# Math 651 Final Project

*Mary Peng, Hillary Dunn, and Max Kearns*

*December 17, 2018*

## 1 Abstract

## 2 Introduction

#### 2.0.0.0.1 needs editing

Every four summers, the Olympic Games become the center of the world's attention, as elite athletes seek honor for both themselves and for their countries. Many countries associate tremendous national pride with their medal counts, since a nation's athletic competence also projects its soft power. For this last reason, some governments invest generously in their sports programs, in hopes of an elevated medal count in the next Olympics.

Given the fierce competition and high profile nature of medal counts, one may wonder what factors influence the number of medals that a country wins at the summer Olympics. Certainly countries with the largest economies and populations, such as the United States and China, commonly dominate the top of the billboard. However, Azerbaijan, which ranks 91st in population and 72nd in total GDP, also ranked in the top 20 countries by total medal count in the 2016 Summer Olympics.

Our team will develop multiple regression models using various predictors, such as GDP, population size, size of athletic investment, and geographic location, to predict medal count. We will build the model on countries' total medal count for the 2004, 2008, and 2012 Olympics, and then project the model onto the 2016 Olympics to understand the accuracy of the model's predictions.

## 3 Methods and Materials

### 3.1 Data

The purpose of this reasearch is to determine whether there is a predictive relationship between popluation, GDP per capita, being a host nation, or at one point being a Communist nation or a member of the Soviet Union and number of medals won at the Olympics in the associated year by country.

#### 3.1.1 EDA

As we can see in table 1, the average number of medals won by a country in a year is 11.79 medals and the minimum number of medals is 1. This is meaningful because this shows that we are only assessing countries that have won at least one medal in at least one of the years we are analyzing.

We can also see in Figure 1 that the medal count is distributed exponentially, which could indicate a need to transform the response variable if we need to fit a linear regression model.

$X_1$: This variable indicates population of people in the country in the associated year. In figure 2, we can see a histogram of the population data. On the left, is the original distribution of the data points and on the right, we've transformed the data to look more normally distributed This should allow for a cleaner regression model, although interpretation of parameters may be slightly more difficult.

$X_2$: This variable indicates GDP per capita in the associated year. These data show a similar distribution to the population variable, so we again decided to transform this variable, for the same reasons.

$X_3$: This is a binary predictor variable that indicates with a 1 if a country has or will host the Olympics within the 16 years surrounding the olympics in question. Of all the data points, 25 are classified as a host by this measure, which is consistent with the definition of the variable, and the number of years in the data set.

$X_4$: This binary predictor variable indicates whether the country was once a member of the Soviet Union or if they were indicated as ever being a Communist country. From ______________ we aggregated a binary predictor variable that identifies countries that were once members of the Soviet Union or at one time classified as communist countries. 111 data points are classified as former Soviet Union or Communist.(INSERT SOURCE AGAIN???)

To properly determine an ideal model for the data, we need to address the relationships between variables. We can do this be analyzing both the scatterplot matrix of variables and the correlation matrix.

The scatterplot matrix in Figure 3 does not obviously show us many strong relationships between the predictor variables and their effect on the response variable. While there might be a relationship between the log of the population and the medal count, it is difficult to determine from the plot if this is a linear relationship. Additional information can be collected from the correlation matrix below.

This correlation matrix in Table 3 suggests that collinearity between variables will not be a significant issue. The most significant correlation between predictor variables is between the log of the GDP per capita and the indicator of Communism or Soviet Union inclusion variable. However, the correlation coefficient is only -0.2989. This suggests that multicollinearity should not be much of an issue.

## 3.2 Model Building and Selection

### 3.2.1 Linear Model

According to the selected model diagnostics shown in Table 3, the model with the smallest $C_p$ statistic, largest adjusted $R^2$ value, smallest AIC and smallest PRESS value is the model that includes all four predictor variables. We constructed a linear model (shown below), using the natural log to transform the count data.

$$X_1 = \ln(\text{population})$$
$$X_2 = \ln(\text{gdp/capita})$$
$$X_3 = \text{host}$$
$$X_4 = \text{Soviet Country or Communist}$$
$$Y = \ln(\text{Count})$$
$$\hat{Y} = -9.63058 + 0.44275X_1 + 0.40830X_2 + 0.86726X_3 + 1.03281X_4$$

According to the Normal Q-Q plot (Figure X), it is reasonable to assume the residuals follow a normal distribution. Hoever, the Residuals vs Fitted plot indicates possible heteroskedasticity or non-constant variance of the errors. The Breusch-Pagan test will help us determine if heteroskedasticity is affecting the model.

