Bike Sharing Data Analysis

Leigh Preimesberger, Jackson Maines, Caitlyn Nguyen, Joshua Jung

Along for the Ride

Table of Contents

Data Description

Introduction

Our bike sharing dataset is from the UC Irvine Machine Learning Repository. The dataset has 17389 observations with 16 variables. In the bike sharing dataset, bike sharing counts have been aggregated on an hourly basis and on a daily basis. We used the daily count of rented bikes between the years 2011 and 2012. Limiting our focus to the daily count of rented bikes reduces the number of observations to 731 and the number of variables to 13 (since the variables: **hr, registered,** and, **cnt** are no longer used).

Analysis Goal

The main goal of the project is to find the best predictor(s) for modeling the daily count of rented bikes. It is reasonable to think that environmental and seasonal settings could affect bike rental behaviors. For example, the daily count of rented bikes could depend on weather conditions, day of the week, and season. Hence, we fit a linear regression model, using the daily count of rented bikes as our response variable and the remaining variables as our predictors. The daily count of rented bikes has been measured in three different ways: the daily count of casual users who rented a bike (**casual**), the daily count of registered users who rented a bike (**registered**), and the total count of users who rented a bike (**cnt**). We decided to limit our focus to the total count of users who rented a bike and fit a regression model accordingly so.

Variable Attributes

Response Variables

- **casual**: Count of casual users
- **registered**: Count of registered users
- **cnt**: Count of total rental bikes including both casual and registered

Regressor Variables

- **season**: Season (1:winter, 2:spring, 3:summer, 4:fall)
- **yr**: year (0:2011, 1:2012)
- **holiday**: Whether day is holiday or not
- **weekday**: Day of the week
- **workingday:**
    - 1: If day is neither weekend or holiday
    - 0: Otherwise
- **weathersit**:
    - 1: Clear
    - 2: Mist

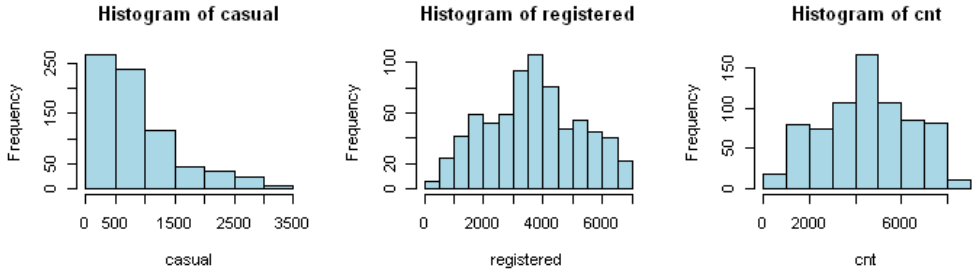
    - 3: Light Rain
    - 4: Heavy Rain
- **temp** : Normalized temperature in Celsius
- **atemp**: Normalized feeling temperature in Celsius
- **hum**: Normalized humidity
- **windspeed**: Normalized wind speed

Data Analysis

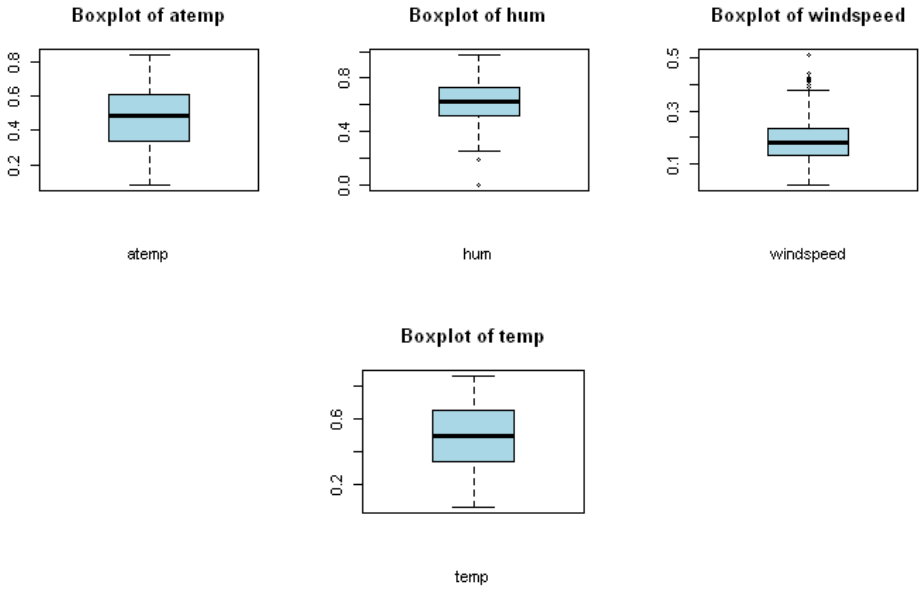
Data Cleaning

When importing the dataset, factored variables: **mnth, season, weekday,** and **weathersit** were imported as numerical values instead of leveled values and were promptly converted into factored levels. Upon initial inspection of the dataset we noticed potential data entry errors that seemed either impossible or were noticeable as an extreme outlier of the data. We decided to remove point 69 from our dataset as a humidity level of 0 was not a possible value. Additionally, point 668 was removed as a bike count of 22 fell widely outside our range of values for bike count.

