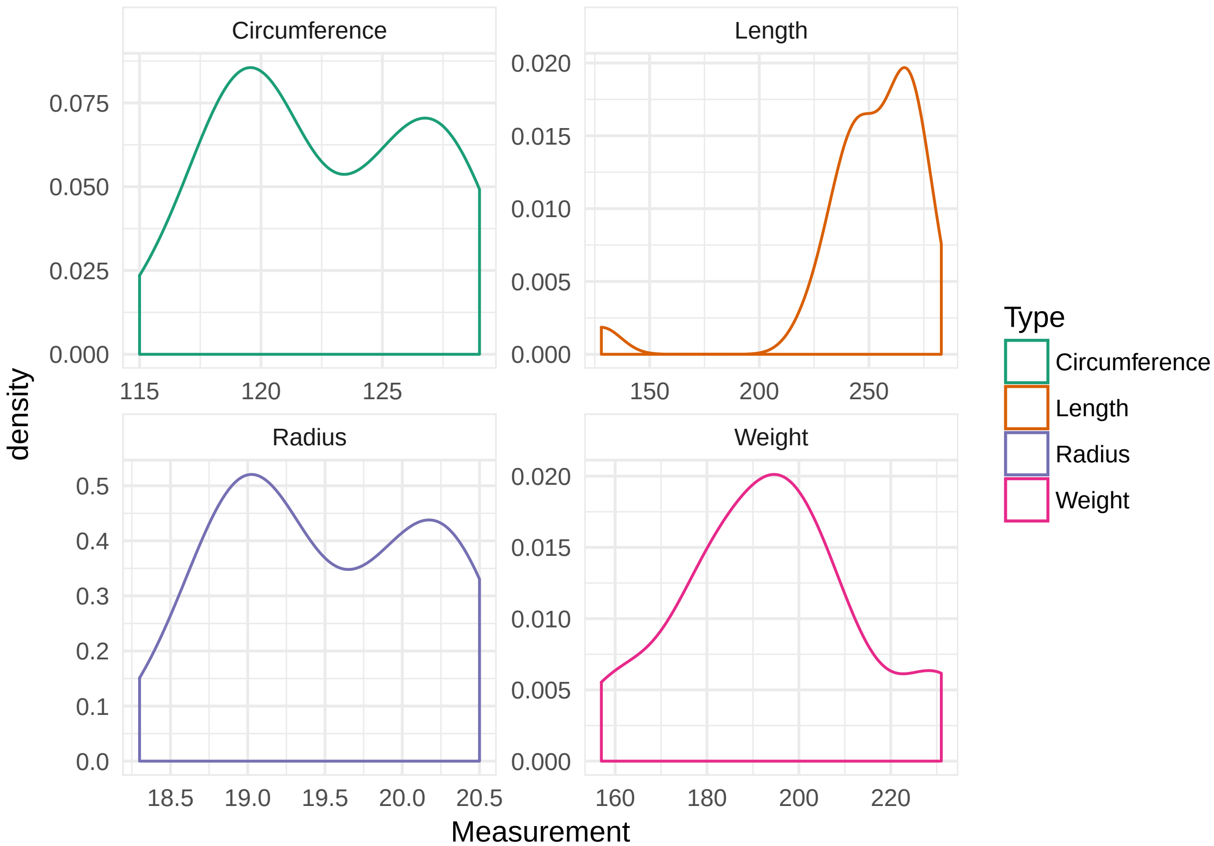


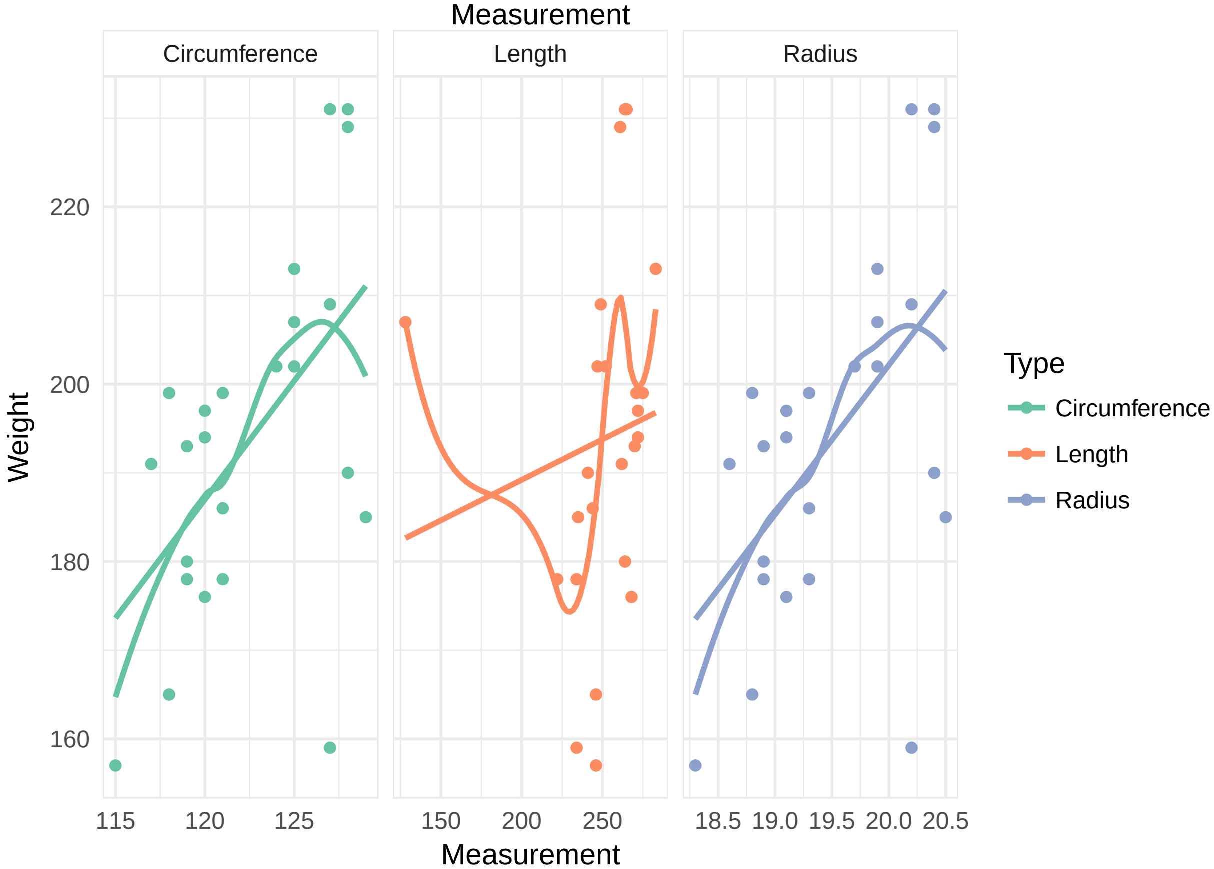

```
## Warning: Computation failed in `stat_bin()`:
## is.numeric(width) is not TRUE

## Warning: Computation failed in `stat_bin()`:
## is.numeric(width) is not TRUE

## Warning: Computation failed in `stat_bin()`:
## is.numeric(width) is not TRUE

## Warning: Computation failed in `stat_bin()`:
## is.numeric(width) is not TRUE
```


Circumference
Length
Radius
Weight
density
Measurement
Type
Circumference
Length
Radius
Weight
0.075
0.050
0.025
0.000
115
120
125
0.020
0.015
0.010
0.005
0.000
150
200
250
0.5
0.4
0.3
0.2
0.1
0.0
18.5
19.0
19.5
20.0
20.5
0.020
0.015
0.010
0.005
0.000
160
180
200
220


