Predicting Eastern Indigo Snake Occurrences in Florida Using Construction Permitting Data

Jonathan Bunch

8/6/2020

Introduction

The conservation of threatened species is an important, albeit daunting, endeavor. Unfortunately, many ecologically important species are overlooked by the general public simply because they are not considered appealing. Snake conservation is particularly difficult to promote, due to the common fears and misconceptions associated with them. The Eastern Indigo snake (Drymarchon couperi) is a large, non-venomous snake native to the south-eastern United States. This species has great value as a predator in the ecosystems it inhabits, feeding on many common pest animals. This species also has the unique ability to predate venomous snake species, helping to control the population of species that are potentially dangerous to humans.

The Eastern Indigo snake is currently listed as "threatened" by the US Fish and Wildlife Service. The primary cause for declining populations is thought to be habitat destruction and fragmentation. These snakes are known to co-occupy the borrows of gopher tortoises (Gopherus polyphemus), whose population has also sharply declined, adding further difficulties for the species. While both of these species are legally protected, it is rumored that loopholes are commonly exploited by real- estate developers to bypass mandatory reporting and damage mitigation processes.

The goal of this project was to explore the impact of new construction projects on the observed occurrences of this species in Florida. Observed occurrences (sightings) of this species are documented and verified by several scientific organizations and made available as an organized data set. New construction data were collected from the U.S. Census Building Permits Survey.

Problem Statement

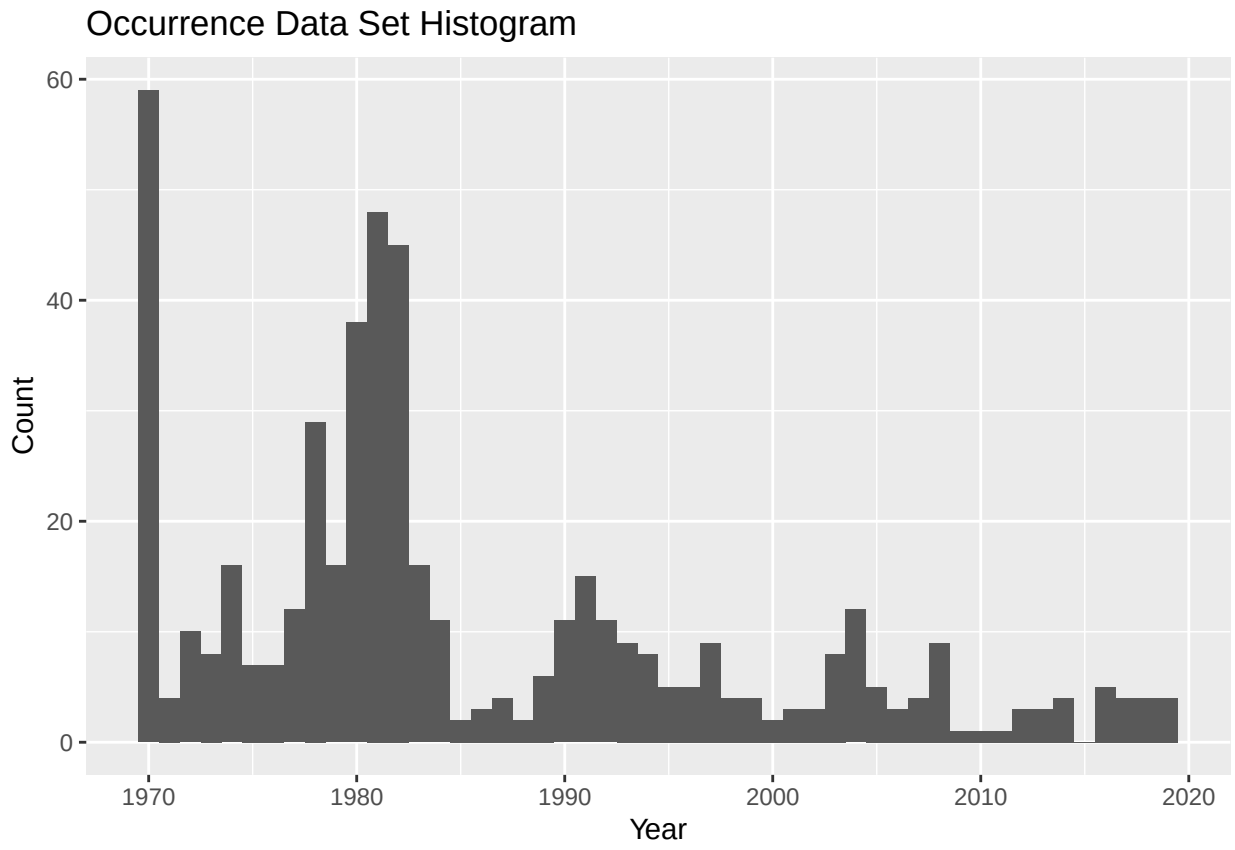
Habitat destruction and fragmentation is a leading cause of species decline, and human habitation is a leading cause of habitat destruction and fragmentation.
Historically, humans simply did not understand the impact that they were having on their environment. Over time we have become increasingly aware of the profound impact our actions have on the natural world, and the importance of taking those impacts into consideration. The more clearly and specifically we can quantify our most harmful practices, the better positioned we will be to change industry standards and create effective conservation legislation. This project will explore different types of residential housing, and different measures of quantity of new housing, to search for specific indicators of species decline.

