Count Data Modeling

Serra Uzun, MSDS_410 FALL 2020
11/22/2020

1. Introduction

Count data models are a common form of statistical modeling when the dataset's response variable is counts. The response variable that is 'count' is discrete and commonly hold the information of 'number of' something, such as visits, people, trips, etc. In this assignment, we will perform count data regression models, which are Poisson, Negative Binomial, Hurdle, Zero-Inflated Regression Models on the Medical Care dataset where the response variable is the number of physician office visits. Instead of a traditional EDA, we will perform a Backward Variable Selection Model to determine the variables that are best fit to our count data regression models and assess the models listed above by comparing various error, model fit, and accuracy metrics.

2. The Dataset and Data Split

2.1 The Dataset

The Medical Care dataset consists of 22 variables, which are all numeric, and 4406 observations. Out of 14 variables, there is one response variable, ofp, and 21 predictor variables related to the individual's medical and personal information. Below is the list of variables and their types:

- ofp (discrete) – Response Variable
- ofnp (discrete)
- opp (discrete)
- opnp (discrete)
- emr (discrete)
- hosp (discrete)
- exclhlth (binary)
- poorhlth (binary)
- numchron (discrete)
- adldiff (binary)
- noreast (binary)
- midwest (binary)
- west (binary)
- age (continuous)
- black (binary)
- male (binary)
- married (binary)
- school (discrete)
- faminc (continuous)
- employed (binary)
- privins (binary)
- medicaid (binary)

Moving forward, we will remove ofnp, opp, opnp, emr and hosp from our dataset. This change leaves us with 16 predictor variables instead of 21 that were a part of the raw dataset.

2.2 The Train/Test Split

Before beginning our predictive model, we will perform a 50/50 split on our dataset consisting of 4,406 observations. This split aims to 'train' our model with 50 percent of the sample data and then 'test' it with the test dataset, which is also 50 percent of the sample data. With the 50/50 split, we continue our model with a train dataset of 2,203 observations and a test dataset of 2,203 observations. While we will mainly use the train data for our models, we will be tapping back to the test data to cross-validate our models in the following sections.

3. Backward Variable Selection

Instead of a traditional exploratory data analysis that we perform at the beginning of all analyses, we will be diving directly into the variable selection and will use the 'backward' variable selection method for this. As the upper limit for the backward variable selection is the full model with all predictor variables and the lower limit will be the intercept model. The backward variable selection model stepAIC result is presented below: (Please see the Appendix for backward.lm summary)

Step: AIC=17610.9

ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west + age + male + school + faminc + privins + medicaid

		Df	Deviance	AIC
	<none>		11223	17611
-	midwest	1	11225	17611
-	faminc	1	11229	17616
-	male	1	11234	17621
-	age	1	11240	17627
-	noreast	1	11251	17638
-	adldiff	1	11258	17644
-	school	1	11280	17667
-	west	1	11284	17670
-	medicaid	1	11294	17680
-	exclhlth	1	11320	17707
-	poorhlth	1	11324	17710
-	privins	1	11429	17816
-	numchron	1	11961	18347

Table 1: Backward Variable Selection Model Results

The backward variable selection linear model results show that 3 of the predictor variables in the train dataset have been eliminated: black, married, and employed. Per the output presented above, the variables we will be using for the remainder of this analysis are exclhlth, poorhlth, numchron, adldiff, noreast, midwest, west, age, male, school, faminc, privins, Medicaid.

4. Count Data Modeling with In-Sample data

As our response variable, ofp, is a count and majority of our predictor variables are discrete, we will be using Poisson Regression model, explore running the same Poisson Regression with dispersion, then followed by Negative Binomial, Hurdle and Zero-inflated Regression models. A further description of each modeling approach will be discussed under each sub-section within Section 4 while the model summary and key findings, such as Mean Squared Error, Mean Absolute Error, Mean Absolute Percentage Error, AIC, BIC and +/-2 Grade Percentage Rate, will be reported.

