Regression Modeling Using Airbnb Data of Melbourne

Xixi Chen; Lixian Chen; George Liu; Yuhao Wang
Deparment of Statistics, Columbia University

Abstract

We analyze a dataset drawn from AirBnB listings in Melbourne, Australia. In order to understand the drivers of conversion and engagement as measured by review count and predict the number of reviews that a listing is able to generate, we attempt to fit a loglinear model (Poisson GLM) as well as variations in order to deal with overdispersion and zero inflation in the data given a set of predictors.

In particular, because we are not privy to data about bookings given our data is scraped from the AirBnB site, we leverage factors that are likely to be predictors of historic conversion such as price, Instant Book, and availability. We also consider other variables such as months listed and property type that may have effects on reviews independent of conversion.

Introduction

Generalized linear models (GLM) play a critical role in categorical data analysis. The Poisson regression model is one special GLM designed to deal with the count data. In this project, we are particularly interested in predicting the number of reviews for each AirBnB host in Melbourne, which can be regarded as a kind of count data and thus may be applied with the Poisson regression model. We will mainly focus on three kinds of models: Poisson regression model, negative binomial regression model and zero inflated Poisson regression model.

Generalized linear model

As the name suggests, GLM is a generalization of the traditional linear regression model. We can also say the classic linear regression model is a special case of the GLM when the random component is the normal distribution. More formally, we define the GLM as follows, which consists of 3 parts:

1. The distribution of the response is from the exponential distribution family, which means it has the form:

$$f(y_i|\theta_i) = a(y_i)b(\theta_i)exp(y_i\theta_i)$$

2. The systematic component:

$$\eta_i = \Sigma_{i=1}^{p}\beta_i x_i$$

3. A link function $g(x)$ which links the systematic component and the mean of the response, such as identity link and canonical link:

$$g(\mu_i) = \eta_i$$

Poisson regression model

As we have mentioned before, Poisson regresssion model is a special GLM. More specifically, it assumes the response has a Poisson distribution and the link function is the canonical link. Mathematically, we write the pdf of the response as:

$$f(y_i|\mu_i) = e^{-\mu_i}\frac{\mu_i^{y_i}}{y_i!}$$

and the canonical link has the form :

$$\eta_i = \log(\mu_i)$$

And this is our Poisson regression model or log-linear model.

Poisson regression model for rates

It is sometimes more natural for us to fit a Poisson regression model for rates than counts when the response is counted over different periods. We thus begin by specifying a rate model:

$$\log(\mu_i/t_i) = \alpha + \beta * x_i$$

or

$$\log(\mu_i) = \alpha + \beta * x_i + \log(t_i)$$

in which we call $\log(t_i)$ as offset.

Quasi-Poisson regression model

Usually, Poisson regression model is simple and thus it has several drawbacks. One deficiency of Poisson regression model is that it can't deal with a phenomena called overdispersion. An overdispersion occurs when the variance of true response appears to be larger than that of the predicted response. Thus, it is natural for people to come up with a more complicated model to tackle this issue. For example, we can use the quasi-poisson regression model by incorparating a dispersion parameter.

To introduce the quasi-likelihood regression, we will briefly talk about the likelihood equation of the glm first. And more generally, we assume that our response is from the exponential dispersion family, which has the form:

$$f(y_i|\theta_i, \phi) = exp(\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i, \phi))$$

.

Then, the log-likelihood function is:

$$l = \Sigma_{i=1}^n \frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + \Sigma_{i=1}^n c(y_i, \phi)$$

By taking derivative w.r.t β, the likelihood equation is:

$$\Sigma_{i=1}^{n} \frac{(y_i - \mu_i)x_{ij}}{var(Y_i)} \frac{\partial \mu_i}{\partial \eta_i} = 0, j = 1, ..., p$$

.

Moreover, the likelihood equation for the Poisson regression is:

$$\Sigma_{i=1}^{n} \frac{(y_i - \mu_i)x_{ij}}{\mu_i} \frac{\partial \mu_i}{\partial \eta_i} = 0, j = 1, ..., p$$

.