Methods

Two data sets were collected and prepared for this project: an observed occurrences of rare species data set and a yearly permit survey data set. Unnecessary and irrelevant variables were removed and the data

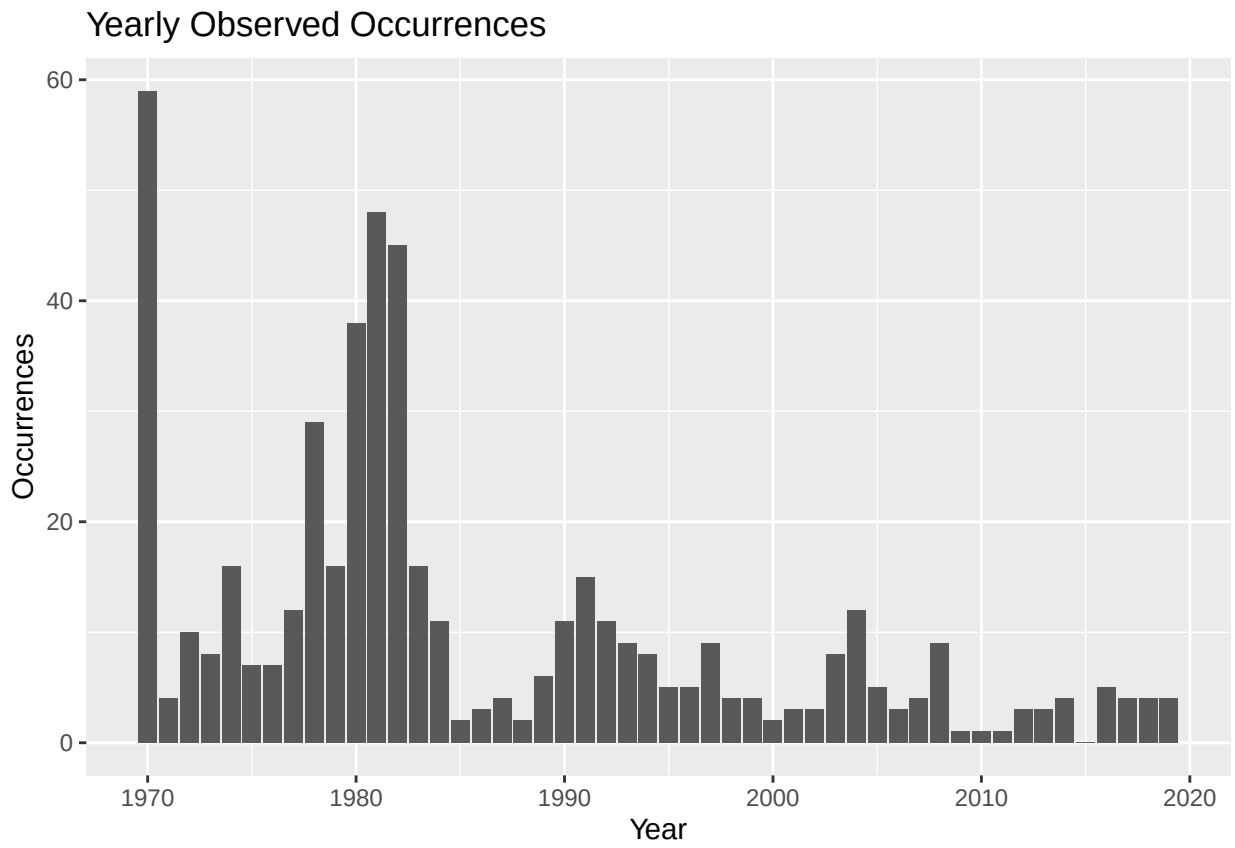
sets were conditionally restructured to match the desired species, time frame, and location. These decisions were based on several considerations, including missing data and number of observations. Once the basic structure was established I began exploratory analysis.

```
#### Occurrences Data Preliminary Preparation
# Import data and create subsets based on values of relevant variables.
occurrences <- read.delim("/Users/jonathanbunch/Documents/dsc520/final-project/data/drymarchon_occ/occu
occ_sub1 <- subset(occurrences, select = c(basisOfRecord, year, stateProvince,
                                           specificEpithet, issue))
occ_sub2 <- subset(occ_sub1, stateProvince == "Florida" & specificEpithet == "couperi"
                   & year >= 1970 & basisOfRecord == "HUMAN_OBSERVATION")
# Plot to see the distribution of sightings.
ggplot(data = occ_sub2, aes(x = year)) + geom_histogram(bins = 50) +
  ggtitle("Occurrence Data Set Histogram") + xlab("Year") + ylab("Count")
```




```
#### Restructure the Occurrences Data to Represent Sightings per Year
# Create a new data table that counts the sightings per year.
occ_count <- as.data.frame(table(occ_sub2[, "year"]))
