Stats 101A Project: Group Project First Draft

Simran Vatsa, Naren Akurati, Sohom Paul, Ashwin Ayyasamy, Jeremy Phan

3/16/2018

Data Cleanup

Consulted codebook to decide which codes could be converted to NAs. Changed "not answered" to NA, as that is information we do not have. Also converted variables coded to denote "_ or more" to NAs, as that is information we do not have and cannot create. We did not convert 8 (8 or more) in Household or Children, and we converted 0 (Inapplicable) and 8 (Don't know) to 2 (No) in Instagram.

```
## Loading required package: car
```

```
## Warning: package 'car' was built under R version 3.4.3
```

First Model

We start with the full model with everything except *Health* and *WorkHrs* predictors. *Health* and *WorkHrs* predictors throw errors due to large number of NAs (811 and 1898 NAs respectively) and few categories. R^2 currently at 0.2811438 and R^2_{adj} at 0.2069141.

```r
attach(happiness_data)
# Factoring Categorical Variables
JobSat.f <- factor(JobSat)
OwnHome.f <- factor(OwnHome)
Marital.f <- factor(Marital)
Instagram.f <- factor(Instagram)
Health.f <- factor(Health)
Household.f <- factor(Household)
Children.f <- factor(Children)
Sex.f <- factor(Sex)

# Couldn't include Health as it throws an error
full_model <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f +
    Marital.f + Children.f + Education + JobSat.f + Income +
    Age + Sex.f)

sum(is.na(Health))
```

```
## [1] 811
```

```r
sum(is.na(WorkHrs))
```

```
## [1] 1898
```

```r
summary(full_model)$r.squared
```

```
## [1] 0.317573
```

```r
summary(full_model)$adj.r.squared
```

```
## [1] 0.2038351
```

Transformation

Transformation of numerical variables *Education*, *Income*, and *Age* using powertransform. Understanding of the effects of wealth lead us to use log transformation of *Income* predictor which proved more effective than the estimated transformation parameter. Inverse rseponse plot suggested lambda close to 0. As such, we took log(*Happy*) for a simpler model. R^2 currently at 0.3518696 and R^2_{adj} at 0.2438479.

```r
# Power transformation
powerTransform(cbind(Household.f, OwnHome.f, Instagram.f, Marital.f,
    Children.f, Education, JobSat.f, Income, Age, Sex.f) ~ 1)
```

```
## Estimated transformation parameters
## Household.f   OwnHome.f Instagram.f   Marital.f  Children.f   Education
##  -0.2139927  -1.9783865   6.3174335   0.2855726  -0.1489214   0.7943619
##    JobSat.f      Income         Age       Sex.f
##   0.1400369   0.2140292   0.3192108  -1.2017747
```

```r
Education_transformed <- Education^0.7943619
Income_transformed <- Income^0.2140292
Income_log <- log(Income)
Age_transformed <- Age^0.3192108

full_model_transform_log <- lm(Happy ~ Household.f + OwnHome.f +
    Instagram.f + Marital.f + Children.f + Education_transformed +
    JobSat.f + Income_log + Age_transformed + Sex.f)

summary(full_model_transform_log)$r.squared
```

```
## [1] 0.3211593
```

```r
summary(full_model_transform_log)$adj.r.squared
```

```
## [1] 0.2080192
```

```r
# Inverse response plot
par(mfrow = c(2, 2))
inverseResponsePlot(full_model_transform_log, key = TRUE)
```

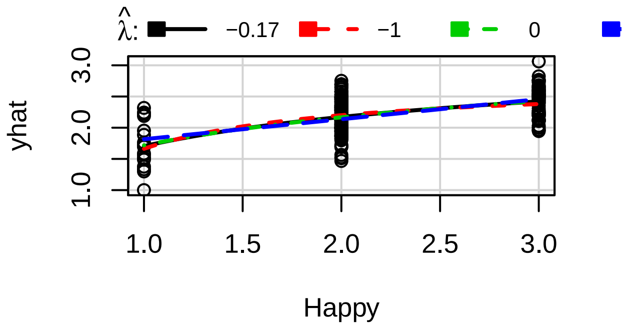
```
##       lambda      RSS
## 1 -0.1711872 15.37896
## 2 -1.0000000 15.61674
## 3  0.0000000 15.39022
## 4  1.0000000 15.86280
```

```r
full_model_transform_log_inverse_response <- lm(log(Happy) ~
    Household.f + OwnHome.f + Instagram.f + Marital.f + Children.f +
        Education_transformed + JobSat.f + Income_log + Age_transformed +
        Sex.f)
summary(full_model_transform_log_inverse_response)$r.squared
```

```
## [1] 0.3518696
```

```r
summary(full_model_transform_log_inverse_response)$adj.r.squared
```

```
## [1] 0.2438479
```