4.1 Poisson Regression Model

The Poisson Regression generalized linear model is the most common model used to analyze the rates when the response variable is a discrete count type data. The model uses the 'Poisson' family and assumes a Poisson distribution of the response variable and where predictor variables are generally discrete. The model summary is attached to the appendix, and the key results are presented below in Table 2.

	MSE	MAE	MAPE	Grade %	AIC	BIC
Poisson Regression	39.167	4.056	Inf	0.310	17610.9	17690.7

Table 2: Poisson Regression Model Results

The Poisson Regression Model results show an MSE of 39.167, an MAE of 4.056, and MAPE as Infinity. The reason for our model returning a positive Inf as MAPE is because MAPE is calculated by the mean of the absolute of actual minus predicted values divided by actual values, and since we have observations within our actuals that are zero, the MAPE output comes back as Inf. Per the computation of MAPE and the nature of our response variable, we will be getting MAPE of Inf with all our models, which makes MAPE not a useful metric for our analysis.

In addition to Error related results presented in Table 2, the Poisson Regression model got an AIC of 17610.9 and BIC of 17690.7. Also, the +/-2 Grade Percentage Rate is shown to be 31%, which indicates that 31% of the predicted values are within +/-2% of the actual values.

4.2 Poisson Regression Model with Dispersion

While the typical Poisson Regression model has only one free parameter and does not allow the adjustment of the variance separate from the mean, in this section, we will bring overdispersion into the Poisson Regression model presented in Section 4.1. As we change the 'family' within the generalized linear model, we are running in this section from "Poisson" to "quasi-poisson" we see that the dispersion increases from 1 to 6.405, meaning the variance is not limited and gets significantly higher. The model summary is attached to the appendix, and the key results are presented below in Table 3.

	MSE	MAE	MAPE	Grade %	AIC	BIC
Poisson Regression with Dispersion	39.167	4.056	Inf	0.310	NA	NA

Table 3: Poisson Regression with Dispersion Model Results

Model performance comparable metrics in the case of 'normal' Poisson distribution, and the case of our Poisson Model with over-dispersion, AIC, and BIC could not be computed. Additionally, the results for MSE, MAE, MAPE, and Grade Percentage are identical with the Poisson Regression Model from Section 4.1, which indicates that the Poisson Regression with Dispersion is problematic and not an ideal model to be used with our dataset.

4.3 Negative Binomial Regression Model

Compared to the typical Poisson Regression Model, Negative Binomial Regression has looser restrictions on dispersion and assumes that variance exceeds the conditional mean. As it is a generalized linear model that focuses on the count data's overdispersion, it is likely to perform better with our dataset than Poisson Regression with dispersion. The model summary is attached to the appendix, and the key results are presented below in Table 4.

	MSE	MAE	MAPE	Grade %	AIC	BIC
Negative Binomial Regression	39.786	4.074	Inf	0.320	12114.9	12200.4

Table 4: Negative Binomial Regression Model Results

Per Table 4, we see that the Negative Binomial Regression Model has an MSE of 39.786, MAE of 4.074, MAPE of Inf, Grade Percentage of 32%, AIC of 12114.9, and finally, BIC score of 12200.4. With the Negative Binomial Regression Model, we were able to obtain the AIC and BIC scores, despite having over-dispersion. The MAPE remained Inf as there was no change to the response variable.

4.4 Hurdle Regression Model

A Hurdle Regression Model has two processes to it, where in one of them, it processes zero counts, and in another, it processes positive counts. These two processes are not constrained to be the same, and it uses Bernoulli probability governs the outcome of the zero or positive count. The model summary is attached to the appendix, and the key results are presented below in Table 5.

	MSE	MAE	MAPE	Grade %	AIC	BIC
Hurdle Regression	38.763	4.033	Inf	0.318	15958.6	16118.2

Table 5: Hurdle Regression Model Results

Per Table 5, we see that the Hurdle Regression Model has an MSE of 38.763, MAE of 4.033, MAPE of Inf, Grade Percentage of 31.8%, AIC of 15958.6, and finally, BIC score of 16118.2. The Hurdle Regression Model seems to have relatively lower Error metrics than previous models yet

have higher AIC and BIC scores. This indicates that while the error in the model's performance has decreased, the likelihood of the model being true has decreased.