Breusch Pagan Test: Assuming $Var(\epsilon_i) = \sigma_i^2$ such that $\log_e \sigma_i^2 = \gamma_0 + \gamma_1 X_i$: Alternatives:

$$H_0 : \gamma_1, \text{vs. } H_a : \gamma_1 \neq 0$$

Decision Rule: At significance level $\alpha = 0.05$, if the p-value of the Breusch-Pagan test is less than $\alpha$ then we reject $H_0$ and accept $H_a$, otherwise we fail to reject $H_0$. Accepting $H_a$ means we accept that the variance is non-constant.

The p-value = $0.001587 < \alpha$, therefore we accept the alternative hypothesis that our variance is not constant.

Test for normality (Lilliefors Test):
Alternatives:

$$H_0 : \text{Sample comes from a N}(\mu, \sigma^2) \text{ distribution}$$

$$H_a : \text{Sample does not come from a N}(\mu, \sigma^2) \text{ distribution}$$

Decision Rule: At signifcance level $\alpha = 0.05$, if the p-value is less than $\alpha$ then we accept $H_a$ and reject $H_0$, otherwise we fail to reject $H_0$.

The p-value= $0.2332 > \alpha$, thus we fail to reject $H_0$. Therefore, it is reasonable to assume that the error terms are distributed normally.

APPENDIX: A. Influential Cases by test:

### 3.2.2 Generalized Linear Model

The first choice for a glm is typically a Poisson model, which may provide a reasonable model for these data. This generalized model was created with the same variables as the linear model; population, GDP per capita, the host dummy variable, and the communist/soviet dummy variable. Like the linear model, the population and GDP per capita were log-transformed, but in this model it is unnecessary to transform the medal count. The full poisson model showed highly significant parameters, but the dispersion of this model was much greater than 1 (disp. = 6.621), so a negative binomial model would likely provide a better fit for these data.

A negative binomial regression model assumes that dispersion is greater than 1, which is consistent with these data. Therefore, it allows for more accurate tests of the parameters. Among the negative binomial models, the best model again proved to be the full model, after a comparison of AIC between all subsets of variables (Table 2).

# 4 Results

# 5 Discussion and Conclusions

# 6 Bibliography

# 7 Appendix A

Table 1: Summary of Count Variable

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max |
|---|---|---|---|---|---|
| 1 | 2 | 5 | 11.78811 | 12.5 | 110 |

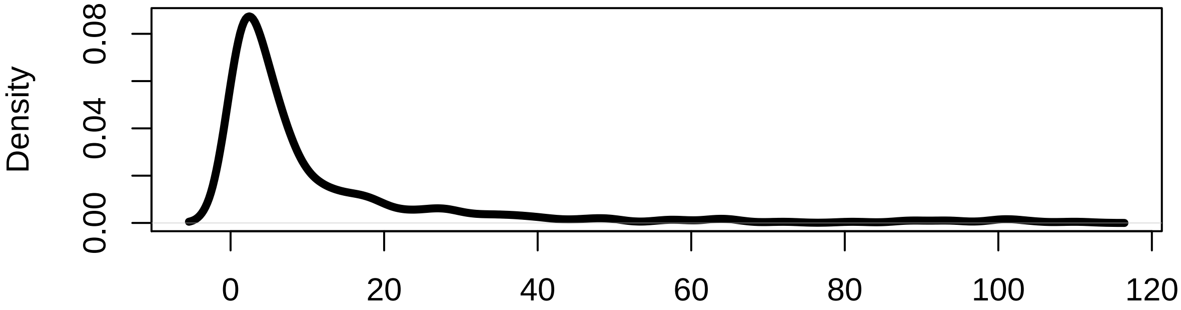

Figure 1: Distribution of Count

Table 2: Correlation Matrix

| | count | log_pop | log_gdp_per_cap | host | comm_soviet |
|---|---|---|---|---|---|
| count | 1.0000000 | 0.4839466 | 0.2291600 | 0.4594891 | 0.0871524 |
| log_pop | 0.4839466 | 1.0000000 | -0.1901839 | 0.2548345 | -0.1480479 |
| log_gdp_per_cap | 0.2291600 | -0.1901839 | 1.0000000 | 0.1636597 | -0.2989122 |
| host | 0.4594891 | 0.2548345 | 0.1636597 | 1.0000000 | -0.0736893 |
| comm_soviet | 0.0871524 | -0.1480479 | -0.2989122 | -0.0736893 | 1.0000000 |