Cursory Variable Selection

We look at number of NAs in our predictors. *OwnHome*, *JobSat*, and *Income* all have a high number of NAs (812, 1612, and 1039 respectively). From summary, predictors showing p-values over 0.05 are *OwnHome*, *Instagram*, *Marital*, *Children*, *Education*, and *Age*. These may need to be removed

```
##    c.sum.is.na.Household.f....sum.is.na.OwnHome.f....sum.is.na.Instagram.f....
## 1                                                                            1
## 2                                                                          812
## 3                                                                           10
## 4                                                                            1
## 5                                                                            6
## 6                                                                            8
## 7                                                                         1612
## 8                                                                         1039
## 9                                                                           29
## 10                                                                           0
```

```
##
## Call:
## lm(formula = log(Happy) ~ Household.f + OwnHome.f + Instagram.f +
##     Marital.f + Children.f + Education_transformed + JobSat.f +
##     Income_log + Age_transformed + Sex.f)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.79208 -0.15001  0.00154  0.16687  0.53956
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            0.8806882  0.3518862   2.503 0.013245 *
## Household.f2          -0.1016777  0.0583719  -1.742 0.083294 .
## Household.f3          -0.0389791  0.0811533  -0.480 0.631608
## Household.f4          -0.3471793  0.1246535  -2.785 0.005943 **
## Household.f5          -0.2595818  0.2069948  -1.254 0.211506
## Household.f6          -0.4162874  0.1986458  -2.096 0.037562 *
## OwnHome.f2             0.0244404  0.0474199   0.515 0.606924
## OwnHome.f3             0.2653555  0.2116575   1.254 0.211632
## Instagram.f2          -0.0231741  0.0514444  -0.450 0.652934
## Marital.f2            -0.3942993  0.1149816  -3.429 0.000756 ***
## Marital.f3            -0.1778557  0.0677965  -2.623 0.009480 **
## Marital.f4            -0.1250419  0.1314115  -0.952 0.342655
## Marital.f5            -0.1839283  0.0687317  -2.676 0.008162 **
```

```
## Children.f1           -0.0394101  0.0663705  -0.594 0.553424
## Children.f2            0.0356377  0.0680488   0.524 0.601148
## Children.f3           -0.0028242  0.0677348  -0.042 0.966790
## Children.f4            0.0396402  0.1033202   0.384 0.701697
## Children.f5           -0.0001376  0.1368955  -0.001 0.999199
## Children.f7           -0.1363095  0.2765553  -0.493 0.622716
## Children.f8           -0.5796678  0.3036189  -1.909 0.057883 .
## Education_transformed  0.0156236  0.0185100   0.844 0.399794
## JobSat.f2             -0.0629262  0.0587428  -1.071 0.285556
## JobSat.f3             -0.1711084  0.0629903  -2.716 0.007266 **
## JobSat.f4             -0.1061410  0.1053511  -1.007 0.315095
## JobSat.f5             -0.3888175  0.0907819  -4.283 3.04e-05 ***
## JobSat.f6             -0.5148107  0.1325893  -3.883 0.000146 ***
## JobSat.f7             -0.0814809  0.2829385  -0.288 0.773704
## Income_log             0.0437128  0.0215905   2.025 0.044434 *
## Age_transformed       -0.1187717  0.0798401  -1.488 0.138661
## Sex.f2                 0.0756915  0.0446454   1.695 0.091790 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2664 on 174 degrees of freedom
##   (2163 observations deleted due to missingness)
## Multiple R-squared:  0.3519, Adjusted R-squared:  0.2438
## F-statistic: 3.257 on 29 and 174 DF,  p-value: 8.207e-07
```