For the quasi-Poisson regression, we change the above equation slightly by introducing the parameter ϕ:

$$\Sigma_{i=1}^{n} \frac{(y_i - \mu_i)x_{ij}}{\phi\mu_i} \frac{\partial \mu_i}{\partial \eta_i} = 0, j = 1, ..., p$$

.

This is so-called quasi-Possion regression model.

Negative binomial regression model

Another option to deal with overdispersion is the negative binomial regression model. That's to say, we will assume the response has a negative binomial distribution. More specifically, if we assume the parameter μ for the Poisson distribution has a gamma distribution $\Gamma(r, \frac{1-p}{p})$, we can show that the posterior distribution is exactly a negative binomial distribution with parameter r and p: $Y \sim NegBinom(r, p)$.

Zero inflated Poisson regression model

In real life, not all count data behaves exactly like a Poisson distribution. More often than not, it is common to see 0 over represented in the data which is called zero inflation, like what we encountered in our project. To take this into consideration, the zero inflated poisson regression or ZIP model is proposed. The idea is that we slightly change the response distribution by including an additional parameter π_i called probability of extra zeros:

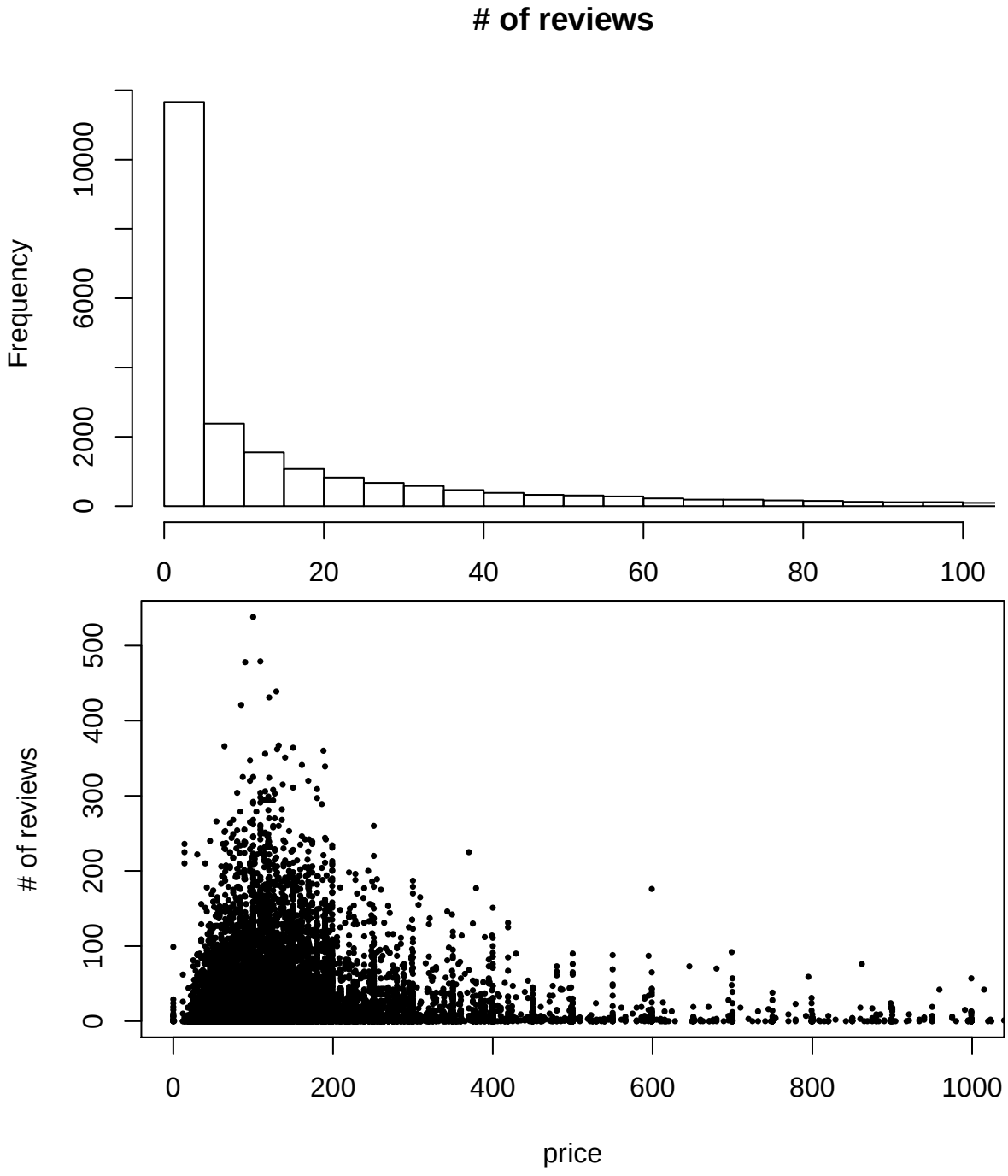
$$\mathbb{P}(y_i = 0) = \pi_i + (1 - \pi_i)e^{-\mu_i}$$