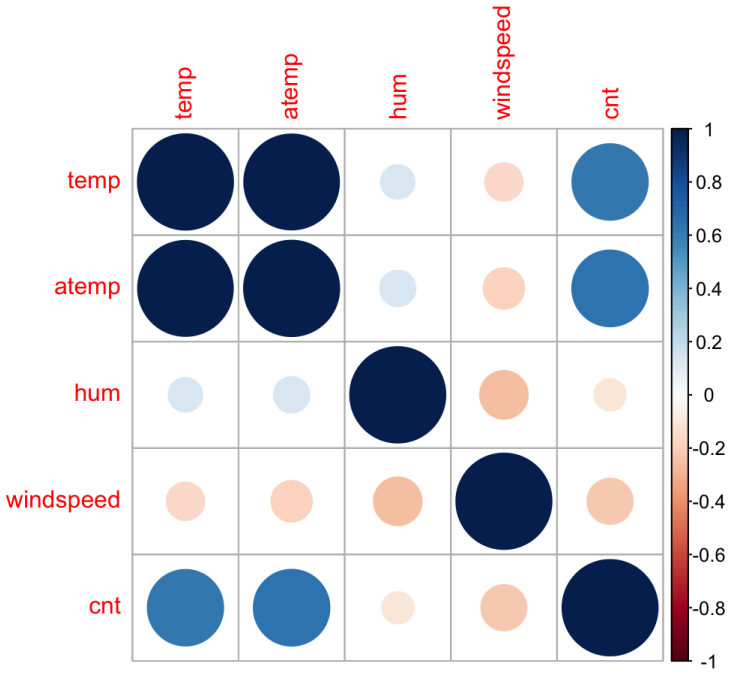
Exploratory Analysis



As shown from our histograms of our potential response variables, **casual** displayed a skewed right response while **registered** and **cnt** displayed a normal distribution. From this, we decided to move forward with limiting our focus of analysis on **cnt.**



Later, we created boxplots of our quantitative predictors to check for distribution and verify our normality assumption of the regressor variables. We see that the boxplot is normally and symmetrically distributed for all of the quantitative predictors: **atemp, hum, windspeed,** and, **temp.**



As shown in our heat map, we were able to see how our quantitative predictors correlated with one another as well as with our response variable of choice. We see that **temp** and **atemp** were highly correlated with one another indicating that one might need to be removed from the

model later on. However, out of all the quantitative regressor variables, **temp** and **atemp** also had the highest correlation with our chosen response variable **cnt** while **hum** had the lowest correlation with the response variable**.** We concluded that the high correlation is indicative of potential linear correlation and high predictive influence.

Model Fitting

Due to our response variable's attribute of being discrete counts with large variance and being normally distributed, implying a negative binomial distribution, it was decided it was appropriate to proceed to fit our data with negative binomial regression models. A negative binomial model would take into account the discrete nature of our data as well as be able to handle the wide range of values and variance of our counts.

Variable Selection

We first fit a model with all of our regressors versus our count variable.

```
Call:
glm.nb(formula = cnt ~ . - registered - casual - dteday - instant,
    data = day, init.theta = 18.11773957, link = log)

Coefficients: (1 not defined because of singularities)
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  7.31949    0.07346  99.637  < 2e-16 ***
season2      0.26746    0.05497   4.866 1.14e-06 ***
season3      0.30452    0.06525   4.667 3.05e-06 ***
season4      0.51656    0.05546   9.314  < 2e-16 ***
yr           0.47873    0.01783  26.856  < 2e-16 ***
mnth2        0.12179    0.04410   2.762 0.005750 **
mnth3        0.20426    0.05071   4.028 5.63e-05 ***
mnth4        0.14960    0.07583   1.973 0.048514 *
mnth5        0.18976    0.08196   2.315 0.020603 *
mnth6        0.06655    0.08648   0.770 0.441593
mnth7       -0.08846    0.09609  -0.921 0.357270
mnth8       -0.01045    0.09282  -0.113 0.910391
mnth9        0.14778    0.08119   1.820 0.068729 .
mnth10       0.07058    0.07401   0.954 0.340318
mnth11       0.03606    0.07072   0.510 0.610072
mnth12       0.02309    0.05589   0.413 0.679462
holiday     -0.19742    0.05525  -3.573 0.000353 ***
weekday1     0.06667    0.03353   1.989 0.046747 *
weekday2     0.08217    0.03280   2.505 0.012250 *
weekday3     0.07444    0.03291   2.262 0.023697 *
weekday4     0.09039    0.03292   2.745 0.006042 **
weekday5     0.11775    0.03289   3.580 0.000344 ***
weekday6     0.08698    0.03263   2.666 0.007686 **
workingday         NA         NA      NA       NA
weathersit2 -0.09603    0.02361  -4.068 4.74e-05 ***
weathersit3 -0.71079    0.06048 -11.753  < 2e-16 ***
temp         0.85161    0.42782   1.991 0.046526 *
atemp        0.74713    0.44741   1.670 0.094940 .
hum         -0.39509    0.08958  -4.411 1.03e-05 ***
windspeed   -0.72060    0.12694  -5.677 1.37e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(18.1177) family taken to be 1)

    Null deviance: 3397.8  on 730  degrees of freedom
Residual deviance:  747.8  on 702  degrees of freedom
AIC: 12155

Number of Fisher Scoring iterations: 1


              Theta:  18.118
          Std. Err.:  0.956

 2 x log-likelihood:  -12095.229
```