Partial F-tests

We start with manual F-tests based on backward selection (removing the least significant variables first each iteration).

```r
drop1(full_model_transform_log_inverse_response, test = "F")
```

```
## Single term deletions
##
## Model:
## log(Happy) ~ Household.f + OwnHome.f + Instagram.f + Marital.f +
##     Children.f + Education_transformed + JobSat.f + Income_log +
##     Age_transformed + Sex.f
##                       Df Sum of Sq    RSS     AIC F value    Pr(>F)
## <none>                             12.352 -512.07
## Household.f            5   0.94817 13.300 -506.99  2.6713  0.023582 *
## OwnHome.f              2   0.13367 12.486 -513.88  0.9415  0.392039
## Instagram.f            1   0.01441 12.367 -513.84  0.2029  0.652934
## Marital.f              4   1.10845 13.461 -502.54  3.9035  0.004623 **
## Children.f             7   0.38638 12.739 -519.79  0.7775  0.606941
## Education_transformed  1   0.05058 12.403 -513.24  0.7124  0.399794
## JobSat.f               6   2.39551 14.748 -487.91  5.6241 2.332e-05 ***
## Income_log             1   0.29100 12.643 -509.32  4.0991  0.044434 *
## Age_transformed        1   0.15710 12.509 -511.49  2.2130  0.138661
## Sex.f                  1   0.20405 12.556 -510.73  2.8744  0.091790 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
drop1(update(full_model_transform_log_inverse_response, ~. -
    Instagram.f), test = "F")
```

```
## Single term deletions
##
## Model:
## log(Happy) ~ Household.f + OwnHome.f + Marital.f + Children.f +
##     Education_transformed + JobSat.f + Income_log + Age_transformed +
##     Sex.f
##                       Df Sum of Sq    RSS     AIC F value    Pr(>F)
## <none>                              12.367 -513.84
## Household.f            5   0.93989 13.307 -508.89  2.6601  0.024064 *
## OwnHome.f              2   0.13060 12.497 -515.69  0.9240  0.398837
## Marital.f              4   1.09482 13.461 -504.53  3.8732  0.004851 **
## Children.f             7   0.39769 12.764 -521.38  0.8039  0.585037
## Education_transformed  1   0.05260 12.419 -514.97  0.7443  0.389463
## JobSat.f               6   2.38407 14.751 -489.87  5.6228 2.325e-05 ***
## Income_log             1   0.29373 12.660 -511.05  4.1566  0.042978 *
## Age_transformed        1   0.20531 12.572 -512.48  2.9053  0.090062 .
## Sex.f                  1   0.21513 12.582 -512.32  3.0443  0.082776 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
drop1(update(full_model_transform_log_inverse_response, ~. -
    Instagram.f - Children.f), test = "F")
```

```
## Single term deletions
##
## Model:
## log(Happy) ~ Household.f + OwnHome.f + Marital.f + Education_transformed +
##     JobSat.f + Income_log + Age_transformed + Sex.f
##                       Df Sum of Sq    RSS     AIC F value   Pr(>F)
## <none>                              12.764 -521.38
## Household.f            5   1.42726 14.192 -509.76  4.0701 0.001586 **
## OwnHome.f              2   0.12205 12.886 -523.44  0.8701 0.420636
## Marital.f              4   1.34914 14.114 -508.88  4.8092 0.001037 **
## Education_transformed  1   0.11613 12.880 -521.53  1.6559 0.199799
## JobSat.f               6   2.37012 15.134 -498.63  5.6324 2.19e-05 ***
## Income_log             1   0.24500 13.009 -519.50  3.4933 0.063225 .
## Age_transformed        1   0.17165 12.936 -520.65  2.4474 0.119454
## Sex.f                  1   0.21448 12.979 -519.98  3.0581 0.082022 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
drop1(update(full_model_transform_log_inverse_response, ~. -
    Instagram.f - Children.f - OwnHome.f), test = "F")
```

```
## Single term deletions
##
## Model:
## log(Happy) ~ Household.f + Marital.f + Education_transformed +
##     JobSat.f + Income_log + Age_transformed + Sex.f
##                       Df Sum of Sq    RSS     AIC F value    Pr(>F)
## <none>                              55.685 -1411.4
## Household.f            6    1.4541 57.139 -1407.7  2.5548   0.01888 *
```

```
## Marital.f              4    1.0160 56.701 -1408.4  2.6776   0.03103 *
## Education_transformed  1    0.1343 55.819 -1411.9  1.4158   0.23458
## JobSat.f               6    3.8444 59.529 -1382.8  6.7544 6.191e-07 ***
## Income_log             1    0.3610 56.046 -1409.5  3.8060   0.05154 .
## Age_transformed        1    0.0265 55.711 -1413.1  0.2789   0.59765
## Sex.f                  1    0.0096 55.694 -1413.3  0.1013   0.75044
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