# This method converted the year to a factor. This will convert it back to a number.
occ_count$Var1 <- as.numeric(as.character(occ_count$Var1))
# The year 2015 had no occurrences, so it was not included in our count data frame.
# We need to add it to the data frame with a zero value for frequency.
occ_count <- rbind(occ_count, c(2015, 0))
# The order() function will move the new row to the correct chronological position.
occ_count <- occ_count[order(occ_count$Var1), ]
# Plot to make sure the "sightings per year" data frame matches the histogram.
```

```
ggplot(data = occ_count, aes(x = Var1, y = Freq)) + geom_col() +
  ggtitle("Yearly Observed Occurrences") + xlab("Year") + ylab("Occurrences")
```




```
#### Permitting Data Preliminary Preparation
# Import data and create subsets based on values of relevant variables.
permits <- read_xlsx("/Users/jonathanbunch/Documents/dsc520/final-project/data/fl_permits.xlsx")
permits$year <- as.integer(permits$year)
permits_sub1 <- subset(permits, select = -c(`1_unit_buildings`, `2_unit_buildings`,
                                            `3_and_4_unit_buildings`, `5_plus_unit_buildings`,
                                            total_buildings))
permits_sub2 <- subset(permits_sub1, year >= 1970)
permits_sub2$year <- as.integer(permits_sub2$year)
# These final two variables were selected based on their correlation with the occurrences data.
permits_sub3 <- subset(permits_sub2, select = c(total_construction_valuation, `1_unit_valuation`))
```