$$\mathbb{P}(y_i = k) = (1 - \pi_i)e^{-\mu_i} \frac{\mu_i^k}{k!}$$

Exploratory data analysis

We quickly explore our response variable by drawing a histogram and a scatter plot against the price and find the empirical distribution of the response variable is not exactly in a form of Poisson distribution. For example, there is more 0 counts in the histogram. Luckliy, the zero inflation Poisson regression model we mentioned above might be used to deal with this issue hopefully. Also, we notice that there is an apparent relationship between the response variable and the price variable by observing the scatter plot. Therefore, we start our basic model by regressing on the price below.

```
## Joining, by = "id"
```



Model building and inference

Poisson regression model

We first try to fit the simplest model, the Poisson regression model using number of reviews as the response and price as the predictor.

```
##
## Call:
```

```
## glm(formula = number_of_reviews ~ price + offset(log(months)),
##     family = poisson, data = listings_joined)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -14.716   -5.217   -1.382    2.335   37.514
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  5.564e-02  2.170e-03   25.65   <2e-16 ***
## price       -8.960e-04  1.203e-05  -74.48   <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: 938271  on 22894  degrees of freedom
## Residual deviance: 931088  on 22893  degrees of freedom
## AIC: 1005455
##
## Number of Fisher Scoring iterations: 6
```

The result above shows that the model coefficient is significant and thus suggests that the price variable should be included. The model can be written as:

$$\log(\mu) = 0.39 - 4.7 * 10^{-4} * x$$

.

It suggests there is a negative relationship between number of reviews and the price, which is legitimate since the more expensive, the less the guests and thus the less the number of reviews. We then plot the residuals against the fitted value and conduct the goodness of fit test.

We then compare the model with the saturated model using analysis with deviance. It also suggests that the p-value is small enough for us to accept the alternative that the number of reviews drops as the price increses.