drop1(update(full_model_transform_log_inverse_response, ~. -
    Instagram.f - Children.f - OwnHome.f - Sex.f), test = "F")
```

```
## Single term deletions
##
## Model:
## log(Happy) ~ Household.f + Marital.f + Education_transformed +
##     JobSat.f + Income_log + Age_transformed
##                       Df Sum of Sq    RSS     AIC F value    Pr(>F)
## <none>                             55.694 -1413.3
## Household.f            6    1.4494 57.144 -1409.7  2.5504   0.01907 *
## Marital.f              4    1.0465 56.741 -1410.0  2.7621   0.02697 *
## Education_transformed  1    0.1260 55.820 -1413.9  1.3304   0.24920
## JobSat.f               6    3.8568 59.551 -1384.6  6.7864 5.702e-07 ***
## Income_log             1    0.4315 56.126 -1410.6  4.5552   0.03323 *
## Age_transformed        1    0.0287 55.723 -1415.0  0.3031   0.58217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
drop1(update(full_model_transform_log_inverse_response, ~. -
    Instagram.f - Children.f - OwnHome.f - Sex.f - Age_transformed),
    test = "F")
```

```
## Single term deletions
##
## Model:
## log(Happy) ~ Household.f + Marital.f + Education_transformed +
##     JobSat.f + Income_log
##                       Df Sum of Sq    RSS     AIC F value    Pr(>F)
## <none>                             56.178 -1427.0
## Household.f            6    1.4064 57.585 -1423.8  2.4784   0.02240 *
## Marital.f              4    1.2140 57.392 -1421.9  3.2090   0.01273 *
## Education_transformed  1    0.1265 56.305 -1427.6  1.3373   0.24797
## JobSat.f               6    3.9810 60.159 -1397.0  7.0156 3.163e-07 ***
## Income_log             1    0.4375 56.616 -1424.2  4.6264   0.03189 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
drop1(update(full_model_transform_log_inverse_response, ~. -
    Instagram.f - Children.f - OwnHome.f - Sex.f - Age_transformed -
    Education_transformed), test = "F")
```

```
## Single term deletions
##
## Model:
## log(Happy) ~ Household.f + Marital.f + JobSat.f + Income_log
##             Df Sum of Sq    RSS     AIC F value    Pr(>F)
```

```
## <none>                    56.305 -1427.6
## Household.f  6    1.3529 57.657 -1425.0  2.3828  0.027742 *
## Marital.f    4    1.2703 57.575 -1421.9  3.3559  0.009913 **
## JobSat.f     6    3.8827 60.187 -1398.7  6.8384 4.964e-07 ***
## Income_log   1    0.5783 56.883 -1423.3  6.1112  0.013711 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(update(full_model_transform_log_inverse_response, ~. -
    Instagram.f - Children.f - OwnHome.f - Sex.f - Age_transformed -
    Education_transformed))
```

```
##
## Call:
## lm(formula = log(Happy) ~ Household.f + Marital.f + JobSat.f +
##     Income_log)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9085 -0.1385  0.0232  0.2066  0.8419
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.493849   0.134165   3.681 0.000254 ***
## Household.f2  0.095511   0.034347   2.781 0.005594 **
## Household.f3  0.044931   0.045800   0.981 0.326981
## Household.f4 -0.083052   0.081541  -1.019 0.308836
## Household.f5 -0.053529   0.129606  -0.413 0.679743
## Household.f6  0.039566   0.180385   0.219 0.826460
## Household.f8  0.501450   0.314927   1.592 0.111855
## Marital.f2   -0.177191   0.069512  -2.549 0.011051 *
## Marital.f3   -0.103372   0.040588  -2.547 0.011119 *
## Marital.f4   -0.176811   0.069652  -2.538 0.011387 *
## Marital.f5   -0.064764   0.033894  -1.911 0.056511 .
## JobSat.f2    -0.009829   0.037340  -0.263 0.792470
## JobSat.f3    -0.124555   0.038943  -3.198 0.001455 **
## JobSat.f4    -0.185987   0.063941  -2.909 0.003764 **
## JobSat.f5    -0.186692   0.060201  -3.101 0.002019 **
## JobSat.f6    -0.348749   0.088456  -3.943 9.02e-05 ***
## JobSat.f7    -0.290503   0.181159  -1.604 0.109337
## Income_log    0.031429   0.012714   2.472 0.013711 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3076 on 595 degrees of freedom
##   (1754 observations deleted due to missingness)
## Multiple R-squared:   0.17,  Adjusted R-squared:  0.1463
## F-statistic:  7.17 on 17 and 595 DF,  p-value: 4.543e-16
```

```r
# summary(step(full_model_transform_log_inverse_response,
# direction = 'forward', trace = 1, scope = ~ Household.f +
# OwnHome.f + Instagram.f + Marital.f + Children.f +
# Education_transformed + JobSat.f + Income_log +
# Age_transformed + Sex.f))
```