Analysis began with trial and error experimentation through restructuring, visualization, and correlation testing. These tests lead to further exclusions of variables that did not appear to have statistical significance in this case.

```
cor(x = occ_count$Freq, y = permits_sub2)
```

```
##            year total_housing_units total_construction_valuation
## [1,] -0.5167775         0.008220874                   -0.3606783
##      1_unit_housing_units 1_unit_valuation 2_unit_housing_units
```

```
## [1,]          -0.1464775       -0.3837584            0.4909972
##      2_unit_valuation 3_and_4_unit_housing_units 3_and_4_unit_valuation
## [1,]       0.08935475                  0.4202969           -0.07335011
##      5_plus_unit_housing_units 5_plus_unit_valuation
## [1,]                0.09040393           -0.2571539
```

Due to the low sample size of the occurrences data set, I chose to aggregate the yearly sighting count into five year blocks. The two most promising variables from the permitting data set were aggregated into the same five year blocks and combined with the occurrences data to produce my final data set.

```
#### Aggregating Yearly Values into Five Year Totals.
# This function returns the sum of each five row block for a given variable.
blocks_func <- function(x) {
  return(c(sum(x[1:5]), sum(x[6:10]), sum(x[11:15]), sum(x[16:20]), sum(x[21:25]),
           sum(x[26:30]), sum(x[31:35]), sum(x[36:40]), sum(x[41:45]), sum(x[46:50])))
}
# Labels for each block of years.
block_labels <- c("1970-1974", "1975-1979", "1980-1984", "1985-1989", "1990-1994",
                  "1995-1999", "2000-2004", "2005-2009", "2010-2014", "2015-2019")
# New data frame with labels and sums of values for each five year period.
blocks_df <- data.frame(years = block_labels, occurrences = blocks_func(occ_count$Freq),
                        total_const_valuation = blocks_func(permits_sub3$total_construction_valuation),
                        one_unit_valuation = blocks_func(permits_sub3$`1_unit_valuation`))
# See what the five year block data frame looks like.
head(blocks_df)
```

```
##       years occurrences total_const_valuation one_unit_valuation
## 1 1970-1974          97              15455073            6090150
## 2 1975-1979          71              16852934           11896964
## 3 1980-1984         158              31898653           18317244
## 4 1985-1989          17              46521110           33811438
## 5 1990-1994          54              44802962           37709542
## 6 1995-1999          27              64728353           50437968
```

```
# Correlation matrix.
cor(blocks_df[,-1])
```

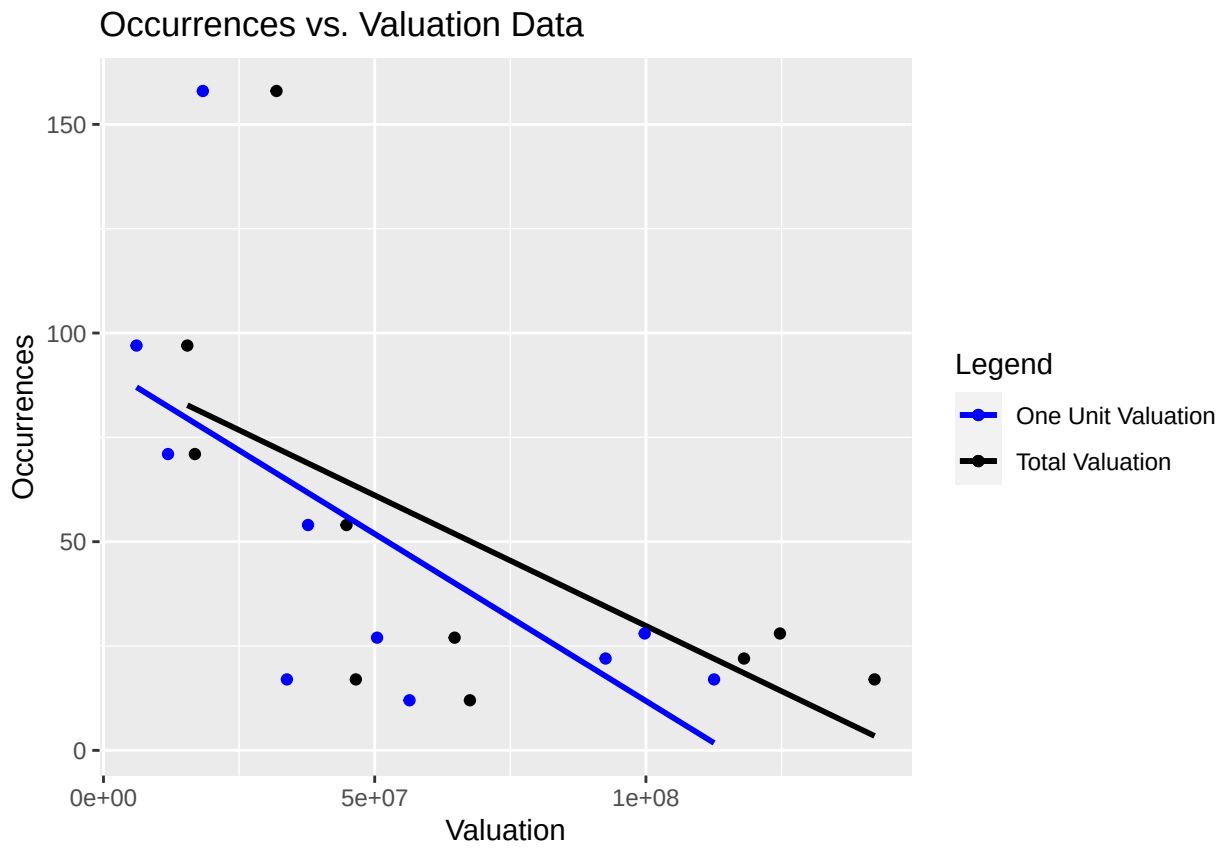
```
##                       occurrences total_const_valuation one_unit_valuation
## occurrences             1.0000000            -0.6121634         -0.6497453
## total_const_valuation  -0.6121634             1.0000000          0.9971958
## one_unit_valuation     -0.6497453             0.9971958          1.0000000
```