4.5 Zero-Inflated Regression Model

The final count data regression model we will be running with our in-sample dataset is the Zero-Inflated Regression Model. As understandable from its name, the Zero-Inflated Regression Model is used to model count data when there are excessive zero counts, and these zero counts are modeled independently. The model summary is attached to the appendix, and the key results are presented below in Table 6.

	MSE	MAE	MAPE	Grade %	AIC	BIC
Zero-Inflated Regression	38.767	4.033	Inf	0.318	15958.3	16117.8

Table 6: Zero-Inflated Regression Model Results

Per Table 6, we see that the Zero-Inflated Regression Model has an MSE of 38.767, MAE of 4.033, MAPE of Inf, Grade Percentage of 31.8%, AIC of 15958.3, and finally, BIC score of 16117.8. While Zero-Inflated Regression Model results sign similarities with the Hurdle Regression Model results at first glance, in the following sections, we will (a) look at the same models using the out-sample dataset compare all model outputs.

4.6 In-Sample Model Comparison

The comparison of error (Mean Squared Error, Mean Absolute Error, and Mean Absolute Percentage Error) and model fit (Grade Percentage, AIC, and BIC) metrics will help us determine the best performing model with an in-sample dataset, which is half of the raw dataset. Presented in Table 7 below, the metrics mentioned below are displayed for all five regression models conducted with the in-sample dataset.

	MSE	MAE	MAPE	Grade %	AIC	BIC
Poisson Regression	39.167	4.056	Inf	0.310	17610.9	17690.7
Poisson Regression with Dispersion	39.167	4.056	Inf	0.310	NA	NA
Negative Binomial Regression	39.786	4.074	Inf	0.320	12114.9	12200.4
Hurdle Regression	38.763	4.033	Inf	0.318	15958.6	16118.2
Zero-Inflated Regression	38.767	4.033	Inf	0.318	15958.3	16117.8

Table 7: Regression Model with In-sample Dataset Results

Firstly, as mentioned previously, each model's MAPE came back as Inf simply due to having zeros in the actual values. We determined that MAPE is not a useful metric for us to evaluate in this context. Secondly, Poisson Regression with Dispersion has not provided us with successful output. The dispersion within a part of the Poisson Regression is not optimal for incorporating dispersion into regression modeling. Instead, Negative Binomial Regression proved to be a better model in incorporating dispersion into our calculations and predictions.

When we observe the Mean Squared Error and Mean Absolute Error of each model, we can see that Negative Binomial Regression has the highest values for both these metrics, while Hurdle Regression has the lowest. On the other hand, if we look at the Grade Percentage, AIC, and BIC metrics results of each model, we can see that Negative Binomial Regression has the lower AIC and BIC together with the highest Grade Percentage. While Negative Binomial Regression was not the best performing model when MSE and MAE were taken into consideration, it has the highest likelihood to fit the out-sample dataset best. Additionally, when other regression models' Grade Percentage, AIC, and BIC results are reviewed, we see that the Poisson Regression model has the highest AIC and BIC as well as the lowest Grade Percentage, Hurdle and Zero-Inflated Regression Models have very similar, almost identical results.

Even though it is early to select the best performing model, per the results of the regression models conducted with the in-sample dataset, we can say that Negative Binomial, Hurdle, and Zero-Inflated Regression models are likely to perform best with the out-sample dataset.

5. Out-Sample Modeling and Comparison

With the aim to validate our models and determine which of them performs best with a dataset that it has not met before, we conduct all the regression models discussed in Section 4 with the out-sample dataset, which is half of our original dataset that we put aside at the beginning of our analysis.

	MSE	MAE	MAPE	Grade %
Poisson Regression	43.224	4.261	Inf	0.308
Poisson Regression with Dispersion	43.224	4.261	Inf	0.308
Negative Binomial Regression	43.510	4.275	Inf	0.309
Hurdle Regression	43.156	4.251	Inf	0.311
Zero-Inflated Regression	43.157	4.251	Inf	0.310

Table 8: Results of Models ran with out-sample data

Firstly, also visible from Table 8 above, we are not computing AIC and BIC scores for the out-sample models. As AIC and BIC are both penalized-likelihood criteria, at this stage, where we are running our models with the test dataset, we do not need the likelihood information. However, only the error metrics to determine the best performing regression model to the out-sample dataset. Due to the changing variety of model performance measurement metrics, we will be comparing the out-sample regression models by their MSE, MAE, and Grade Percentage results.

