Lab 7

Kent Codding

2023-10-27

#read in BACS data

df <- read.csv("BACS(1).csv")

# Step 1

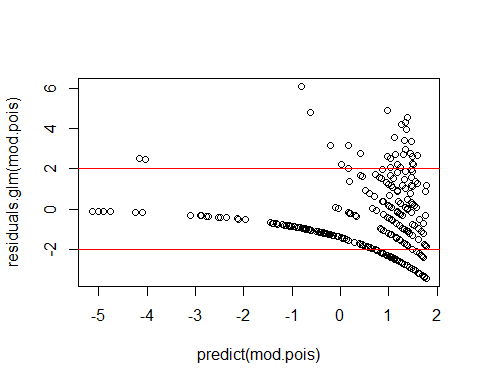
mod.pois <- glm(df$BACS ~ df$years\_since\_burn +  
df$area\_small\_pines +  
df$hardwood\_cov\_perc   
 , family = "poisson")  
summary(mod.pois)

##   
## Call:  
## glm(formula = df$BACS ~ df$years\_since\_burn + df$area\_small\_pines +   
## df$hardwood\_cov\_perc, family = "poisson")  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -3.4363 -1.4353 -0.8066 0.3630 6.0875   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.7756590 0.0648134 27.397 < 2e-16 \*\*\*  
## df$years\_since\_burn -0.1393793 0.0187578 -7.430 1.08e-13 \*\*\*  
## df$area\_small\_pines -0.0008761 0.0002511 -3.489 0.000484 \*\*\*  
## df$hardwood\_cov\_perc -0.0604147 0.0056306 -10.730 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for poisson family taken to be 1)  
##   
## Null deviance: 1264.99 on 286 degrees of freedom  
## Residual deviance: 873.11 on 283 degrees of freedom  
## AIC: 1352  
##   
## Number of Fisher Scoring iterations: 6

numerically, residual deviance / df > 1.5, which provides evidence of overdispersion

## visualization

plot(residuals.glm(mod.pois) ~ predict(mod.pois))  
abline(h = 2, col = "red")  
abline(h = -2, col = "red")

 The variance changes within the range of predicted values, suggesting that the variance is not fixed. Therefore, overdispersion is present.

# Step 2

library(countreg)

## Loading required package: MASS

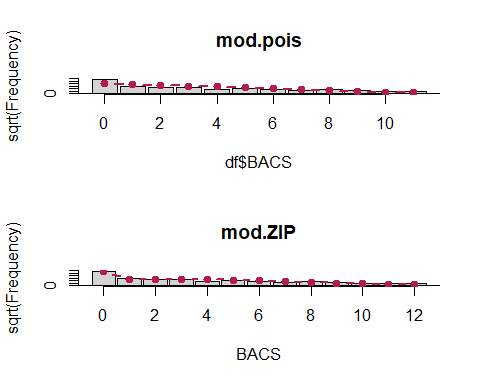
## Loading required package: Formula

## Warning: package 'Formula' was built under R version 4.1.3

mod.ZIP <- countreg::zeroinfl(  
 BACS ~ years\_since\_burn + area\_small\_pines + hardwood\_cov\_perc | #count   
 Shrub, #binomial  
 data = df, dist = "poisson")  
summary(mod.ZIP)

##   
## Call:  
## countreg::zeroinfl(formula = BACS ~ years\_since\_burn + area\_small\_pines +   
## hardwood\_cov\_perc | Shrub, data = df, dist = "poisson")  
##   
## Pearson residuals:  
## Min 1Q Median 3Q Max   
## -1.6600 -0.7170 -0.4434 0.3855 8.5957   
##   
## Count model coefficients (poisson with log link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.8416339 0.0756841 24.333 < 2e-16 \*\*\*  
## years\_since\_burn -0.0750090 0.0235261 -3.188 0.00143 \*\*   
## area\_small\_pines -0.0004292 0.0002869 -1.496 0.13457   
## hardwood\_cov\_perc -0.0463830 0.0064724 -7.166 7.71e-13 \*\*\*  
##   
## Zero-inflation model coefficients (binomial with logit link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.936578 0.474836 -6.184 6.23e-10 \*\*\*  
## Shrub 0.048552 0.008082 6.007 1.89e-09 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1   
##   
## Number of iterations in BFGS optimization: 13   
## Log-likelihood: -558.8 on 6 Df