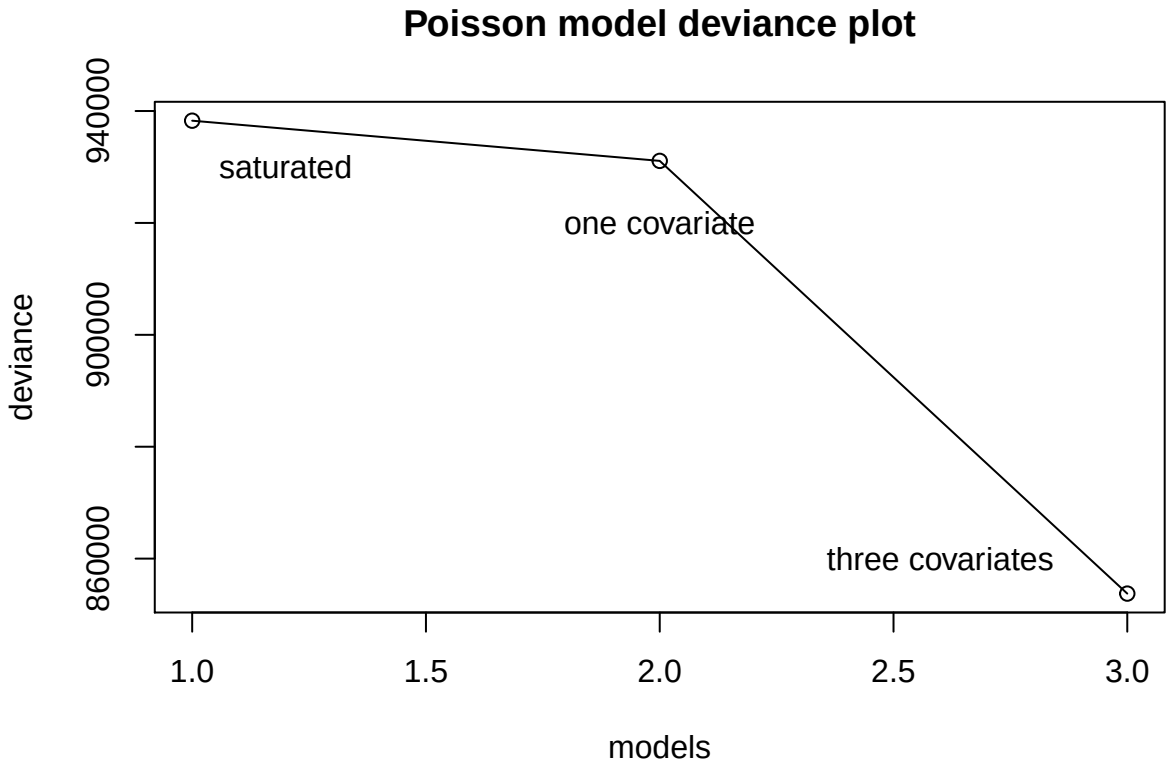
```
## Analysis of Deviance Table
##
## Model 1: number_of_reviews ~ 1 + offset(log(months))
## Model 2: number_of_reviews ~ price + offset(log(months))
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1     22894     938271
## 2     22893     931088  1   7182.2 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

More covariates

Next, we try to add two more covariate into the above and perform the same pocedure, the availability_365 and instant_bookable. We also try to compare models with the saturated model using deviance. The results shows the two added covariates is also significant. The deviance plot comparing the three model is more straight forward.

```
## Analysis of Deviance Table
##
## Model 1: number_of_reviews ~ 1 + offset(log(months))
```

```
## Model 2: number_of_reviews ~ price + availability_365 + instant_bookable +
##     offset(log(months))
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1     22894     938271
## 2     22891     853748  3    84523 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



Even though the models above are significant to some extent. We know that Poisson regression model is not perfect. Especially, it has a common problem called overdispersion. We then try to check the issue by comparing the response variance and the predicted value variance.

```
## [1] "true variance:"
## [1] 1522.061
## [1] "predicted variance 1:"
## [1] 297.7534
## [1] "predicted variance 2:"
## [1] 388.681
```

It is obvious that, the original data has a severe problem of overdispersion which suggests us other methods to predict the response. We then turn to fit the negative binomial model.

Fitting negative binomial regression

We have already selected three significant variable above and we will use them as the assumed predictors for our negative binomial model here. The model output is shown below. We can see that the results are significant. We then calculte the predicted respose variance of the model and find it increases comparing to those Poisson models, which is 549.05.

```
##
## Call:
## glm.nb(formula = number_of_reviews ~ price + availability_365 +
##     instant_bookable + +offset(log(months)), data = listings_joined,
##     init.theta = 0.3941475277, link = log)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2371  -1.1736  -0.4181   0.2448   4.1934
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -1.833e-01  2.026e-02  -9.046   <2e-16 ***
## price             -8.443e-04  6.997e-05 -12.067   <2e-16 ***
## availability_365   1.317e-03  8.131e-05  16.194   <2e-16 ***
## instant_bookablet  5.653e-01  2.157e-02  26.211   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for Negative Binomial(0.3941) family taken to be 1)
##
##     Null deviance: 27189  on 22894  degrees of freedom
## Residual deviance: 26230  on 22891  degrees of freedom
## AIC: 165836
##
## Number of Fisher Scoring iterations: 1
##
##
##               Theta:  0.39415
##           Std. Err.:  0.00374
##
##  2 x log-likelihood:  -165826.33700

## [1] "true variance:"

## [1] 1522.061

## [1] "predicted variance:"

## [1] 549.0513
```