Upon reviewing the results presented in Table 8, we can see that Hurdle and Zero-Inflated Regression models are best performing both in regard to error metrics, where they have the lowest error rates and highest Grade Percentage Accuracy, meaning the number of predicted variables that are between +/- 2% of the actual value. The results of these two models are very similar, even almost identical to each other, that they can be considered both to be the most

successful. However, if we were to choose one of them, we would choose Hurdle Regression, due to the slightly better results.

6. Patient Classification Accuracy

As the final step of our analysis, we will be conducting a segmentation of the response variable results, which is ofp. Our aim is to review the predictive accuracy of the fitted values of each model through our confusion matrix by separating them into three segments. These segments are:

- Segment #1 = 0 - 5 physician office visits
- Segment #2 = 6 - 10 physician office visits
- Segment #3 = 11+ physician office visits

The 3x3 confusion matrices for each model using in-sample and out-sample dataset is presented below in Figure 1.

ACTUAL		PREDICTED	IN-SAMPLE			OUT-SAMPLE		
			0 - 5	6 - 10	11+	0 - 5	6 - 10	11+
POISSON	0 - 5		**57.7%**	39.3%	3.0%	**46.4%**	51.2%	2.4%
	6 - 10		30.9%	**63.0%**	6.1%	23.1%	**71.3%**	5.6%
	11+		18.6%	64.6%	**16.8%**	22.4%	67.7%	**9.9%**
POISSON w/ DISP.			0 - 5	6 - 10	11+	0 - 5	6 - 10	11+
	0 - 5		**57.7%**	39.3%	3.0%	**46.4%**	51.2%	2.4%
	6 - 10		30.9%	**63.0%**	6.1%	23.1%	**71.3%**	5.6%
	11+		18.6%	64.6%	**16.8%**	22.4%	67.7%	**9.9%**
NEG. BINOMINAL			0 - 5	6 - 10	11+	0 - 5	6 - 10	11+
	0 - 5		**60.5%**	35.6%	3.9%	**48.9%**	47.9%	3.2%
	6 - 10		35.2%	**55.9%**	8.9%	25.6%	**66.3%**	8.1%
	11+		20.4%	59.9%	**19.8%**	24.3%	63.3%	**12.4%**
HURDLE			0 - 5	6 - 10	11+	0 - 5	6 - 10	11+
	0 - 5		**55.2%**	41.4%	3.4%	**45.2%**	52.4%	2.4%
	6 - 10		28.0%	**66.1%**	5.9%	21.5%	**72.5%**	6.0%
	11+		15.3%	67.8%	**16.8%**	21.5%	67.4%	**11.0%**
ZERO-INFLATED			0 - 5	6 - 10	11+	0 - 5	6 - 10	11+
	0 - 5		**55.3%**	41.3%	3.4%	**45.1%**	52.5%	2.4%
	6 - 10		28.0%	**66.1%**	5.9%	21.5%	**72.5%**	6.0%
	11+		15.3%	67.8%	**16.8%**	21.3%	67.7%	**11.0%**

Figure 1: All Models and Datasets Confusion Matrices

Firstly, the in-sample confusion matrices of all models show us that the accuracy rate follows a similar pattern within Poisson Regression, Hurdle, and Zero-Inflated Regression Models, where segment #2 accuracy rate is the highest (ranging between 63% and 66.1%) followed by segment #1 and finally segment #3 accuracy rate. This pattern is only different for the Negative Binomial Regression Model matrix, where the highest to lower accuracy for each segment goes from #1 to #3 directly. By looking at each model's accuracy rates and matrix, we can see that Hurdle and

Zero-Inflated Model are the overall best performing models when in-sample predictive accuracy is taken into account.