Once the data were prepared, I performed simple and multiple regression to create models of observed occurrences based on the two permitting variables. This resulted in three models: one for each predictor variable plus one that included both predictor variables. I then compared the three models to determine which, if any, was most effective.

```
# Create regression models for each predictor variable, as well as the combination of both.
tot_val_lm <- lm(occurrences ~ total_const_valuation, data = blocks_df)
one_unit_val_lm <- lm(occurrences ~ one_unit_valuation, data = blocks_df)
multi_lm <- lm(occurrences ~ total_const_valuation + one_unit_valuation, data = blocks_df)
# Plot the data with regression lines.
```

```
ggplot(data = blocks_df, aes(y = occurrences)) + geom_point(aes(x = total_const_valuation, color = "Tota
  geom_smooth(aes(x = total_const_valuation, color="Total Valuation"), method = "lm", se=FALSE) +
  geom_point(aes(x = one_unit_valuation, color = "One Unit Valuation")) +
  geom_smooth(aes(x = one_unit_valuation, color="One Unit Valuation"), method = "lm", se=FALSE) +
  ggtitle("Occurrences vs. Valuation Data") + xlab("Valuation") + ylab("Occurrences") +
  labs(color = "Legend") + scale_color_manual(values = c("Total Valuation" = "black", "One Unit Valuatio
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



Analysis

I began by examining the summary output for the three regression models. We can see from the summary that using the total construction valuation as a predictor creates a fairly weak model, with an R-squared value of about 0.375 and a significance value slightly above 0.05. The model using one unit construction valuation as the predictor variable is slightly better, with an R-squared value of about 0.422 and a significance value slightly below 0.05. The model using both predictor variables has a better fit than the other models, with an R-squared value of about 0.651, but the significance values are slightly above the 0.05 level.

Next, I used the gvlma library to assess the model assumptions. All models passed the assessments, indicating that the models do not violate any of the basic assumptions required for an effective regression model. Next, I used the "deletion" function from the same library to calculate deletion statistics for each model. This calculation indicates unusual values that may be having an undue influence on the model. The results

indicate that deletion of the third row of data could improve model performance. However, there is no obvious justification to remove that observation and doing so may be manipulative.

```
# View summaries of the three models.
summary(tot_val_lm)
```

```
##
## Call:
## lm(formula = occurrences ~ total_const_valuation, data = blocks_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.284 -21.381  -3.438  13.610  85.568
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            9.239e+01  2.289e+01   4.037  0.00375 **
## total_const_valuation -6.256e-07  2.857e-07  -2.190  0.05995 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.29 on 8 degrees of freedom
## Multiple R-squared:  0.3747, Adjusted R-squared:  0.2966
## F-statistic: 4.795 on 1 and 8 DF,  p-value: 0.05995
```

```
summary(one_unit_val_lm)
```

```
##
## Call:
## lm(formula = occurrences ~ one_unit_valuation, data = blocks_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -47.830 -21.232  -1.748  13.914  80.763
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          9.190e+01  2.095e+01   4.387  0.00233 **
## one_unit_valuation -8.007e-07  3.312e-07  -2.418  0.04201 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.77 on 8 degrees of freedom
## Multiple R-squared:  0.4222, Adjusted R-squared:  0.3499
## F-statistic: 5.845 on 1 and 8 DF,  p-value: 0.04201
```

```
summary(multi_lm)
```

```
##
## Call:
## lm(formula = occurrences ~ total_const_valuation + one_unit_valuation,
##     data = blocks_df)
```

```
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -54.79 -11.02   0.31  13.13  47.64
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            6.061e+01  2.275e+01   2.664   0.0323 *
## total_const_valuation  6.526e-06  3.051e-06   2.138   0.0698 .
## one_unit_valuation    -8.648e-06  3.680e-06  -2.350   0.0511 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.4 on 7 degrees of freedom
## Multiple R-squared:  0.6505, Adjusted R-squared:  0.5506
## F-statistic: 6.514 on 2 and 7 DF,  p-value: 0.02524
```

```r
# Assess model assumptions.
gvmodel_tot <- gvlma(tot_val_lm)
summary(gvmodel_tot)
```