From this output we can see that there are regressors that are not significant. **mnth 6** through **mnth 12**, **workingday**, and **atemp** all have p-values greater than .05. We had the goal to create a model that was as realistic as possible while still being able to accurately predict rented bikes. Removing half of our months from the model would lower the AIC but this wouldn't be realistic. Instead we removed months in favor of **seasons**, as the p-values of all of the seasons were less than .05. Regarding the **workingday** regressor, we could not figure out why we were getting NA's in our model. It was concluded that we would likely remove this regressor regardless since we had the **holiday** and **weekday** regressors in our reduced model. **atemp** was removed due to having a p-value greater than .05, and from our VIF we noticed that **temp** and **atemp** had an issue with multicollinearity.

	GVIF	Df	GVIF^(1/(2*Df))
season	169.711671	3	2.352982
yr	1.053005	1	1.026160
mnth	408.953062	11	1.314354
holiday	1.121609	1	1.059061
weekday	1.163157	6	1.012674
weathersit	2.025399	2	1.192965
temp	80.790634	1	8.988361
atemp	70.113658	1	8.373390
hum	2.294761	1	1.514847
windspeed	1.281761	1	1.132149

Name	Model	AIC (weights)	AICc (weights)	BIC (weights)	Nagelkerke's R2	RMSE	Sigma	Score_log	Score_spherical
model_cnt3	negbin	12232.9 (<.001)	12234.9 (<.001)	12352.3 (<.001)	0.971	907.992	1.000	-8.923	0.031
model_cnt4	negbin	12193.2 (0.996)	12194.1 (0.995)	12275.9 (0.958)	0.975	929.088	1.000	-8.915	0.031
model_cnt7	negbin	12204.0 (0.004)	12204.8 (0.005)	12282.1 (0.042)	0.973	935.341	1.000	-8.919	0.031

We explored removing additional regressors and decided that we preferred model_cnt4. For the reason that model_cnt4 had the lowest AIC and no issues with multicollinearity.