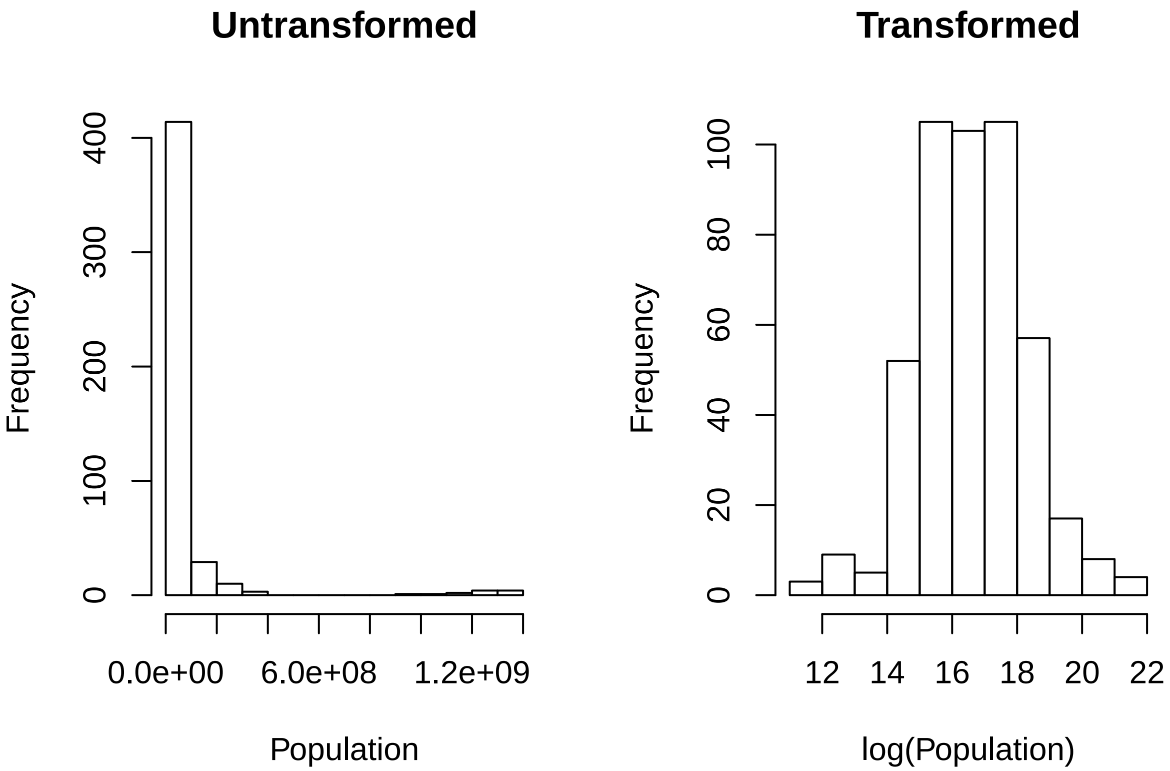

Figure 2: Histogram of the Population Variables

Table 3: Model Selection for Linear Model

| | 1 | 2 | 3 | 4 | Cp | aR2 | AIC | PRESS |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 | 250.23 | 0.22 | 2497.82 | 481.0928 |
| 1 | 0 | 0 | 1 | 0 | 319.74 | 0.13 | 2633.68 | 532.7428 |
| 1 | 0 | 1 | 0 | 0 | 357.45 | 0.09 | 2658.69 | 561.6892 |
| 1 | 0 | 0 | 0 | 1 | 418.54 | 0.01 | 2695.53 | 608.1812 |
| 2 | 1 | 1 | 0 | 0 | 123.00 | 0.38 | 2474.29 | 498.7497 |
| 2 | 1 | 0 | 1 | 0 | 197.75 | 0.29 | 2428.33 | 384.4047 |
| 2 | 1 | 0 | 0 | 1 | 220.38 | 0.26 | 2489.15 | 458.7142 |
| 2 | 0 | 1 | 1 | 0 | 273.23 | 0.19 | 2601.80 | 498.7497 |
| 2 | 0 | 0 | 1 | 1 | 302.56 | 0.16 | 2626.84 | 520.7314 |
| 2 | 0 | 1 | 0 | 1 | 317.78 | 0.14 | 2614.08 | 532.9378 |
| 3 | 1 | 1 | 0 | 1 | 23.77 | 0.50 | 2338.79 | 366.8114 |
| 3 | 1 | 1 | 1 | 0 | 100.33 | 0.41 | 2417.52 | 309.0094 |
| 3 | 1 | 0 | 1 | 1 | 164.64 | 0.33 | 2457.19 | 416.4538 |
| 3 | 0 | 1 | 1 | 1 | 230.32 | 0.25 | 2556.83 | 466.7762 |
| 4 | 1 | 1 | 1 | 1 | 5.00 | 0.53 | 2319.23 | 294.4012 |