par(mfrow = c(2,1))  
  
rootogram(mod.pois, style = "standing")  
rootogram(mod.ZIP, style = "standing")

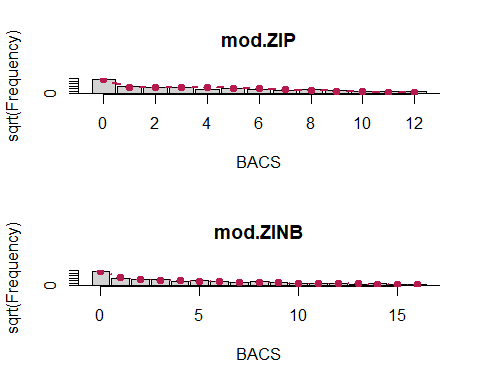
 The ZIP model does a much better job than the Poisson model at predicting the zeros within the observed data. Thus, the ZIP model is more appropriate.

# Step 3a

mod.ZINB <- countreg::zeroinfl(BACS ~  
 years\_since\_burn + area\_small\_pines + hardwood\_cov\_perc | #count   
 Shrub, # binomial  
 data = df,   
 dist = "negbin" )  
summary(mod.ZINB)

##   
## Call:  
## countreg::zeroinfl(formula = BACS ~ years\_since\_burn + area\_small\_pines +   
## hardwood\_cov\_perc | Shrub, data = df, dist = "negbin")  
##   
## Pearson residuals:  
## Min 1Q Median 3Q Max   
## -1.0370 -0.6081 -0.3842 0.2461 7.6480   
##   
## Count model coefficients (negbin with log link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 2.0190935 0.1493744 13.517 < 2e-16 \*\*\*  
## years\_since\_burn -0.1688469 0.0391204 -4.316 1.59e-05 \*\*\*  
## area\_small\_pines -0.0007989 0.0005317 -1.503 0.133   
## hardwood\_cov\_perc -0.0509518 0.0073437 -6.938 3.97e-12 \*\*\*  
## Log(theta) 0.4711089 0.2143754 2.198 0.028 \*   
##   
## Zero-inflation model coefficients (binomial with logit link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.45616 0.90610 -4.918 8.75e-07 \*\*\*  
## Shrub 0.06221 0.01302 4.779 1.76e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1   
##   
## Theta = 1.6018   
## Number of iterations in BFGS optimization: 15   
## Log-likelihood: -500.4 on 7 Df

par(mfrow = c(2,1))  
  
rootogram(mod.ZIP, style = "standing")  
rootogram(mod.ZINB, style = "standing")

 Graphically, the ZINB does a better job of predicting the lower, nonzero observed values than the ZIP model due to the presence of an additional parameter to address overdispersion. Numerically, the Log-likelihood is higher and the years\_since\_burn predictor has an additional level of significance in the count portion of the model for the ZINB.

# Step 4

## drop area\_small\_pines from count model

mod.ZINB2 <- countreg::zeroinfl(BACS ~  
 years\_since\_burn + hardwood\_cov\_perc | #count   
 Shrub, # binomial  
 data = df,   
 dist = "negbin" )  
  
library(lmtest)

## Warning: package 'lmtest' was built under R version 4.1.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 4.1.3

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

lrtest(mod.ZINB, mod.ZINB2)