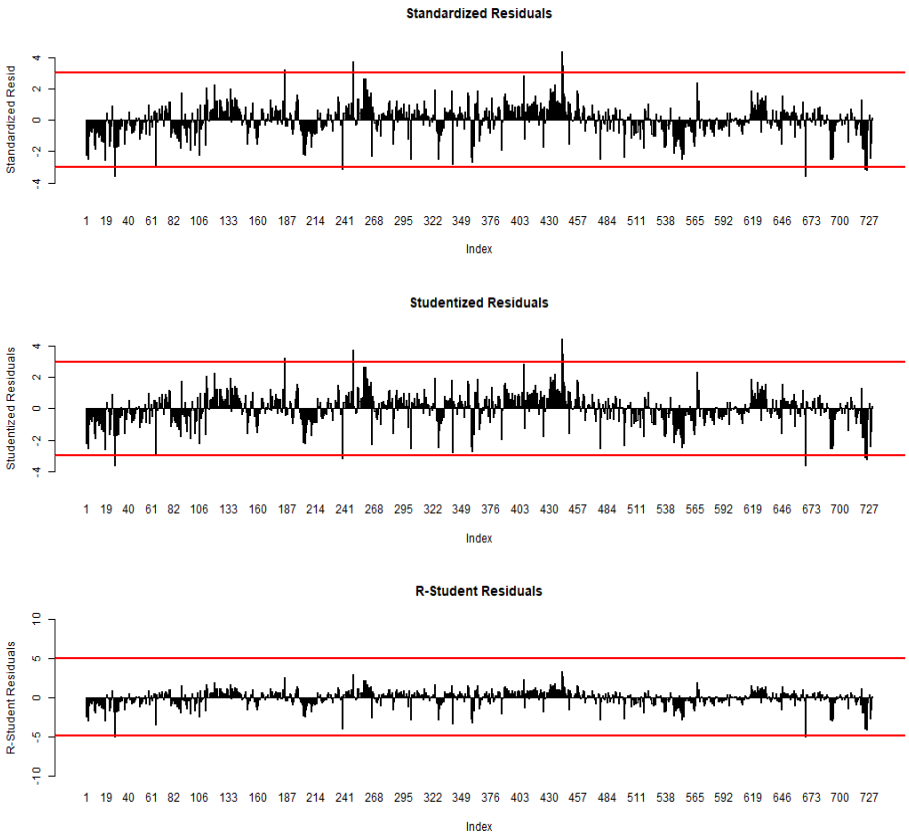
```
Call:
glm.nb(formula = cnt ~ . - registered - casual - dteday - atemp -
    workingday - instant - mnth, data = day, init.theta = 16.63488969,
    link = log)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  7.37348    0.07055 104.507  < 2e-16 ***
season2      0.32864    0.03396   9.677  < 2e-16 ***
season3      0.22234    0.04484   4.958 7.12e-07 ***
season4      0.48022    0.02891  16.609  < 2e-16 ***
yr           0.48155    0.01846  26.085  < 2e-16 ***
holiday     -0.21173    0.05688  -3.723 0.000197 ***
weekday1     0.06506    0.03493   1.862 0.062574 .
weekday2     0.08247    0.03414   2.416 0.015713 *
weekday3     0.07277    0.03422   2.126 0.033464 *
weekday4     0.09459    0.03420   2.765 0.005685 **
weekday5     0.11528    0.03421   3.370 0.000752 ***
weekday6     0.08950    0.03402   2.631 0.008522 **
weathersit2 -0.10274    0.02437  -4.215 2.49e-05 ***
weathersit3 -0.72785    0.06244 -11.657  < 2e-16 ***
temp         1.50579    0.09182  16.400  < 2e-16 ***
hum         -0.29505    0.08876  -3.324 0.000886 ***
windspeed   -0.68154    0.12806  -5.322 1.03e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Negative Binomial(16.6349) family taken to be 1)

    Null deviance: 3121.86  on 730  degrees of freedom
Residual deviance:  747.07  on 714  degrees of freedom
AIC: 12193

Number of Fisher Scoring iterations: 1


              Theta:  16.635
          Std. Err.:  0.875

 2 x log-likelihood:  -12157.160
```

All generalized variance inflation factors (GVIFs) are significantly less than 10 and the GVIF^(1/(2*Df)) are less than 5, thus we have no evidence to suggest that our model has multicollinearity problems.

```
                GVIF Df GVIF^(1/(2*Df))
season      3.540953  3        1.234582
yr          1.031298  1        1.015529
holiday     1.091334  1        1.044669
weekday     1.131968  6        1.010383
weathersit  1.827121  2        1.162631
temp        3.412302  1        1.847242
hum         1.929807  1        1.389175
windspeed   1.190008  1        1.090875
```

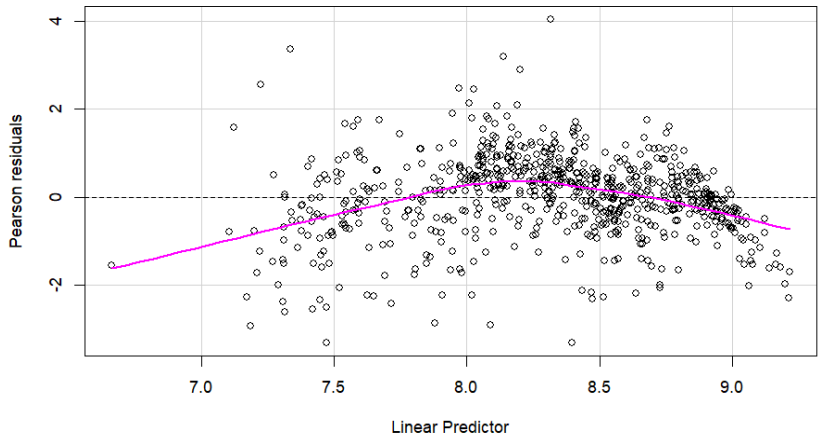
Residual Analysis (Pre-Transformation)

After concluding that our reduced model is the best fit, we conducted residual analysis. We plotted the Standardized Residuals, Studentized Residuals, and R-Student Residuals to observe any extreme values in our data.