Secondly, upon reviewing the out-sample confusion matrices, we see that all models have the same pattern, having the segment #2 accuracy rate as highest, segment #1 rate as the middle, and finally segment #3 rate as the lowest. While Negative Binomial Regression has the highest segment #1 accuracy rate, it also has the lowest segment #2 accuracy rate; therefore, does not show a significant advantage over other regression models presented. Overall, we can see that Poisson, Hurdle, and Zero-Inflated Regression Models have relatively close accuracy rates, differing from each other no more than +/- 2%.

If we take the average of the predictive accuracy of each in-sample model, we get 46% for Poisson, Hurdle, and Zero-Inflated Regression models, and 45% for the Negative Binomial Regression model, whereas when the average predictive accuracy of each out-sample model is calculated, we see that all models have an average true/true accuracy of 43% when all segments are taken into account.

Through all the accuracy rates presented in Figure 1, we can conclude that with none of the models, we achieve 80% accuracy that was expected of this analysis. Poisson, Hurdle, and Zero-Inflated Regression Models all show relatively similar and high accuracies yet not enough to meet our threshold.

7. Conclusion

In conclusion, we performed Poisson Regression with and without dispersion, Negative Binomial, Hurdle, and Zero-Inflated Regression Models on in and out sample dataset of Medical Care dataset. Through our analysis and the review of error, fit, and accuracy metrics results, we conclude that (a) none of the models have the accuracy of predicting the physician office visits at the required accuracy rate which is 80%, and (b) shortcomings on error and model fit related metrics aside, per the predictive accuracy results, we see that the Poisson Regression model, which is considered as the simplest model in our analysis, shows predictions as accurate as more complex models such as Hurdle and Zero-Inflated Regression Models. Although the performance of none of these models met our expectations, Poisson Regression would be the optimal model to select due to its simplicity along with its accuracy rate.

APPENDIX

Backward LM

	Dependent variable:
	ofp
	Backward LM
Constant	1.22***
	(0.12)
exclhlth	-0.43***
	(0.05)
poorhlth	0.27***
	(0.03)
numchron	0.18***
	(0.01)
adldiff	0.14***
	(0.02)
noreast	0.14***
	(0.03)
midwest	0.04
	(0.02)
west	0.20***
	(0.03)
age	-0.06***
	(0.02)
male	-0.06***
	(0.02)
school	0.02***
	(0.003)
faminc	0.01***
	(0.003)
privins	0.38***
	(0.03)
medicaid	0.30***
	(0.03)
Observations	2,203
Log Likelihood	-8,791.45
Akaike Inf. Crit.	17,610.90
Note:	*p<0.1; **p<0.05; ***p<0.01

Backward.lm Summary

Poisson Regression

	Dependent variable:
	ofp
	Poisson Regression
Constant	1.22***
	(0.12)
exclhlth	-0.43***
	(0.05)
poorhlth	0.27***
	(0.03)
numchron	0.18***
	(0.01)
adldiff	0.14***
	(0.02)
noreast	0.14***
	(0.03)
midwest	0.04
	(0.02)
west	0.20***
	(0.03)
age	-0.06***
	(0.02)
male	-0.06***
	(0.02)
school	0.02***
	(0.003)
faminc	0.01***
	(0.003)
privins	0.38***
	(0.03)
medicaid	0.30***
	(0.03)
Observations	2,203
Log Likelihood	-8,791.45
Akaike Inf. Crit.	17,610.90
Note:	*p<0.1; **p<0.05; ***p<0.01

Poisson Regression with Dispersion

	Dependent variable:
	ofp
	Poisson Regression with Dispersion
Constant	1.22***
	(0.30)
exclhlth	-0.43***
	(0.12)
poorhlth	0.27***
	(0.07)
numchron	0.18***
	(0.02)
adldiff	0.14**
	(0.06)
noreast	0.14**
	(0.06)
midwest	0.04
	(0.06)
west	0.20***
	(0.06)
age	-0.06*
	(0.04)
male	-0.06
	(0.05)
school	0.02***
	(0.01)
faminc	0.01
	(0.01)
privins	0.38***
	(0.07)
medicaid	0.30***
	(0.09)
Observations	2,203
Note:	*p<0.1; **p<0.05; ***p<0.01