AIC

```
library(leaps)
#X <- cbind(Household.f, OwnHome.f, Instagram.f, Marital.f, Children.f, Education_transformed, JobSat.f
#b <- regsubsets(as.matrix(X), log(Happy))
#rs <- summary(b)
#rs$adjr2
Rad <- summary(full_model_transform_log_inverse_response)$adj.r.squared
om1 <- lm(Happy ~ Household.f)
om2 <- lm(Happy ~ Household.f + OwnHome.f)
om3 <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f)
om4 <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f + Marital.f)
om5 <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f + Marital.f + Children.f)
om6 <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f + Marital.f + Children.f + Education_transformed
om7 <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f + Marital.f + Children.f + Education_transformed
om8 <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f + Marital.f + Children.f + Education_transformed
om9 <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f + Marital.f + Children.f + Education_transformed
om10 <- lm(Happy ~ Household.f + OwnHome.f + Instagram.f + Marital.f + Children.f + Education_transform
n = length(om1$residuals); Rad <-

p <- 1
AIC1 <- extractAIC(om1,k=2)[2] # AIC
AICc1 <- extractAIC(om1,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC1 <- extractAIC(om1,k=log(n))[2] # BIC
p <- 2
AIC2 <- extractAIC(om2,k=2)[2] # AIC
AICc2 <- extractAIC(om2,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC2 <- extractAIC(om2,k=log(n))[2] # BIC
p <- 3
AIC3 <- extractAIC(om3,k=2)[2] # AIC
AICc3 <- extractAIC(om3,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC3 <- extractAIC(om3,k=log(n))[2] # BIC
p <- 4
AIC4 <- extractAIC(om4,k=2)[2] # AIC
AICc4 <- extractAIC(om4,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC4 <- extractAIC(om4,k=log(n))[2] # BIC
p <- 5
AIC5 <- extractAIC(om5,k=2)[2] # AIC
AICc5 <- extractAIC(om5,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC5 <- extractAIC(om5,k=log(n))[2] # BIC
p <- 6
AIC6 <- extractAIC(om6,k=2)[2] # AIC
AICc6 <- extractAIC(om6,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC6 <- extractAIC(om6,k=log(n))[2] # BIC
p <- 7
AIC7 <- extractAIC(om7,k=2)[2] # AIC
AICc7 <- extractAIC(om7,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC7 <- extractAIC(om7,k=log(n))[2] # BIC
p <- 8
AIC8 <- extractAIC(om8,k=2)[2] # AIC
AICc8 <- extractAIC(om8,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC8 <- extractAIC(om8,k=log(n))[2] # BIC
p <- 9
```

```r
AIC9 <- extractAIC(om9,k=2)[2] # AIC
AICc9 <- extractAIC(om9,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC9 <- extractAIC(om9,k=log(n))[2] # BIC
p <- 10
AIC10 <- extractAIC(om10,k=2)[2] # AIC
AICc10 <- extractAIC(om10,k=2)[2]+2*(p+2)*(p+3)/(n-p-1) # AICc
BIC10 <- extractAIC(om10,k=log(n))[2] # BIC

AIC <- c(AIC1,AIC2,AIC3, AIC4, AIC5, AIC6, AIC7, AIC8, AIC9, AIC10)
AICc <- c(AICc1, AICc2, AICc3, AICc4, AICc5, AICc6, AICc7, AICc8, AICc9, AICc10)
BIC <- c(BIC1, BIC2, BIC3, BIC4, BIC5, BIC6, BIC7, BIC8, BIC9, BIC10)

opmodel <- data.frame(Size=1:10, Radj2= Rad, AIC=AIC, AICc=AICc, BIC=BIC)
opmodel
```

```
##    Size Radj2        AIC       AICc         BIC
## 1     1     1 -2099.9811 -2099.9709 -2059.61616
## 2     2     1 -1495.1291 -1495.1121 -1443.23135
## 3     3     1 -1493.2182 -1493.1927 -1435.55401
## 4     4     1 -1510.4634 -1510.4277 -1429.73358
## 5     5     1 -1506.5509 -1506.5034 -1379.68977
## 6     6     1 -1508.0539 -1507.9927 -1375.42626
## 7     7     1  -291.2076  -291.1311  -135.51438
## 8     8     1  -232.2593  -232.1657   -70.79958
## 9     9     1  -229.5688  -229.4564   -62.34268
## 10   10     1  -229.3383  -229.2055   -56.34583
```

```r
#Lowest AIC, AICc, BIC values occur when size = 6. Thus, we are retaining all variables
```