Fitting ZIP model

As we mentioned before, when observing the response histogram plot, we notice that there is an interesting phenomenon: the number of 0 counts is far more than others. We now try to fit the so called zero inflation Poisson model. Similarly, we find those predictors are also significant in the zero inflation Poisson model according to the output.

```
##
## Call:
## zeroinfl(formula = number_of_reviews ~ price + availability_365 +
##     instant_bookable + offset(log(months)), data = listings_joined)
##
## Pearson residuals:
##       Min        1Q    Median        3Q       Max
##   -2.8308   -1.3625   -0.8207    0.8899 1190.7501
```

```
##
## Count model coefficients (poisson with log link):
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -2.800e-02  2.879e-03  -9.725   <2e-16 ***
## price             -6.694e-04  6.990e-06 -95.763   <2e-16 ***
## availability_365   9.651e-04  1.029e-05  93.796   <2e-16 ***
## instant_bookablet  6.276e-01  2.877e-03 218.109   <2e-16 ***
##
## Zero-inflation model coefficients (binomial with logit link):
##                      Estimate Std. Error  z value Pr(>|z|)
## (Intercept)       -4.3180279  0.0309359 -139.580  < 2e-16 ***
## price              0.0007930  0.0001008    7.866 3.67e-15 ***
## availability_365  -0.0018317  0.0001325  -13.828  < 2e-16 ***
## instant_bookablet  0.2752776  0.0349757    7.871 3.53e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Number of iterations in BFGS optimization: 17
## Log-likelihood: -2.942e+05 on 8 Df
```

Also, in the zero inflation model function, by changing the link to negative binomial link, it will give us the zero inflation negative binomial model. The output is shown below.