Poisson Models Summary

Negative Binominal

	Dependent variable:
	ofp
	Negative Binominal
Constant	0.96***
	(0.28)
exclhlth	-0.41***
	(0.09)
poorhlth	0.31***
	(0.07)
numchron	0.21***
	(0.02)
adldiff	0.14**
	(0.06)
noreast	0.12**
	(0.06)
midwest	0.06
	(0.06)
west	0.22***
	(0.06)
age	-0.04
	(0.04)
male	-0.07
	(0.04)
school	0.02***
	(0.01)
faminc	0.01
	(0.01)
privins	0.42***
	(0.06)
medicaid	0.30***
	(0.09)
Observations	2,203
Log Likelihood	-6,043.45
theta	1.23*** (0.05)
Akaike Inf. Crit.	12,114.90
Note:	*p<0.1; **p<0.05; ***p<0.01

Hurdle Regression

	Dependent variable:
	ofp
	Hurdle Regression
Constant	1.70***
	(0.12)
exclhlth	-0.41***
	(0.05)
poorhlth	0.24***
	(0.03)
numchron	0.13***
	(0.01)
adldiff	0.15***
	(0.02)
noreast	0.11***
	(0.03)
midwest	-0.01
	(0.02)
west	0.14***
	(0.03)
age	-0.07***
	(0.02)
male	-0.01
	(0.02)
school	0.02***
	(0.003)
faminc	0.01**
	(0.003)
privins	0.23***
	(0.03)
medicaid	0.26***
	(0.04)
Observations	2,203
Log Likelihood	-7,951.32
Note:	*p<0.1; **p<0.05; ***p<0.01

Negative Binomial & Hurdle Model Summary

Zero-Inflated Regression

	Dependent variable:
	ofp
	Zero-Inflated Regression
Constant	1.70***
	(0.12)
exclhlth	-0.41***
	(0.05)
poorhlth	0.24***
	(0.03)
numchron	0.13***
	(0.01)
adldiff	0.15***
	(0.02)
noreast	0.11***
	(0.03)
midwest	-0.01
	(0.02)
west	0.14***
	(0.03)
age	-0.07***
	(0.02)
male	-0.01
	(0.02)
school	0.02***
	(0.003)
faminc	0.01**
	(0.003)
privins	0.23***
	(0.03)
medicaid	0.26***
	(0.04)
Observations	2,203
Log Likelihood	-7,951.32
Note:	*p<0.1; **p<0.05; ***p<0.01