## Likelihood ratio test  
##   
## Model 1: BACS ~ years\_since\_burn + area\_small\_pines + hardwood\_cov\_perc |   
## Shrub  
## Model 2: BACS ~ years\_since\_burn + hardwood\_cov\_perc | Shrub  
## #Df LogLik Df Chisq Pr(>Chisq)  
## 1 7 -500.37   
## 2 6 -501.49 -1 2.2303 0.1353

summary(mod.ZINB2)

##   
## Call:  
## countreg::zeroinfl(formula = BACS ~ years\_since\_burn + hardwood\_cov\_perc |   
## Shrub, data = df, dist = "negbin")  
##   
## Pearson residuals:  
## Min 1Q Median 3Q Max   
## -1.0347 -0.6026 -0.3854 0.2520 7.3101   
##   
## Count model coefficients (negbin with log link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.892114 0.121572 15.564 < 2e-16 \*\*\*  
## years\_since\_burn -0.162969 0.039039 -4.174 2.99e-05 \*\*\*  
## hardwood\_cov\_perc -0.050624 0.007374 -6.865 6.65e-12 \*\*\*  
## Log(theta) 0.468567 0.214893 2.180 0.0292 \*   
##   
## Zero-inflation model coefficients (binomial with logit link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.46845 0.91291 -4.895 9.84e-07 \*\*\*  
## Shrub 0.06290 0.01301 4.835 1.33e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1   
##   
## Theta = 1.5977   
## Number of iterations in BFGS optimization: 14   
## Log-likelihood: -501.5 on 6 Df

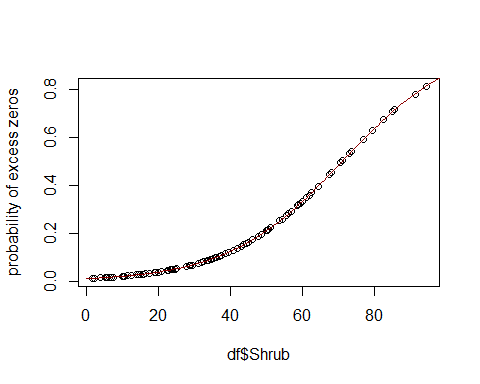
having area\_small\_pines in the model does not significantly improve the model, so it should be dropped: mod.ZINB2 is the better model.

The coefficient for shrub is positive and significant within the binomial portion of the model, so shrub increases the probability of false zeros.

For the counts of BACS, only years\_since\_burn and hardwood\_cov\_perc significantly affect the counts of BACS.

# Question 4

#first make new data set holding all the coviariates constant  
  
ND.Shrub <- expand.grid(list(Shrub = 0:100,   
 years\_since\_burn = median(df$years\_since\_burn),  
 hardwood\_cov\_perc = median(df$hardwood\_cov\_perc)))  
  
#plot raw data  
plot(predict(mod.ZINB2, type = "zero") ~ df$Shrub,  
 ylab = "probability of excess zeros")  
   
#draw fit line using dummy data set  
lines(ND.Shrub$Shrub,   
 predict(mod.ZINB2, type = "zero", newdata = ND.Shrub),  
 lwd = 1.5, col = "darkred")

 # Question 5

library(pscl)

## Warning: package 'pscl' was built under R version 4.1.3

## Registered S3 methods overwritten by 'pscl':  
## method from   
## print.zeroinfl countreg  
## print.summary.zeroinfl countreg  
## summary.zeroinfl countreg  
## coef.zeroinfl countreg  
## vcov.zeroinfl countreg  
## logLik.zeroinfl countreg  
## predict.zeroinfl countreg  
## residuals.zeroinfl countreg  
## fitted.zeroinfl countreg  
## terms.zeroinfl countreg  
## model.matrix.zeroinfl countreg  
## extractAIC.zeroinfl countreg  
## print.hurdle countreg  
## print.summary.hurdle countreg  
## summary.hurdle countreg  
## coef.hurdle countreg  
## vcov.hurdle countreg  
## logLik.hurdle countreg  
## predict.hurdle countreg  
## residuals.hurdle countreg  
## fitted.hurdle countreg  
## terms.hurdle countreg  
## model.matrix.hurdle countreg  
## extractAIC.hurdle countreg