Circumference
Length
Radius
Weight
Measurement
Type
Circumference
Length
Radius
220
200
180
160
115
120
125
150
200
250
18.5
19.0
19.5
20.0
20.5

Cross Validation

```
## Analysis of Variance Table
##
## Response: Weight_log
##            Df Sum Sq Mean Sq F value Pr(>F)
## Length_log  1 0.0013  0.0013    0.16 0.6928
## Radius_log  1 0.0822  0.0822   10.01 0.0047 **
## Residuals  21 0.1725  0.0082
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## fold 1
## Observations in test set: 8
##               [,1]   [,2]   [,3]    [,4]   [,5]    [,6]   [,7]   [,8]
## Predicted   5.2368 5.2368  5.196  5.1998 5.3143  5.2304  5.344  5.215
## cvpred      5.2428 5.2428  5.207  5.2131 5.3184  5.2464  5.358  5.222
## Weight_log  5.2832 5.2679  5.106  5.1818 5.3613  5.1818  5.220  5.193
## CV residual 0.0404 0.0251 -0.101 -0.0314 0.0429 -0.0646 -0.137 -0.029
##
## Sum of squares = 0.04    Mean square = 0    n = 8
##
## fold 2
## Observations in test set: 8
##                 [,1]  [,2]  [,3]    [,4]   [,5]    [,6]   [,7]   [,8]
## Predicted    5.24195 5.348 5.333  5.2350 5.2548  5.3386 5.1853 5.3251
## cvpred       5.23389 5.324 5.310  5.2248 5.2417  5.3184 5.1822 5.3056
## Weight_log   5.22575 5.434 5.442  5.1705 5.2933  5.2470 5.2523 5.3423
## CV residual -0.00814 0.109 0.132 -0.0543 0.0516 -0.0714 0.0701 0.0367
##
## Sum of squares = 0.05    Mean square = 0.01    n = 8
##
## fold 3
## Observations in test set: 8
##             [,1]   [,2]    [,3]   [,4]   [,5]   [,6]   [,7]   [,8]
## Predicted   5.22  5.318  5.1488 5.3497 5.2822 5.2976 5.2173 5.2101
## cvpred      4.78  5.267  5.1137 5.3771 5.2744 5.2788 5.2464 5.2500
## Weight_log  5.33  5.069  5.0562 5.4424 5.3083 5.3083 5.2627 5.2933
## CV residual 0.55 -0.198 -0.0574 0.0654 0.0339 0.0295 0.0163 0.0433
##
## Sum of squares = 0.35    Mean square = 0.04    n = 8
##
## Overall (Sum over all 8 folds)
##     ms
## 0.0183
```

```
## # A tibble: 24 x 13
##       ID Weight Radius Length Circumference Weight_log Radius_log
##    <int>  <int>  <dbl>  <int>         <int>      <dbl>      <dbl>
##  1     1    197   19.1    272           120       5.28       2.95
##  2     2    194   19.1    272           120       5.27       2.95
##  3     3    165   18.8    246           118       5.11       2.93
##  4     4    186   19.3    244           121       5.23       2.96
```

```
## 5     5    178   18.9    234            119       5.18       2.94
## 6     6    207   19.9    128            125       5.33       2.99
## 7     7    213   19.9    283            125       5.36       2.99
## 8     8    178   19.3    222            121       5.18       2.96
## 9     9    229   20.4    261            128       5.43       3.02
## 10    10    231   20.2    265            127       5.44       3.01
## # ... with 14 more rows, and 6 more variables: Length_log <dbl>,
## #   Circumference_log <dbl>, Predicted <dbl>, cvpred <dbl>, `CV
## #   Residual` <dbl>, Residual <dbl>
```

MAE

```
## # A tibble: 1 x 2
##     MAE   MPAE
##   <dbl>  <dbl>
## 1  13.7 0.0722
```