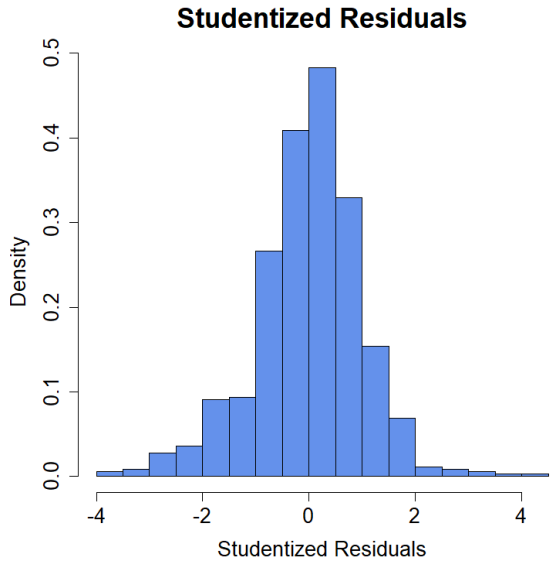


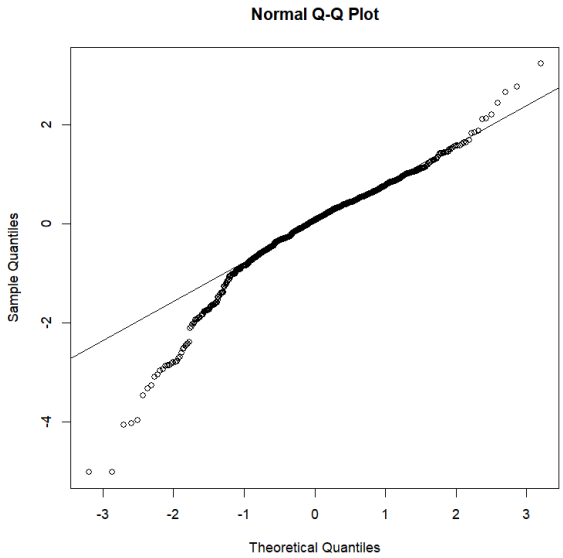
We noticed approximately 9 values in our Standardized Residuals, Studentized Residuals plots and one value in our R-Student Residuals plot that would be considered as extreme.

We plotted our Pearson Residuals vs our fitted values to observe if the points are randomly distributed and if there is a horizontal band around the zero.



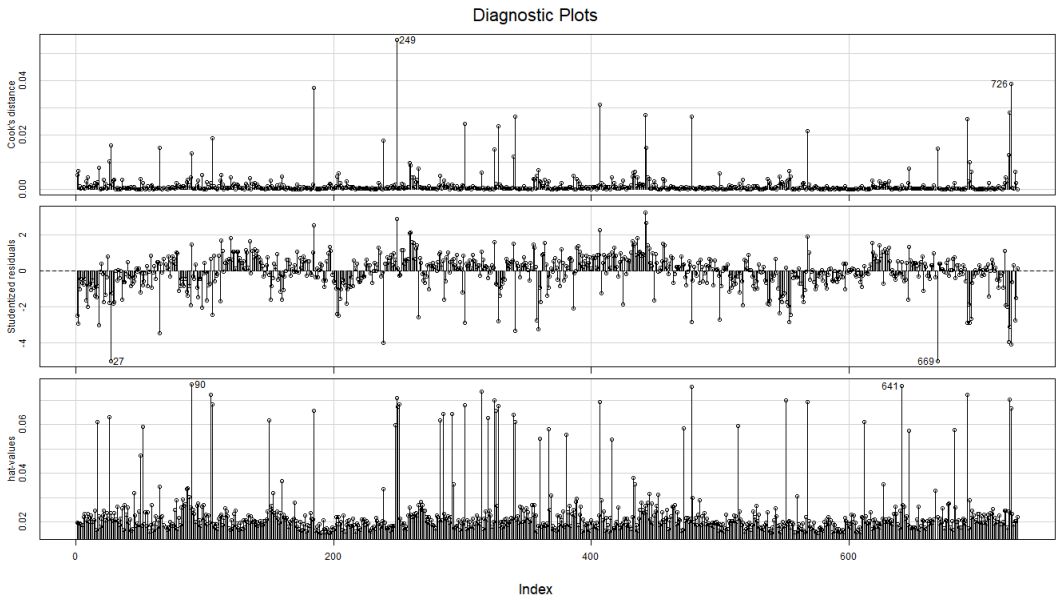
We can see that the points form a slight curve. To rectify this, we considered a transformation. Our barplot of the studentized residuals has a normal distribution. In the qq-plot, the sample quantiles roughly follow a straight line, except near the ends of the plot. Potentially removing influential points may help the normality assumption.