```
##
## Call:
## lm(formula = occurrences ~ total_const_valuation, data = blocks_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.284 -21.381  -3.438  13.610  85.568
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            9.239e+01  2.289e+01   4.037  0.00375 **
## total_const_valuation -6.256e-07  2.857e-07  -2.190  0.05995 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 39.29 on 8 degrees of freedom
## Multiple R-squared:  0.3747, Adjusted R-squared:  0.2966
## F-statistic: 4.795 on 1 and 8 DF,  p-value: 0.05995
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = tot_val_lm)
##
##                      Value p-value                Decision
## Global Stat        5.3588  0.2524 Assumptions acceptable.
## Skewness           1.8905  0.1691 Assumptions acceptable.
## Kurtosis           0.4083  0.5229 Assumptions acceptable.
## Link Function      1.9949  0.1578 Assumptions acceptable.
## Heteroscedasticity 1.0651  0.3021 Assumptions acceptable.
```

```
gvmodel_one <- gvlma(one_unit_val_lm)
summary(gvmodel_one)
```

```
##
## Call:
## lm(formula = occurrences ~ one_unit_valuation, data = blocks_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -47.830 -21.232  -1.748  13.914  80.763
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         9.190e+01  2.095e+01   4.387  0.00233 **
## one_unit_valuation -8.007e-07  3.312e-07  -2.418  0.04201 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.77 on 8 degrees of freedom
## Multiple R-squared:  0.4222, Adjusted R-squared:  0.3499
## F-statistic: 5.845 on 1 and 8 DF,  p-value: 0.04201
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = one_unit_val_lm)
##
##                     Value p-value                Decision
## Global Stat        5.4565  0.2436 Assumptions acceptable.
## Skewness           1.5149  0.2184 Assumptions acceptable.
## Kurtosis           0.2803  0.5965 Assumptions acceptable.
## Link Function      2.6590  0.1030 Assumptions acceptable.
## Heteroscedasticity 1.0024  0.3167 Assumptions acceptable.
```

```
gvmodel_multi <- gvlma(multi_lm)
summary(gvmodel_multi)
```

```
##
## Call:
## lm(formula = occurrences ~ total_const_valuation + one_unit_valuation,
##     data = blocks_df)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -54.79 -11.02   0.31  13.13  47.64
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             6.061e+01  2.275e+01   2.664   0.0323 *
```

```
## total_const_valuation  6.526e-06  3.051e-06   2.138   0.0698 .
## one_unit_valuation    -8.648e-06  3.680e-06  -2.350   0.0511 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.4 on 7 degrees of freedom
## Multiple R-squared:  0.6505, Adjusted R-squared:  0.5506
## F-statistic: 6.514 on 2 and 7 DF,  p-value: 0.02524
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = multi_lm)
##
##                       Value p-value                Decision
## Global Stat        5.110979 0.27610 Assumptions acceptable.
## Skewness           0.089596 0.76469 Assumptions acceptable.
## Kurtosis           0.008137 0.92812 Assumptions acceptable.
## Link Function      3.639565 0.05642 Assumptions acceptable.
## Heteroscedasticity 1.373681 0.24118 Assumptions acceptable.
```

```
# Deletion statistics for each model.
gvmodel_tot_del <- deletion.gvlma(gvmodel_tot)
summary(gvmodel_tot_del)
```

```
##
## Global test deletion statistics.
##
## Linear Model:
##  lm(formula = occurrences ~ total_const_valuation, data = blocks_df)
##
## gvlma call:
##  gvlma(x = tot_val_lm)
##
##
## Summary values:
##                          Min.      1st Qu.       Median        Mean     3rd Qu.
## DeltaGlobalStat -35.91159509 -27.1392235 -20.7994438 -10.9154041   1.0767674
## GStatpvalue       0.12029407   0.2520908   0.3740622   0.3351329   0.4193153
## DeltaStat1      -97.51213984 -35.6923458 -16.3672751 -23.6607904  -9.4604476
## Stat1pvalue       0.11241537   0.1908543   0.2086062   0.2763151   0.2702017
## DeltaStat2      -87.64689284 -67.5221721 -58.0101368 -45.7673897 -35.1566615
## Stat2pvalue       0.43925770   0.6074830   0.6788580   0.6599002   0.7157621
## DeltaStat3      -47.25479405 -26.3859918 -12.0605288   5.1993993  -8.0162043
## Stat3pvalue       0.01874416   0.1755592   0.1853807   0.1815352   0.2266693
## DeltaStat4      -65.19504358 -42.1848891 -38.9336763  -5.1168666  21.0339144
## Stat4pvalue       0.11407191   0.2601568   0.4199958   0.3572783   0.4326301
##                       Max.
## DeltaGlobalStat  36.4482124
## GStatpvalue       0.4879316
## DeltaStat1       33.2942670
```