ASHWIN ADD AIC, BIC, and WHATNOT HERE

OLD STUFF

The first conclusion we came to in our model selection process was that WorkHrs had to be excluded, as there were 1898 missing values, most of which were -1 (Inapplicable). Its sheer amount of missing values made it ineligible for model fitting - every attempt to include it resulted in an error being thrown.

```r
# Finding the number of NAs
sum(is.na(happiness_data$WorkHrs))
```

```
## [1] 1898
```

```r
# Insta = 10, Marital = 1; #Household = 1; #Health = 811;
# #OwnHome = 812; #JobSat = 1612; #WorkHrs = 1898; #Income =
# 1039
```

We plotted the full model (without WorkHrs). This factored in around 190 observations, as all the rest had NAs under some variables. We found the Residuals vs Fitted plot showed a linear trend, a result of some of the predictor variables being categorical. The standardized residual plot also showed a pattern that skewed the plot much more than it did in the Residuals vs. Fitted plot.

```r
attach(happiness_data)
```

```
## The following objects are masked from happiness_data (pos = 4):
##
```

```
##     Age, Children, Education, Happy, Health, Household, Income,
##     Instagram, JobSat, Marital, OwnHome, Sex, WorkHrs
```

```
# Factoring Categorical Variables
JobSat.f <- factor(JobSat)
OwnHome.f <- factor(OwnHome)
Marital.f <- factor(Marital)
Instagram.f <- factor(Instagram)
Health.f <- factor(Health)
# Couldn't include Health as it throws an error
full_model <- lm(Happy ~ Household + OwnHome.f + Instagram.f +
    Marital.f + Children + Education + JobSat.f + Income + Age +
    Sex)
```

We decided the best way to proceed would be to test each variable's significance individually. We created models with individual variables and Happy, finding that Instagram and Sex did not have statistically significant linear relationships to Happy.

```
insta <- lm(Happy ~ Instagram)
# summary(insta); plot(insta); Instagram is insignificant
sex <- lm(Happy ~ Sex)
# summary(sex); plot(sex); Sex is insignificant
```

We then used partial F-tests to verify these findings, as well as potentially weed out other variables. To do so, we created models that each excluded one variable and then tested them against our full model. This method found Children and Education to be insignificant in addition to Instagram and Sex. OwnHome, JobSat, Income and Age threw errors in partial F-testing, so we have not included the code for those.

```
noMarital <- lm(Happy ~ Sex + JobSat.f + OwnHome.f + Household +
    Education + Age + Income + Children + Instagram.f)
anova(full_model, noMarital)$`Pr(>F)`
```

```
## [1]          NA 0.003016323
```

```
# Marital is significant
noHousehold <- lm(Happy ~ Sex + JobSat.f + OwnHome.f + Marital.f +
    Education + Age + Income + Children + Instagram.f)
anova(full_model, noHousehold)$`Pr(>F)`
```

```
## [1]          NA 0.003873214
```

```
# Household is significant
noSex <- lm(Happy ~ JobSat.f + OwnHome.f + Household + Marital.f +
    Education + Age + Income + Children + Instagram.f)
anova(full_model, noSex)$`Pr(>F)`
```

```
## [1]        NA 0.2419433
```

```
# Sex is insignificant
noInstagram <- lm(Happy ~ Sex + JobSat.f + OwnHome.f + Household +
    Marital.f + Education + Age + Income + Children)
anova(full_model, noInstagram)$`Pr(>F)`
```

```
## [1]        NA 0.4980311
```

```
# Instagram is insignificant
noChildren <- lm(Happy ~ Sex + JobSat.f + OwnHome.f + Household +
    Marital.f + Education + Age + Income + Instagram.f)
anova(full_model, noChildren)$`Pr(>F)`
```

```
## [1]        NA 0.4352332
```

```r
# Children is insignificant
noEducation <- lm(Happy ~ Sex + JobSat.f + OwnHome.f + Household +
    Marital.f + Age + Income + Children + Instagram.f)
anova(full_model, noEducation)$`Pr(>F)`
```

```
## [1]        NA 0.2048761
```

```r
# Education is insignificant
```