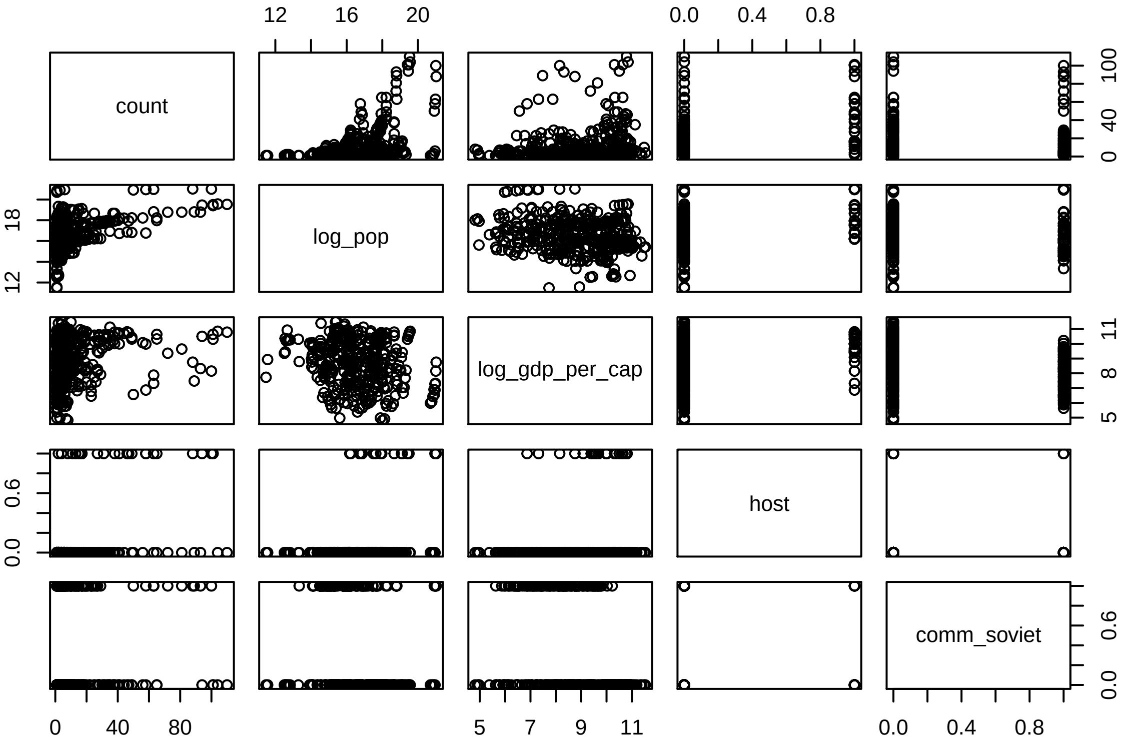

Figure 3: Scatter Matrix

Table 4: Model Selection Diagnostics for a Negative Binomial Model

| Pop | GDP/C | Host | Soviet | Parameters | Cp.nb | AIC.nb |
|---:|---:|---:|---:|---:|---:|---:|
| 1 | 0 | 0 | 0 | 2 | 198.46 | 2497.82 |
| 0 | 0 | 1 | 0 | 2 | 215.98 | 2633.68 |
| 0 | 1 | 0 | 0 | 2 | 336.41 | 2658.69 |
| 0 | 0 | 0 | 1 | 2 | 370.52 | 2695.53 |
| 1 | 0 | 1 | 0 | 3 | 108.70 | 2474.29 |
| 1 | 1 | 0 | 0 | 3 | 119.18 | 2428.33 |
| 1 | 0 | 0 | 1 | 3 | 180.88 | 2489.15 |
| 0 | 1 | 1 | 0 | 3 | 199.48 | 2601.80 |
| 0 | 0 | 1 | 1 | 3 | 206.80 | 2626.84 |
| 0 | 1 | 0 | 1 | 3 | 318.21 | 2614.08 |
| 1 | 1 | 0 | 1 | 4 | 58.39 | 2338.79 |
| 1 | 1 | 1 | 0 | 4 | 60.95 | 2417.52 |
| 1 | 0 | 1 | 1 | 4 | 87.77 | 2457.19 |
| 0 | 1 | 1 | 1 | 4 | 178.38 | 2556.83 |
| 1 | 1 | 1 | 1 | 5 | 5.00 | 2319.23 |

Table 5: Negative Binomial Model Coefficients

| | Estimate | Std. Error | z value | p-value |
|---|---:|---:|---:|---:|
| Intercept | -10.1657683 | 0.6468624 | -15.715504 | 0.00e+00 |
| log(Population) | 0.4984760 | 0.0288138 | 17.299910 | 0.00e+00 |
| log(GDP/Capita) | 0.4023884 | 0.0317426 | 12.676614 | 0.00e+00 |
| Host | 0.6926724 | 0.1593195 | 4.347693 | 1.38e-05 |
| Comm/Soviet | 1.0337603 | 0.0984874 | 10.496376 | 0.00e+00 |