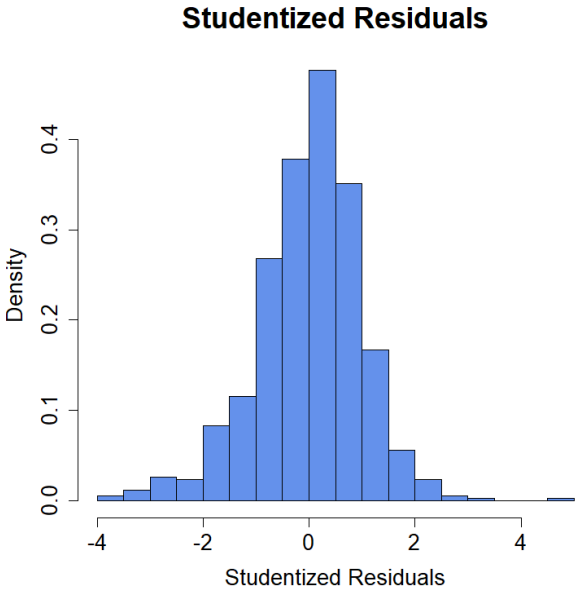
Influential Points Analysis

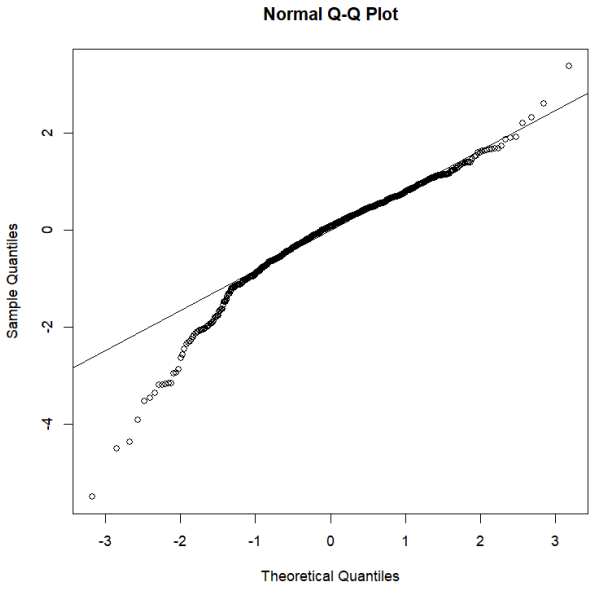
We plotted the Cook's Distance, Studentized Residuals, and hat values to observe any influential points in our data.



Influential Measures	COVRATIO	COOK's D	DFFITS	HAT
# of points	48	0	15	7

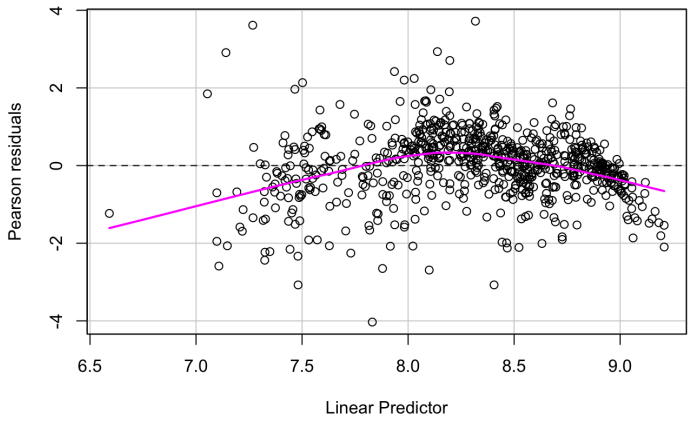
We removed the influential points from our data and plotted a bar plot of studentized residuals and a qq-plot. We still have a normal distribution in our barplot of studentized residuals. Additionally the qq-plot experienced a noticeable difference in the top right of the plot with points falling closer to the line.