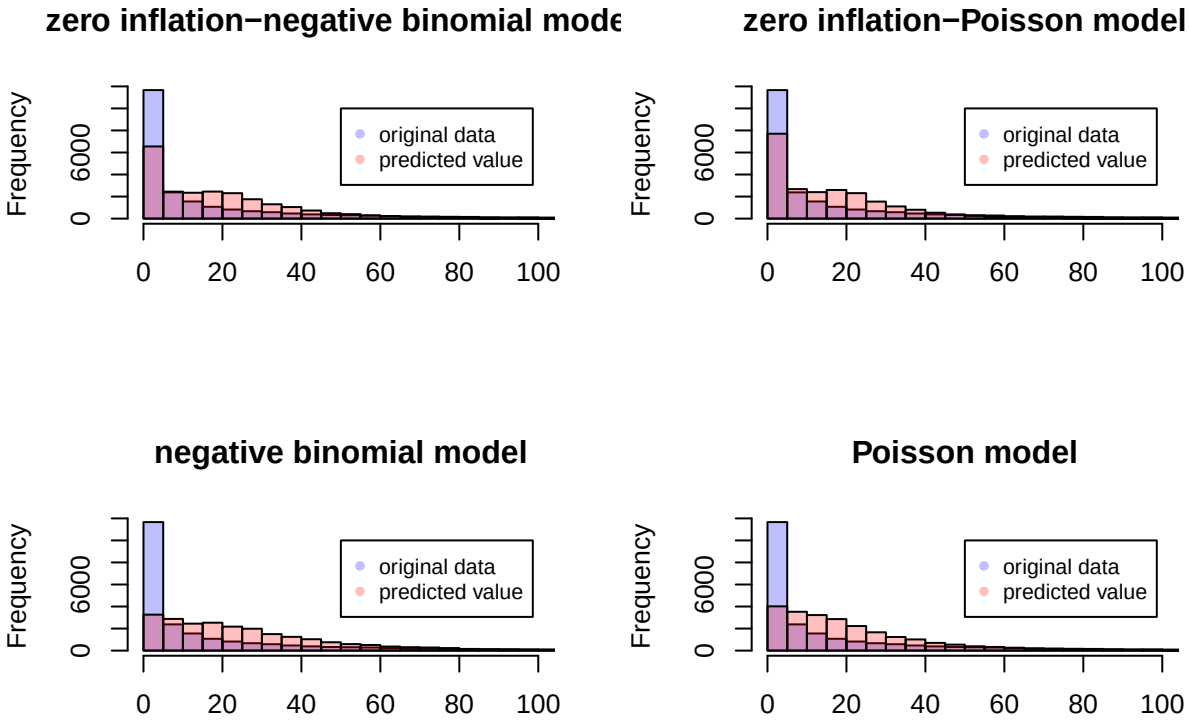
```
##
## Call:
## zeroinfl(formula = number_of_reviews ~ price + availability_365 +
##     instant_bookable + offset(log(months)), data = listings_joined,
##     dist = "negbin")
##
## Pearson residuals:
##      Min       1Q   Median       3Q      Max
##  -0.8221  -0.6355  -0.4021   0.3047 440.1621
##
## Count model coefficients (negbin with log link):
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)        3.993e-02  1.846e-02   2.163   0.0306 *
## price             -5.161e-04  7.039e-05  -7.333 2.25e-13 ***
## availability_365   8.291e-04  7.478e-05  11.087  < 2e-16 ***
## instant_bookablet  5.285e-01  1.765e-02  29.943  < 2e-16 ***
## Log(theta)        -2.863e-01  1.226e-02 -23.358  < 2e-16 ***
##
## Zero-inflation model coefficients (binomial with logit link):
##                      Estimate Std. Error  z value Pr(>|z|)
## (Intercept)       -4.5297124  0.0372372 -121.645  < 2e-16 ***
## price              0.0008192  0.0001209    6.776 1.23e-11 ***
## availability_365  -0.0026005  0.0001724  -15.086  < 2e-16 ***
## instant_bookablet  0.1995159  0.0426965    4.673 2.97e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Theta = 0.751
## Number of iterations in BFGS optimization: 15
## Log-likelihood: -8.057e+04 on 9 Df
```

Model comparison

We compare models mainly from three perspectives: predicted value, residual and Vuong test.