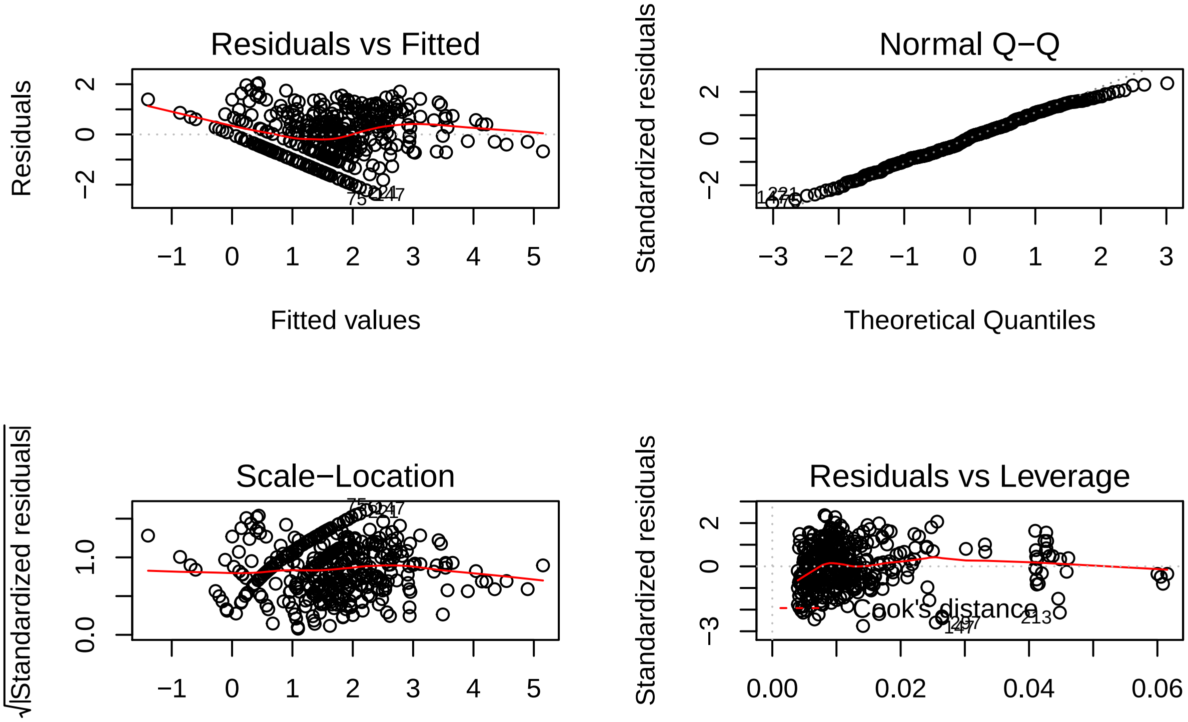

Figure 4: Exploratory Linear Model Plots

## 8 Appendix B

```
knitr::opts_chunk$set(echo = FALSE, comment = NA)
base<-read.csv('data/base_data.csv', stringsAsFactors = F)
P__disp <- function(x) {
  pr <- sum(residuals(x, type="pearson")^2)
  dispersion <- pr/x$df.residual
  c(pr, dispersion)
}
library(dplyr)
library(qpcR)
library(MuMIn)

base.total = base[which(base$year!=2016),]
Olympic = base.total[,c(2,3,4,5,6,7,8)]

#Clean the data (Mary to add)

#Make new dataframe with GDP / capita
Olympic_v2 <- data.frame(year = Olympic$year, country = Olympic$country, count = Olympic$count, log_pop
library(knitr)
summary(base$count)
```

```
hist(base$count)
hist(base[which(base$year == c("2008")),c("count")])
summary(base$gdp)
summary(base$pop)
sum(base$comm_soviet)
nrow(base)-sum(base$comm_soviet)
sum(base$host)
par(mfrow=c(1,2))
#Histogram Population
hist(base$pop,main = "Untransformed",xlab = "Population")
#Histogram log(Population)
hist(log(base$pop),main = "Transformed",xlab = "log(Population)")
#Log(GDP)
boxplot(log(base$gdp),ylab = "log(GDP)")
#log(Population)
boxplot(log(base$pop),ylab = "log(Population)")
pairs(Olympic[,c(2,4,5,6,7)])
kable(cor(Olympic[,c(2,4,5,6,7)]))
library(leaps)
#CP
olympic.leapCP <- leaps(y=log(Olympic_v2$count), x=Olympic_v2[,4:7])
#R2a
olympic.leapR2a <- leaps(y=log(Olympic_v2$count), x=Olympic_v2[,4:7], method = 'adjr2')

  ### Code for AIC
xList <- names(Olympic_v2)[4:7]
  #### Remove the last row that has all False's
vec <- olympic.leapCP$which
  ### Name the columns in the grid
names(vec) <- paste("X", 1:4, sep="")
  #### Build matrix of formula for every row
allModelsList <- apply(vec, 1, function(x) as.formula(
  paste(c("count ~ 1", xList[x]), collapse = "+")))
  ### Calculate the coefficients for all 16 models
allModelsResults <- lapply(allModelsList,
                           function(x) lm(x, data=Olympic_v2))

olympic.lmfinal <- lm(log_count ~ log_pop + log_gdp_per_cap + host + comm_soviet, data = Olympic_v2)
par(mfrow=c(2,2))
plot(olympic.lmfinal)
library(lmtest)

bptest(log_count ~ log_pop + log_gdp_per_cap + host + comm_soviet,data = Olympic_v2,studentize = FALSE)
library(nortest)
lillie.test(olympic.lmfinal$residuals)