```
## Stat1pvalue       0.8283088
## DeltaStat2       46.5303641
## Stat2pvalue       0.8223125
## DeltaStat3      176.9630778
## Stat3pvalue       0.3049937
## DeltaStat4      134.4347611
## Stat4pvalue       0.5426255
##
##
## Unusual observations for Global Stat:
## [1] Delta Global Stat (%)  Global Stat p-value
## <0 rows> (or 0-length row.names)
##
##
## Unusual observations for Directional Stat1:
##   Delta Directional Stat1 (%)  Directional Stat1 p-value
## 3                   -97.51214                  0.8283088
##
##
## Unusual observations for Directional Stat2:
## [1] Delta Directional Stat2 (%)  Directional Stat2 p-value
## <0 rows> (or 0-length row.names)
##
##
## Unusual observations for Directional Stat3:
##   Delta Directional Stat3 (%)  Directional Stat3 p-value
## 3                    176.9631                 0.01874416
##
##
## Unusual observations for Directional Stat4:
## [1] Delta Directional Stat4 (%)  Directional Stat4 p-value
## <0 rows> (or 0-length row.names)
```

```
gvmodel_one_del <- deletion.gvlma(gvmodel_one)
summary(gvmodel_one_del)
```

```
##
## Global test deletion statistics.
##
## Linear Model:
##  lm(formula = occurrences ~ one_unit_valuation, data = blocks_df)
##
## gvlma call:
##  gvlma(x = one_unit_val_lm)
##
##
## Summary values:
##                          Min.      1st Qu.       Median        Mean      3rd Qu.
## DeltaGlobalStat -32.68079601 -23.1118982 -19.6716831 -10.0460602  -1.4212074
## GStatpvalue       0.12416486   0.2548277   0.3566429   0.3158675   0.3802452
## DeltaStat1      -90.07404302 -37.2813943 -18.0210544 -23.0337368  -7.5210745
## Stat1pvalue       0.17027818   0.2367206   0.2652249   0.3101258   0.3298807
## DeltaStat2      -93.64690613 -75.3028558 -60.8535972 -48.7966128 -16.2836893
## Stat2pvalue       0.58163482   0.6294939   0.7404880   0.7256346   0.7924677
```

```
## DeltaStat3      -36.99918455 -21.6530776  -9.7609802  -0.4202123  -1.1933317
## Stat3pvalue       0.01980422   0.1052788   0.1213834   0.1196845   0.1495819
## DeltaStat4      -66.82487660 -44.9307456 -37.0036600  -5.1159584  18.9022470
## Stat4pvalue       0.12430481   0.2779885   0.4268429   0.3718557   0.4574976
##                       Max.
## DeltaGlobalStat  32.5227616
## GStatpvalue       0.4520206
## DeltaStat1       24.1368459
## Stat1pvalue       0.6981879
## DeltaStat2        8.3062581
## Stat2pvalue       0.8938381
## DeltaStat3      104.1793154
## Stat3pvalue       0.1955670
## DeltaStat4      135.6556656
## Stat4pvalue       0.5641619
##
##
## Unusual observations for Global Stat:
## [1] Delta Global Stat (%)  Global Stat p-value
## <0 rows> (or 0-length row.names)
##
##
## Unusual observations for Directional Stat1:
##   Delta Directional Stat1 (%)  Directional Stat1 p-value
## 3                   -90.07404                  0.6981879
##
##
## Unusual observations for Directional Stat2:
## [1] Delta Directional Stat2 (%)  Directional Stat2 p-value
## <0 rows> (or 0-length row.names)
##
##
## Unusual observations for Directional Stat3:
##   Delta Directional Stat3 (%)  Directional Stat3 p-value
## 3                    104.1793                 0.01980422
##
##
## Unusual observations for Directional Stat4:
## [1] Delta Directional Stat4 (%)  Directional Stat4 p-value
## <0 rows> (or 0-length row.names)
```

```
gvmodel_multi_del <- deletion.gvlma(gvmodel_multi)
summary(gvmodel_multi_del)
```