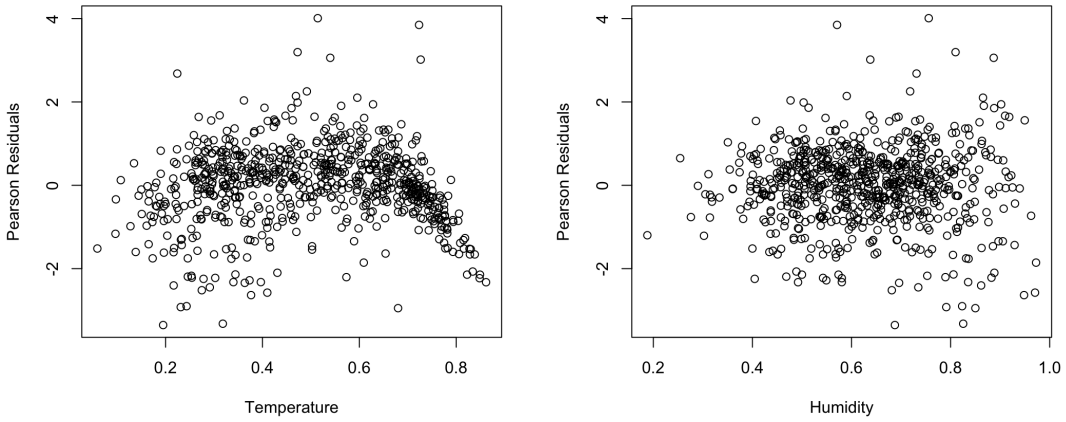


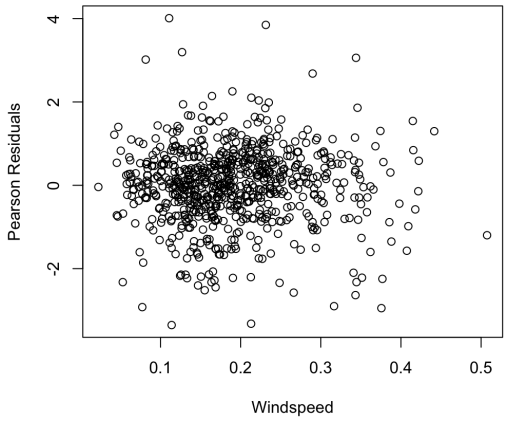
Transformation

Next, we considered whether a transformation may be helpful. Taking another look at the residual plot of our original reduced model, i.e., the one we had before we removed the influential points, we noticed an inverted U shape in the residual plot, which indicates some nonlinearity. Hence, we explored transforming one of the predictor variables. In particular, we looked into adding an additional quadratic term as a predictor variable. When considering which predictors to transform, we only considered the quantitative predictors, since squaring a categorical variable is nonsensical.



To help determine which quantitative predictor to transform, we plot the residuals against the corresponding values of each quantitative predictor. We noticed little curvature in the plot of residuals vs. humidity and the plot of residuals vs. windspeed. In contrast, the plot of residuals vs. temperature is curved, which indicates that higher order terms of temperature should be considered. Hence, we decided to include **temp^2** as an additional predictor variable.

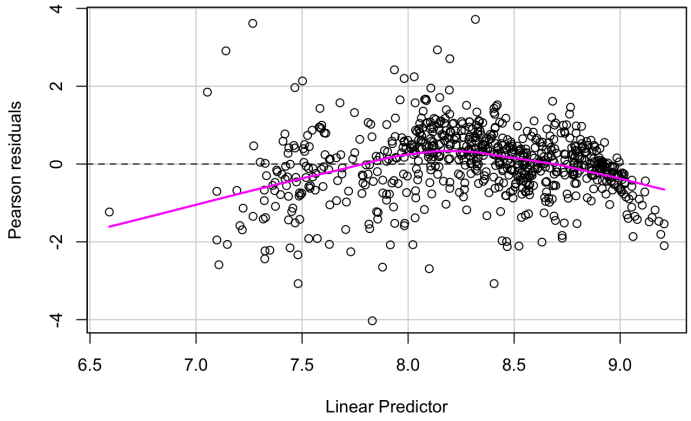


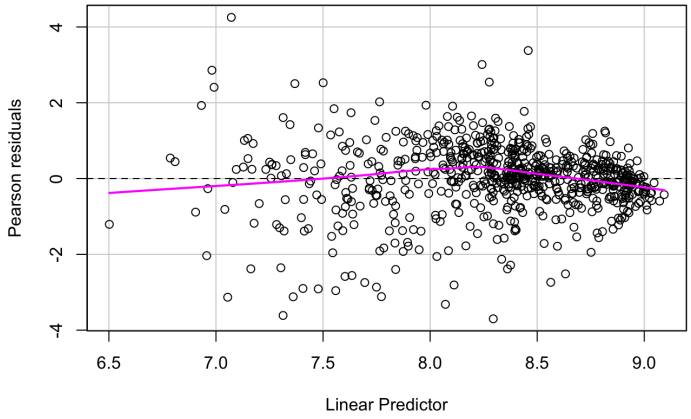


We fitted a new regression model, including **temp^2** as another predictor. We noticed that **temp^2** is indeed a significant predictor, since its p-value is much smaller than 0.05. In addition, the AIC decreased when we included this higher ordered term.

```
Call:
glm.nb(formula = cnt ~ . - registered - casual - dteday - instant -
    atemp - workingday - mnth, data = day2, init.theta = 26.14106762,
    link = log)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.4254  -0.4057   0.0648   0.5095   3.4415

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  6.76782    0.07044  96.078  < 2e-16 ***
season2      0.23668    0.02788   8.489  < 2e-16 ***
season3      0.28496    0.03628   7.855 4.01e-15 ***
season4      0.37333    0.02432  15.349  < 2e-16 ***
yr           0.45443    0.01490  30.497  < 2e-16 ***
holiday     -0.18234    0.04550  -4.007 6.14e-05 ***
weekday1     0.07290    0.02798   2.605 0.009182 **
weekday2     0.07472    0.02728   2.739 0.006164 **
weekday3     0.06948    0.02736   2.540 0.011098 *
weekday4     0.09459    0.02734   3.459 0.000542 ***
weekday5     0.10441    0.02734   3.819 0.000134 ***
weekday6     0.07949    0.02719   2.923 0.003463 **
weathersit2 -0.08511    0.01982  -4.293 1.76e-05 ***
weathersit3 -0.61299    0.05379 -11.397  < 2e-16 ***
temp         5.74843    0.28162  20.412  < 2e-16 ***
hum         -0.58188    0.07580  -7.676 1.64e-14 ***
windspeed   -0.88539    0.10403  -8.511  < 2e-16 ***
temp.sq     -4.44960    0.28546 -15.587  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Negative Binomial(26.1411) family taken to be 1)

    Null deviance: 4623.42  on 728  degrees of freedom
Residual deviance:  737.22  on 711  degrees of freedom
AIC: 11837
```

Moreover, comparing the residual plots from before and after transformation, we notice that the residual plot after transformation is less curved, which indicates that **temp^2** is useful in predicting rented bike counts.