#Influential cases
olympic.lm_inf=influence.measures(olympic.lmfinal)$is.inf
idx=which(apply(olympic.lm_inf,1,any))

Olympic[idx,]
olympic.lm_inf[idx,]
Olympic.pois<-glm(count~log_pop + log_gdp_per_cap + host + comm_soviet, data = Olympic_v2, family = pois
P__disp(Olympic.pois)
```

```
library(MASS)
olympic.nb <- glm.nb(count~log_pop + log_gdp_per_cap + host + comm_soviet, data = Olympic_v2)
library(leaps)
olympic.nb_leap <- leaps(y=Olympic_v2$count, x=Olympic_v2[,4:7])
Cp.nb<-round(olympic.nb_leap$Cp, 2)

which<-olympic.nb_leap$which
rownames(which) <-NULL
colnames(which)<-c('Pop', 'GDP/C', 'Host', 'Soviet')
xList <- names(Olympic_v2)[4:7]
vec <- olympic.nb_leap$which

#Name the columns in the grid
names(vec) <- paste("X", 1:4, sep="")

#Build matrix of formula for every row
allModelsList <- apply(vec, 1, function(x) as.formula(
  paste(c("count ~ 1", xList[x]), collapse = "+")))

#Calculate the coefficients for all 16 models
allModelsResults <- lapply(allModelsList,
                           function(x) glm.nb(x, data=Olympic_v2))


AIC.nb<-matrix(unlist(lapply(allModelsResults, function(x) round(extractAIC(x),2))), ncol = 2, byrow = 
library(knitr)
df_1<-data.frame(matrix(summary(Olympic_v2$count), ncol = 6))
colnames(df_1)<-c('Min.', '1st Qu.','Median',    'Mean', '3rd Qu.',    'Max')
kable(df_1, caption = 'Summary of Count Variable')
plot(density(Olympic_v2$count), lwd = 4, xlab = '', main = '')
par(mfrow=c(1,2))
#Histogram Population
hist(base$pop,main = "Untransformed",xlab = "Population")
#Histogram log(Population)
hist(log(base$pop),main = "Transformed",xlab = "log(Population)")
plot(Olympic_v2[,c(3,4,5,6,7)])
kable(cor(Olympic_v2[,c(3,4,5,6,7)]), caption = 'Correlation Matrix')
#PRESS (Non-Mac)
library(qpcR)
olympic.lm = PRESS(lm(log_count~log_pop+log_gdp_per_cap+host+comm_soviet, data = Olympic_v2))
olympic.lmX1 = PRESS(lm(log_count~log_pop, data = Olympic_v2))
olympic.lmX2 = PRESS(lm(log_count~log_gdp_per_cap, data = Olympic_v2))
olympic.lmX3 = PRESS(lm(log_count~host, data = Olympic_v2))
olympic.lmX4 = PRESS(lm(log_count~comm_soviet, data = Olympic_v2))
olympic.lmX1X2 = PRESS(lm(log_count~log_pop+log_gdp_per_cap, data = Olympic_v2))