```
##
## Global test deletion statistics.
##
## Linear Model:
##  lm(formula = occurrences ~ total_const_valuation + one_unit_valuation,      data = blocks_df)
##
## gvlma call:
##  gvlma(x = multi_lm)
##
##
```

```
## Summary values:
##                          Min.      1st Qu.       Median         Mean
## DeltaGlobalStat -29.70682975 -20.86926491 -15.34840098   0.12183139
## GStatpvalue       0.09544758   0.17215722   0.36375738   0.30479010
## DeltaStat1      -57.49852686 -19.29732024  88.82479739 209.96584004
## Stat1pvalue       0.30149759   0.61087545   0.68207488   0.65504050
## DeltaStat2      -99.98490194 -75.03384862  14.13401020 473.93726819
## Stat2pvalue       0.55462036   0.86004495   0.92367214   0.89022067
## DeltaStat3      -53.47880732 -18.32106863  -8.49718478  -2.49849542
## Stat3pvalue       0.02408510   0.03972079   0.06801444   0.07169354
## DeltaStat4      -49.68331162 -42.80448642 -33.28629277  -9.42891309
## Stat4pvalue       0.10326523   0.23688675   0.33879867   0.29472552
##                      3rd Qu.         Max.
## DeltaGlobalStat  26.15702803   54.4988786
## GStatpvalue       0.40008804    0.4639284
## DeltaStat1      189.23100341 1091.5243961
## Stat1pvalue       0.78802226    0.8452833
## DeltaStat2      298.65505653 4190.4426289
## Stat2pvalue       0.96411316    0.9991156
## DeltaStat3       16.24108339   39.8105081
## Stat3pvalue       0.08509906    0.1931838
## DeltaStat4        1.86755901   93.2308775
## Stat4pvalue       0.37563743    0.4057596
##
##
## Unusual observations for Global Stat:
## [1] Delta Global Stat (%)  Global Stat p-value
## <0 rows> (or 0-length row.names)
##
##
## Unusual observations for Directional Stat1:
##   Delta Directional Stat1 (%)  Directional Stat1 p-value
## 3                    1091.524                  0.3014976
##
##
## Unusual observations for Directional Stat2:
##   Delta Directional Stat2 (%)  Directional Stat2 p-value
## 4                    4190.443                  0.5546204
##
##
## Unusual observations for Directional Stat3:
## [1] Delta Directional Stat3 (%)  Directional Stat3 p-value
## <0 rows> (or 0-length row.names)
##
##
## Unusual observations for Directional Stat4:
## [1] Delta Directional Stat4 (%)  Directional Stat4 p-value
## <0 rows> (or 0-length row.names)
```

Implications

The results of this modeling indicate that a multiple regression model using yearly total- and one-unit construction valuation as predictor variables could predict the yearly observational occurrences of the Eastern

Indigo snake in Florida. The R-squared value indicates that these variables could theoretically account for about 55% of the variation in yearly observational occurrences when applied as a general model.

Limitations

The most obvious limitation of this analysis is the very small sample size of the observational occurrences data set. Five year time blocks had as little as 12 occurrences, and aggregating the data this way resulted in only 10 observations. The data itself has many potential issues as well. It is based on human observation and identification of species, often by non-experts.

Conclusion

The data available regarding the Eastern Indigo snake is, not surprisingly, very limited. Species sighting data is relatively laborious to create, especially if the curators of the data are attempting to verify the species identification and other pertinent information. After removing all of the observations that had issues or were otherwise not usable, I found that the sample size had fallen well below what I would consider sufficient for meaningful analysis. That being said, there were some statistically significance patterns between species occurrences and construction permitting data, suggesting that there could be legitimacy to my hypothesis. This was a very interesting and educational project, and I hope to continue this avenue of research in the future.

References

Field, A., J. Miles, and Z. Field. 2012. *Discovering Statistics Using R*. SAGE Publications. https://books.google.com/books?id=wd2K2zC3swIC.

"GBIF Occurrence Download." n.d. https://doi.org/10.15468/dl.2jw9ek.

Lander, J. P. 2014. *R for Everyone: Advanced Analytics and Graphics*. Addison-Wesley Data and Analytics Series. Addison-Wesley. https://books.google.com/books?id=3eBVAgAAQBAJ.

"U.S. Census Bureau: New Privately-Owned Housing Units Authorized by Building Permits in Permit-Issuing Places in the State of: Florida." n.d. https://www.census.gov/construction/bps/stateannual.html.