Zero_inflated Regression Summary

```
# Assignment_9
# Serra Uzun
# 11.22.2020

getwd()
setwd("/Users/serrauzun/Desktop/MSDS_410_Supervised/Assignment #9")
medical.df <- read.delim("medical_care.txt",header = FALSE, sep = "", dec = ".")

## ADD Column Names
colnames(medical.df) <-
c("ofp","ofnp","opp","opnp","emr","hosp","exclhlth","poorhlth","numchron","adldiff","noreast",
"midwest","west","age","black","male","married","school","faminc","employed","privins","medicaid
")
colnames(medical.df)
str(medical.df)
table(medical.df$ofp)

#drop ofnp, opp, opnp, emr,and hosp
colnames(medical.df)
dim(medical.df)
medical.df_clean <- medical.df[,-c(2:6)]
dim(medical.df_clean)
## SPLIT Data
set.seed(789)
medical.df_clean$u <- runif(n=dim(medical.df_clean)[1],min=0,max=1)
sample <- sample.split(medical.df_clean$u, SplitRatio = .50)
train.df <- subset(medical.df_clean, sample == TRUE)
test.df  <- subset(medical.df_clean, sample == FALSE)

# Check your data split. The sum of the parts should equal the whole. # Do your totals add up?
dim(medical.df_clean)[1]
dim(train.df)[1]
dim(test.df)[1]
dim(train.df)[1] + dim(test.df)[1]

dim(medical.df_clean)
colnames(medical.df_clean)
colnames(train.df)

train.df <- train.df[,-18]
test.df <- test.df[,-18]

upper.glm <- glm(ofp ~ ., data=train.df,family=c('poisson'))
lower.glm <- glm(ofp ~ 1, data=train.df,family=c('poisson'))

#backward lm
backward.lm <- stepAIC(object=upper.glm,direction=c('backward'))
```

```
summary(backward.lm)
out.path = "/Users/serrauzun/Desktop/MSDS_410_Supervised/Assignment #9"
bckwrd_lm <- 'Backward.html';
stargazer(backward.lm, type=c('html'),out=paste(out.path,bckwrd_lm,sep=''),
          title=c('Backward LM'),
          align=TRUE, digits=2, digits.extra=2, initial.zero=TRUE,
          column.labels=c('Backward LM'), intercept.bottom=FALSE)

##############################################
##############################################
##############################################
##############################################
##############################################
##############################################
gof.ins <- function(model,response){

  f.aic <- AIC(model);
  f.bic <- BIC(model);

  f.mse <- mean((response - model$fitted.values)^2);
  f.mae <- mean(abs(response - model$fitted.values));
  f.mape <- mean(abs(response - model$fitted.values)/response);

  abs.residual <- abs(response - model$fitted.values);
  grade.2 <- mean(ifelse(abs.residual<=2,1,0));

  output <- list(AIC=f.aic,BIC=f.bic,MSE=f.mse,MAE=f.mae,MAPE=f.mape,GRADE=grade.2);
  return(output)
}

gof.oos <- function(model,response){

  f.mse <- mean((response - model$fitted.values)^2);
  f.mae <- mean(abs(response - model$fitted.values));
  f.mape <- mean(abs(response - model$fitted.values)/response);

  abs.residual <- abs(response - model$fitted.values);
  grade.2 <- mean(ifelse(abs.residual<=2,1,0));

  output <- list(MSE=f.mse,MAE=f.mae,MAPE=f.mape,GRADE=grade.2);
  return(output)
}

##############################################
##############################################
##############################################
##############################################
```

```
##############################################
##############################################
#POISSON REGRESSION
model.poisson <- glm(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west + age
+ male + school + faminc + privins + medicaid,family="poisson", data=train.df)
summary(model.poisson)

poisson <- 'Poisson.html';
stargazer(model.poisson, type=c('html'),out=paste(out.path,poisson,sep=''),
        title=c('Poisson Regression'),
        align=TRUE, digits=2, digits.extra=2, initial.zero=TRUE,
        column.labels=c('Poisson Regression'), intercept.bottom=FALSE)

#POISSON REGRESSION WITH DISPERSION
model.poisson_disp <- glm(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west
+ age + male + school + faminc + privins + medicaid,family="quasipoisson", data=train.df)
summary(model.poisson_disp)
help("abs")
poisson_disp <- 'Poisson_disp.html';
stargazer(model.poisson_disp, type=c('html'),out=paste(out.path,poisson_disp,sep=''),
        title=c('Poisson Regression with Dispersion'),
        align=TRUE, digits=2, digits.extra=2, initial.zero=TRUE,
        column.labels=c('Poisson Regression with Dispersion'), intercept.bottom=FALSE)

##############################################
#NEGATIVE BINOMIAL
help(glm.nb)
neg.binom <- glm.nb(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west + age
+ male + school + faminc + privins + medicaid, data=train.df)