olympic.lmX1X3 = PRESS(lm(log_count~log_gdp_per_cap+host, data = Olympic_v2))
olympic.lmX1X4 = PRESS(lm(log_count~log_pop+comm_soviet, data = Olympic_v2))
olympic.lmX2X3 = PRESS(lm(log_count~log_gdp_per_cap+host, data = Olympic_v2))
olympic.lmX2X4 = PRESS(lm(log_count~log_gdp_per_cap+comm_soviet, data = Olympic_v2))
olympic.lmX3X4 = PRESS(lm(log_count~host+comm_soviet, data = Olympic_v2))
olympic.lmX1X2X3 = PRESS(lm(log_count~log_pop+log_gdp_per_cap+host, data = Olympic_v2))
olympic.lmX1X2X4 = PRESS(lm(log_count~log_pop+log_gdp_per_cap+comm_soviet, data = Olympic_v2))
olympic.lmX2X3X4 = PRESS(lm(log_count~log_gdp_per_cap+host+comm_soviet, data = Olympic_v2))
```

```r
olympic.lmX1X3X4 = PRESS(lm(log_count~log_pop+host+comm_soviet, data = Olympic_v2))

olympic.lm_press <- rbind(olympic.lmX1$stat,
                          olympic.lmX3$stat,
                          olympic.lmX2$stat,
                          olympic.lmX4$stat,
                          olympic.lmX1X3$stat,
                          olympic.lmX1X2$stat,
                          olympic.lmX1X4$stat,
                          olympic.lmX2X3$stat,
                          olympic.lmX3X4$stat,
                          olympic.lmX2X4$stat,
                          olympic.lmX1X2X3$stat,
                          olympic.lmX1X2X4$stat,
                          olympic.lmX1X3X4$stat,
                          olympic.lmX2X3X4$stat,
                          olympic.lm$stat)

#Summary
Diagnostics = cbind(olympic.leapCP$which, Cp=round(olympic.leapCP$Cp,2), aR2=round(olympic.leapR2a$adjr2
      AIC=matrix(unlist(lapply(allModelsResults, function(x) round(extractAIC(x),2))), ncol=2, byrow=TRU
#PRESS wasn't showing as column name
colnames(Diagnostics) = c("1","2","3","4","Cp","aR2","AIC","PRESS")
par(mfrow=c(2,2))
plot(olympic.lmfinal)
kable(Diagnostics, caption = 'Model Selection for Linear Model')
kable(cbind(which, Parameters = olympic.nb_leap$size, Cp.nb, AIC.nb), caption = 'Model Selection Diagnos
tab<-summary(olympic.nb)$coefficients
colnames(tab)[4]<-'p-value'
rownames(tab)<-c('Intercept', 'log(Population)', 'log(GDP/Capita)', 'Host', 'Comm/Soviet')
kable(tab, format.args = list(justify = 'centre'), caption = 'Negative Binomial Model Coefficients')
```