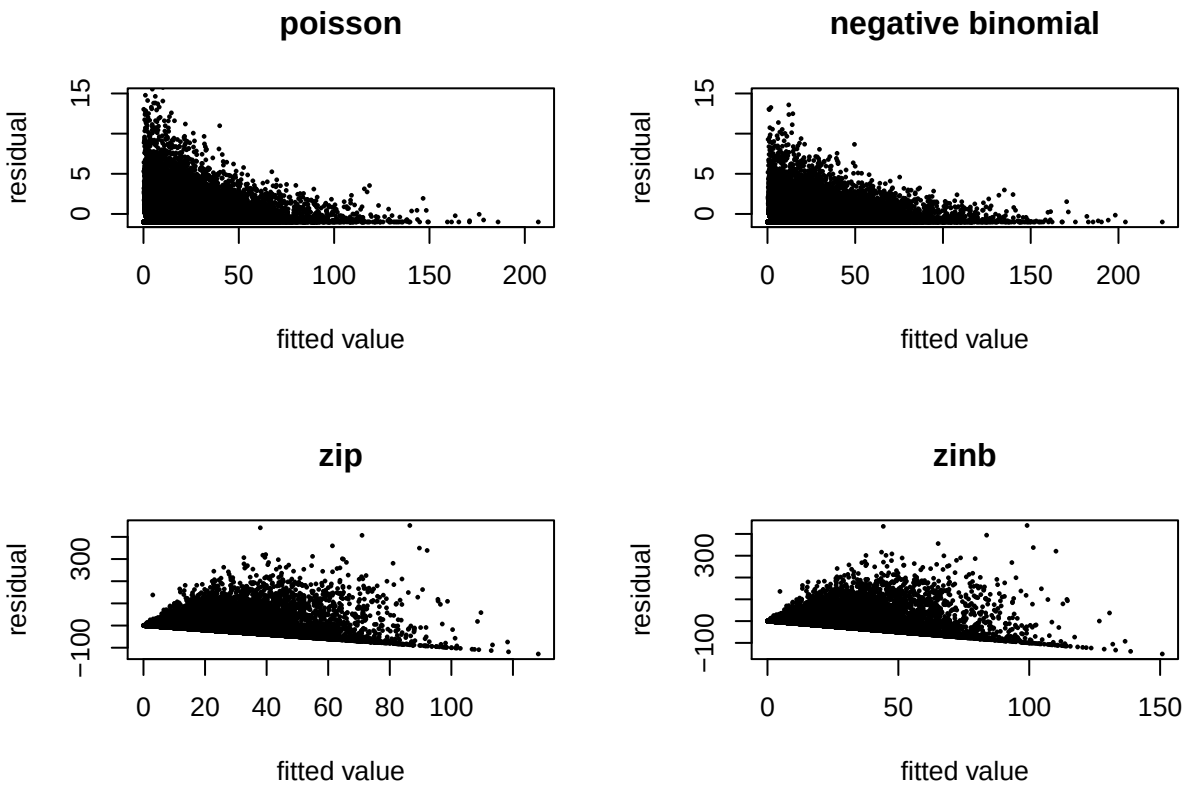
Predicted value comparison

According to the following predicted value plots of the models, we find both zero inflation models and zero inflation negative binomial models deal with the zero inflation issue relatively better.



Residual comparison

We then plot the residual plots and find for both Poisson regression model and negative binomial model have a problem of heteroscidaticity. But for zero inflated Poisson and zero inflated negarive binomail model, this issue seems less severe.



Vuong test

Vuong test is a likelihood based test for model comparison using the KL divergence criterion. It is suitable for not just nested models, but also non-nested models. According to the output below, we conclude that: zero inflation binomial model > negative binomial model > zero inflation Poisson model > Poisson model. It is also explainable. Because for the zero inflated negative binomial model, it takes both zero inflation and overdispersion into account and thus it is undoubtedly the best. While for zero inflated Poisson and negative binomial model, both models only deal with one part of the issues and thus it is not easy to predict which one is better beforehand. All Vuong test results are summarized in the table below.

Models Compared	Vuong z-statistic
Poisson and Negative Binomial	79.34
Poisson and Zero-inflated Poisson	79.74
Zero-inflated Poisson and Negative Binomial	62.27
Zero-inflated Negative Binomial and Negative Binomial	36.45
Zero-inflated Poisson and Zero-inflated Negative Binomial	63.45

Conclusion

We tested the Poisson, negative binomial Poisson, quasi-Poisson, zero-inflated Poisson, zero-inflated negative binomial models. We also compared Poisson models with 0 predictors, 1 predictor, and 3 predictors. For the quasi Poisson, due to irregular outputs with an extremely high predicted ϕ value, we decided to not utilize this model.

We were able to produce a model which we felt had reasonable predictive power, in which we see a negative

relationship between price and reviews, but a positive one in availability and Instant Book status. None of the coefficients provides a shocking conclusion, so this model is one that we would feel comfortable with.

However as we can see from the histogram there are still areas which can be refined. For instance, we can see that our model provides higher weights in the mid-range of review count. We also observe that there are certain residuals that are quite large.

The natural next steps are to understand the points with large residuals to see whether they have excessive leverage, and to refine our understanding of the number of low-review count and whether we could introduce additional variables to improve the fit.

Reference

- Tyler Xie. (2019). Melbourne AirBnB Open Data. (Version 10) [Data files]. Retrieved from https://www.kaggle.com/tylerx/melbourne-airbnb-open-data/
- Vuong, Quang H. (1989). "Likelihood Ratio Tests for Model Selection and non-nested Hypotheses". Econometrica. 57 (2): 307–333. JSTOR 1912557.
- Agresti, Alan. (2012) Categorical Data Analysis (3rd ed.). Hoboken, NJ: John Wiley and Sons