After eliminating the four insignificant variables, we obtained AIC, AICc and BIC values, which were lowest when all six variables were included. Performing forward selection showed OwnHome to be insignificant and performing backward selection showed Age to be insignificant. Since the forward and backward selections were not in agreement, this did not seem like strong enough evidence to exclude the variables to us. We found including all 6 variables gave us the lowest values for each, so we did not choose to omit any variables from the model in this process.

```r
# Eliminating education, instagram, children, sex
new_model <- lm(Happy ~ JobSat.f + OwnHome.f + Marital.f + Household +
    Income + Age)

Rad <- summary(new_model)$adj.r.squared
om1 <- lm(Happy ~ JobSat.f)
om2 <- lm(Happy ~ JobSat.f + OwnHome.f)
om3 <- lm(Happy ~ JobSat.f + OwnHome.f + Marital.f)
om4 <- lm(Happy ~ JobSat.f + OwnHome.f + Marital.f + Household)
om5 <- lm(Happy ~ JobSat.f + OwnHome.f + Marital.f + Household +
    Age)
om6 <- lm(Happy ~ JobSat.f + OwnHome.f + Marital.f + Household +
    Age + Income)
n = length(Happy)

p <- 1
oms1 <- summary(om1)
AIC1 <- extractAIC(om1, k = 2)[2]  # AIC
AICc1 <- extractAIC(om1, k = 2)[2] + 2 * (p + 2) * (p + 3)/(n -
    p - 1)  # AICc
BIC1 <- extractAIC(om1, k = log(n))[2]  # BIC
p <- 2
oms2 <- summary(om2)
AIC2 <- extractAIC(om2, k = 2)[2]  # AIC
AICc2 <- extractAIC(om2, k = 2)[2] + 2 * (p + 2) * (p + 3)/(n -
    p - 1)  # AICc
BIC2 <- extractAIC(om2, k = log(n))[2]  # BIC
p <- 3
oms3 <- summary(om3)
AIC3 <- extractAIC(om3, k = 2)[2]  # AIC
AICc3 <- extractAIC(om3, k = 2)[2] + 2 * (p + 2) * (p + 3)/(n -
    p - 1)  # AICc
BIC3 <- extractAIC(om3, k = log(n))[2]  # BIC
p <- 4
oms4 <- summary(om4)
AIC4 <- extractAIC(om4, k = 2)[2]  # AIC
AICc4 <- extractAIC(om4, k = 2)[2] + 2 * (p + 2) * (p + 3)/(n -
    p - 1)  # AICc
```

```r
BIC4 <- extractAIC(om4, k = log(n))[2]  # BIC
p <- 5
oms5 <- summary(om5)
AIC5 <- extractAIC(om5, k = 2)[2]  # AIC
AICc5 <- extractAIC(om5, k = 2)[2] + 2 * (p + 2) * (p + 3)/(n -
    p - 1)  # AICc
BIC5 <- extractAIC(om5, k = log(n))[2]  # BIC
p <- 6
oms6 <- summary(om6)
AIC6 <- extractAIC(om6, k = 2)[2]  # AIC
AICc6 <- extractAIC(om6, k = 2)[2] + 2 * (p + 2) * (p + 3)/(n -
    p - 1)  # AICc
BIC6 <- extractAIC(om6, k = log(n))[2]  # BIC

AIC <- c(AIC1, AIC2, AIC3, AIC4, AIC5, AIC6)
AICc <- c(AICc1, AICc2, AICc3, AICc4, AICc5, AICc6)
BIC <- c(BIC1, BIC2, BIC3, BIC4, BIC5, BIC6)

opmodel <- data.frame(Size = 1:6, Radj2 = Rad, AIC = AIC, AICc = AICc,
    BIC = BIC)
opmodel
```

```
##   Size     Radj2       AIC      AICc       BIC
## 1    1 0.2013563 -721.0678 -721.0577 -680.6822
## 2    2 0.2013563 -286.4328 -286.4159 -234.5084
## 3    3 0.2013563 -293.8557 -293.8303 -218.8538
## 4    4 0.2013563 -298.4634 -298.4278 -217.6921
## 5    5 0.2013563 -296.6381 -296.5906 -210.0974
## 6    6 0.2013563 -239.8424 -239.7814 -147.5323
```

```r
# Lowest AIC, AICc, BIC values occur when size = 6. Thus, we
# are retaining all variables

# Checking Forward Selection
add1(lm(Happy ~ 1), Happy ~ JobSat.f + OwnHome.f + Marital.f +
    Household + Age + Income, test = "F")$`Pr(>F)`  #prints p-value
```

```
## Warning in add1.lm(lm(Happy ~ 1), Happy ~ JobSat.f + OwnHome.f + Marital.f
## + : using the 204/2361 rows from a combined fit

## [1]           NA 1.728212e-73 6.417063e-05 2.519737e-43 3.824285e-08
## [6] 4.212786e-01 5.915053e-32
```

```r
# Age seems to be insignificant

# Performing another forward selection to see if age is
# actually insignificant
add1(lm(Happy ~ JobSat.f), Happy ~ JobSat.f + OwnHome.f + Marital.f +
    Household + Age + Income, test = "F")$`Pr(>F)`  #prints p-value
```

```
## Warning in add1.lm(lm(Happy ~ JobSat.f), Happy ~ JobSat.f + OwnHome.f + :
## using the 204/754 rows from a combined fit

## [1]           NA 4.392858e-02 1.009862e-09 1.578902e-03 5.276440e-01
## [6] 6.963063e-07
```

```r
# Backward Selection to see check the significance of
# variables
drop1(lm(Happy ~ JobSat.f + OwnHome.f + Marital.f + Household +
    Age + Income), test = "F")$`Pr(>F)`  #prints p-value
```

```
## [1]           NA 0.0008187357 0.1868556335 0.0037590269 0.0034175331
## [6] 0.0239082085 0.0385886570
```

```r
# Ownhome seems to be insignificant
drop1(lm(Happy ~ JobSat.f + OwnHome.f + Marital.f + Household +
    Income), test = "F")$`Pr(>F)`  #prints p-value
```

```
## [1]           NA 0.0009888392 0.3642221333 0.0139487313 0.0153868191
## [6] 0.0637063446
```