## Classes and Methods for R developed in the  
## Political Science Computational Laboratory  
## Department of Political Science  
## Stanford University  
## Simon Jackman  
## hurdle and zeroinfl functions by Achim Zeileis

##   
## Attaching package: 'pscl'

## The following objects are masked from 'package:countreg':  
##   
## hurdle, hurdle.control, hurdletest, zeroinfl, zeroinfl.control

mod.ZANB <- hurdle(BACS ~  
 years\_since\_burn + hardwood\_cov\_perc | #count   
 Shrub, # binomial  
 data = df,   
 dist = "negbin" )  
summary(mod.ZANB)

##   
## Call:  
## hurdle(formula = BACS ~ years\_since\_burn + hardwood\_cov\_perc | Shrub,   
## data = df, dist = "negbin")  
##   
## Pearson residuals:  
## Min 1Q Median 3Q Max   
## -1.2100 -0.6341 -0.3707 0.3486 3.5548   
##   
## Count model coefficients (truncated negbin with log link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.599928 0.129300 12.374 <2e-16 \*\*\*  
## years\_since\_burn -0.050212 0.043074 -1.166 0.2437   
## hardwood\_cov\_perc -0.022845 0.009046 -2.526 0.0116 \*   
## Log(theta) 0.607904 0.266419 2.282 0.0225 \*   
## Zero hurdle model coefficients (binomial with logit link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.910824 0.314991 6.066 1.31e-09 \*\*\*  
## Shrub -0.039770 0.006271 -6.342 2.27e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1   
##   
## Theta: count = 1.8366  
## Number of iterations in BFGS optimization: 16   
## Log-likelihood: -531.2 on 6 Df

summary(mod.ZINB2)

##   
## Call:  
## countreg::zeroinfl(formula = BACS ~ years\_since\_burn + hardwood\_cov\_perc |   
## Shrub, data = df, dist = "negbin")  
##   
## Pearson residuals:  
## Min 1Q Median 3Q Max   
## -1.0347 -0.6026 -0.3854 0.2520 7.3101   
##   
## Count model coefficients (negbin with log link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 1.892114 0.121572 15.564 < 2e-16 \*\*\*  
## years\_since\_burn -0.162969 0.039039 -4.174 2.99e-05 \*\*\*  
## hardwood\_cov\_perc -0.050624 0.007374 -6.865 6.65e-12 \*\*\*  
## Log(theta) 0.468567 0.214893 2.180 0.0292 \*   
##   
## Zero-inflation model coefficients (binomial with logit link):  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -4.46845 0.91291 -4.895 9.84e-07 \*\*\*  
## Shrub 0.06290 0.01301 4.835 1.33e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1   
##   
## Theta = 1.5977   
## Number of iterations in BFGS optimization: 14   
## Log-likelihood: -501.5 on 6 Df

## similarities and differences

Both the ZINB and ZANB recognize Shrub as a significant predictor and have a similar log likelihood, although the ZINB’s log likelihood is slightly higher. However, the ZINB recognizes both years\_since\_burn and hardwood\_cov\_perc as significant predictors but the ZANB recognized hardwood\_cov\_perc as less significant and did not recognize years\_since\_burn as any level of significance within the count portion of the model.

## Interpretation

When interpreting the data, the ZINB suggests that both the count process and the process of shrubs obstructing bird observers contributed to zeros. On the other hand, the ZANB suggests that only the shrubs contributed to zeros within the data.

## final decision

Given that it is possible for a sparrow to not be present, I believe that the ZINB is more appropriate for these data as the count process could definitely explain at least some zeros. Additionally, the study comparing the zero-inflated and zero-affected models by Dr. Feng recommends that zero-inflated models are better to use when covariates are continuous, and in this case, all covariates are continuous.