summary(neg.binom)

neg_binom <- 'Neg Binom.html';
stargazer(neg.binom, type=c('html'),out=paste(out.path,neg_binom,sep=''),
        title=c('Negative Binomial'),
        align=TRUE, digits=2, digits.extra=2, initial.zero=TRUE,
        column.labels=c('Negative Binomial'), intercept.bottom=FALSE)

##############################################
#HURDLE
model.hurdle <- hurdle(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west +
age + male + school + faminc + privins + medicaid, data=train.df)
summary(model.hurdle)

hurdle_file <- 'Hurdle Regression.html';
stargazer(model.hurdle, type=c('html'),out=paste(out.path,hurdle_file,sep=''),
        title=c('Hurdle Regression'),
        align=TRUE, digits=2, digits.extra=2, initial.zero=TRUE,
```

```
        column.labels=c('Hurdle Regression'), intercept.bottom=FALSE)

##########################################
#ZERO-INFLATED
model.zeroinfl <- zeroinfl(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west +
age + male + school + faminc + privins + medicaid, data=train.df)
summary(model.zeroinfl)

zeroinfl_file <- 'Zero-Inflated Regression.html';
stargazer(model.hurdle, type=c('html'),out=paste(out.path,zeroinfl_file,sep=''),
        title=c('Zero-Inflated Regression'),
        align=TRUE, digits=2, digits.extra=2, initial.zero=TRUE,
        column.labels=c('Zero-Inflated Regression'), intercept.bottom=FALSE)

### Results
gof.ins(model.poisson,train.df$ofp)
gof.ins(model.poisson_disp,train.df$ofp)
gof.ins(neg.binom,train.df$ofp)
gof.ins(model.hurdle,train.df$ofp)
gof.ins(model.zeroinfl,train.df$ofp)

############################################################
############################################################
############################################################
#POISSON REGRESSION TEST
model.poisson_test <- glm(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west
+ age + male + school + faminc + privins + medicaid,family="poisson", data=test.df)
summary(model.poisson_test)

##########################################
#POISSON REGRESSION WITH DISPERSION TEST
model.poisson_disp_test <- glm(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest +
west + age + male + school + faminc + privins + medicaid,family="poisson", data=test.df)
summary(model.poisson_disp_test)

##########################################
#NEGATIVE BINOMIAL TEST
neg.binom_test <- glm.nb(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west +
age + male + school + faminc + privins + medicaid, data=test.df)
summary(neg.binom_test)

#HURDLE TEST
help(hurdle)
model.hurdle_test <- hurdle(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest + west
+ age + male + school + faminc + privins + medicaid, data=test.df)
summary(model.hurdle_test)
```

```
##############################################
#ZERO-INFLATED TEST
help(zeroinfl)
model.zeroinfl_test <- zeroinfl(ofp ~ exclhlth + poorhlth + numchron + adldiff + noreast + midwest +
west + age + male + school + faminc + privins + medicaid, data=test.df)
summary(model.zeroinfl_test)

### Results
gof.oos(model.poisson_test,test.df$ofp)
gof.oos(model.poisson_disp_test,test.df$ofp)
gof.oos(neg.binom_test,test.df$ofp)
gof.oos(model.hurdle_test,test.df$ofp)
gof.oos(model.zeroinfl_test,test.df$ofp)

# Classification - Function
conf_matrix <- function(actual, predict){
  actual_segment <- ifelse(round(actual,0)<=5,'0-5',
                           ifelse(round(actual,0)>5 & round(actual,0)<=10,'6-10',
                                  ifelse(round(actual,0)>10,'11+',
                                         'Other')))
  predict_segment <- ifelse(round(predict,0)<=5,'0-5',
                            ifelse(round(predict,0)>5 & round(predict,0)<=10,'6-10',
                                   ifelse(round(predict)>10,'11+',
                                          'Other')))
  t <- table(actual_segment, predict_segment)
  r <- apply(t,MARGIN=1,FUN=sum)
  return (t/r)
}

# Classification - In-Sample
conf_matrix(train.df$ofp, model.poisson$fitted.values)
conf_matrix(train.df$ofp, model.poisson_disp$fitted.values)
conf_matrix(train.df$ofp, neg.binom$fitted.values)
conf_matrix(train.df$ofp, model.hurdle$fitted.values)
conf_matrix(train.df$ofp, model.zeroinfl$fitted.values)

# Classification - Out-Sample
conf_matrix(test.df$ofp, model.poisson_test$fitted.values)
conf_matrix(test.df$ofp, model.poisson_disp_test$fitted.values)
conf_matrix(test.df$ofp, neg.binom_test$fitted.values)
conf_matrix(test.df$ofp, model.hurdle_test$fitted.values)
conf_matrix(test.df$ofp, model.zeroinfl_test$fitted.values)
```