Our untransformed model thus contained JobSat, OwnHome, Marital and Household in factor form, as well as the numerical variables Income and Age. Its R squared value was 0.2844, and its adjusted R squared value improved to 0.2105.

```r
Household.f <- factor(Household)

Final_model <- lm(Happy ~ JobSat.f + OwnHome.f + Marital.f +
    Household.f + Income + Age)
summary(Final_model)$r.squared
```

```
## [1] 0.2844382
```

```r
summary(Final_model)$adj.r.squared
```

```
## [1] 0.2105487
```

We used both the Box Cox and inverse response plot methods in transforming our model. Per Box Cox, we raised Age to the power of 0.226. Intuitively, looking at the relationship between Income and Happy, we decided on a logarithmic transformation on Income, as well. The adjusted R squared value reduced a little from this transformation, to 0.2099, and R squared dropped to 0.2838. The inverse response plot suggested a lambda of -0.1130549, but the RSS for a lambda of 0 was very similar, so we took the log transformation of our response variable instead, as it it represented a better representation of real world data. We ended with an R^2 value of 0.3092 and an adjusted R^2 of 0.2378.

```r
# Box Cox transformation
powerTransform(cbind(JobSat.f, OwnHome.f, Marital.f, Household.f,
    Income, Age) ~ 1)
```

```
## Estimated transformation parameters
##    JobSat.f   OwnHome.f   Marital.f Household.f      Income         Age
##   0.1523107  -1.9515759   0.2229948  -0.2117924   0.2160991   0.4521715
```

```r
Age_transformed <- Age^0.226
m_new <- lm(Happy ~ JobSat.f + OwnHome.f + Marital.f + Household.f +
    log(Income) + Age_transformed)
# summary(m_new)

# Inverse response plot
par(mfrow = c(2, 2))
m_new <- lm(log(Happy) ~ JobSat.f + OwnHome.f + Marital.f + Household.f +
    log(Income) + Age_transformed)
summary(m_new)
```

```
##
```

```
## Call:
## lm(formula = log(Happy) ~ JobSat.f + OwnHome.f + Marital.f +
##     Household.f + log(Income) + Age_transformed)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.85310 -0.15026 -0.00986  0.17885  0.54401
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.26028    0.39631   3.180 0.001728 **
## JobSat.f2       -0.06340    0.05695  -1.113 0.267085
## JobSat.f3       -0.16091    0.06045  -2.662 0.008453 **
## JobSat.f4       -0.12654    0.10126  -1.250 0.213043
## JobSat.f5       -0.34313    0.08838  -3.882 0.000144 ***
## JobSat.f6       -0.48979    0.13130  -3.730 0.000255 ***
## JobSat.f7       -0.13356    0.27702  -0.482 0.630291
## OwnHome.f2       0.02446    0.04382   0.558 0.577330
## OwnHome.f3       0.24536    0.20878   1.175 0.241428
## Marital.f2      -0.40775    0.10880  -3.748 0.000239 ***
## Marital.f3      -0.15417    0.06445  -2.392 0.017759 *
## Marital.f4      -0.13232    0.12918  -1.024 0.307024
## Marital.f5      -0.18972    0.06156  -3.082 0.002374 **
## Household.f2    -0.09814    0.05651  -1.736 0.084157 .
## Household.f3    -0.03689    0.07957  -0.464 0.643503
## Household.f4    -0.45965    0.11490  -4.000 9.16e-05 ***
## Household.f5    -0.23737    0.20499  -1.158 0.248370
## Household.f6    -0.40293    0.19714  -2.044 0.042389 *
## log(Income)      0.03511    0.01981   1.772 0.077969 .
## Age_transformed -0.23907    0.14813  -1.614 0.108244
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2675 on 184 degrees of freedom
##   (2163 observations deleted due to missingness)
## Multiple R-squared:  0.3092, Adjusted R-squared:  0.2378
## F-statistic: 4.334 on 19 and 184 DF,  p-value: 6.092e-08
```

Now we see how many bad leverage points we have. We have 7 bad leverage points. We conclude that our final model is accurate.

```r
StanRes1 <- rstandard(m_new); leverage1 <- hatvalues(m_new); cookd1 <- cooks.distance(m_new); p <- 7; n
a <- which(StanRes1 > 2 | StanRes1 < -2); b <- which(leverage1 > 2*(p+1)/n)
intersect(a, b)
```

```
## [1]  44  62  72 108 145 170 172
```