We do notice a slight funnel shape in the plot of residuals vs. the fitted values, which shows nonconstant variance. In general, transformations on the response variable are generally employed to stabilize the variance. However, since our response variable is counts, we wanted to preserve the distribution of our response variable, so we decided that this transformation was sufficient and decided against exploring additional transformations on the response variable.

ANOVA Test

We performed an ANOVA test between our original reduced model (model_cnt4) and this new model. The p-value is less than 0.05, which confirms that the coefficient of **temp^2** is non-zero. Hence, we conclude that **temp^2** contributes significantly to the model.

```
Likelihood ratio tests of Negative Binomial Models

Response: cnt

Model
1           (instant + dteday + season + yr + mnth + holiday + weekday + workingday + weathersit + temp + atemp + hum +
windspeed + casual + registered) - registered - casual - dteday - atemp - workingday - instant - mnth
2 (instant + dteday + season + yr + mnth + holiday + weekday + workingday + weathersit + temp + atemp + hum + windspeed
+ casual + registered + temp.sq) - registered - casual - dteday - atemp - workingday - instant - mnth
     theta Resid. df    2 x log-lik.   Test    df LR stat. Pr(Chi)
1 16.63489       714       -12157.16
2 26.14107       711       -11799.49 1 vs 2     3 357.6717       0
```

Final Model

We assume that our response variable follows a negative binomial distribution with mean μ and dispersion parameter k: y ~ NB(μ, k). The conditional expectation of our response variable is set equal to μ: E[y|X] = μ. The negative binomial regression model uses a log link function, so $\eta = \log(\mu)$, where we get the following two expressions for η.

In the untransformed model:
$\eta = 7.37 + 0.33season2 + 0.22season3 + 0.48season4 + 0.48yr - 0.21holiday + 0.065weekday1 + 0.082weekday2 + 0.073weekday3 + 0.095weekday4 + 0.12weekday5 + 0.090weekday6 - 0.10weathersit2 - 0.73weathersit3 + 1.5temp - 0.30hum - 0.68windspeed$

In the transformed model:
$\eta = 6.77 + 0.24season2 + 0.28season3 + 0.37season4 + 0.45yr - 0.18holiday + 0.073weekday1 + 0.075weekday2 + 0.069weekday3 + 0.095weekday4 + 0.10weekday5 + 0.079weekday6 - 0.085weathersit2 - 0.61weathersit3 + 5.7temp - 4.45temp^2 - 0.58hum - 0.89windspeed$

When modeling counts using the negative binomial distribution with a log link, the effects are multiplicative on the response scale. For example, in the untransformed model, 1.5 represents the change in η per unit change in **temp** when all of the remaining regressor variables are held constant. Since the response scale is the exponent of the link scale, exp(1.5) represents the expected change in the response per unit change in **temp** (when all of the remaining regressor variables are held constant). In the untransformed model, the daily count of rented bikes increases as **temp** increases. In the transformed model, the coefficient of $\mathbf{temp^2}$ is negative, so as **temp** increases, the daily count of rented bikes first increases and then starts to decrease as **temp** takes on large enough values. This agrees with what we would expect: when it is too hot outside, people do not want to go biking as much, so the count of rented bikes would decrease. For given values of the predictors, to get a prediction on the response scale, we simply take exp(η).

Conclusion

In our analysis of bike sharing data, we found that weather conditions and time variables significantly impact the bike count. The impact of each variable is shown through our final model. For our future work, we could extend the project to create separate models for registered and casual users. Additionally, using a dataset with more numerical variables and observations could create a better model that could give more insight into bike counts. By implementing these ideas, we can enhance the models and predictions of the bike counts.

Reflection

Throughout this project, we gained valuable skills and insights in regards to linear models, including how to create, compare, and evaluate model selection. We are satisfied with our overall analysis as our final residual plots look well distributed. However, we did run into standard issues along the way with choosing what type of linear regression model to use and comparing different types of models.

Originally, we had used a regular linear regression model but later took into consideration the nature of the response variable of our data being discrete counts and attempted exploring Poisson modeling. We noticed the large variance within the distribution of our response variable and changed directions to negative binomial modeling to take into account the variance of our data to rework our analysis. After our initial residual analysis, removal of highly influential points and a transformation was deemed necessary to help the distribution of our residual plots and applied this transformation to our final model. Throughout this analysis, we learned about different types of linear models and when to use them depending on distribution that we can apply to future projects.

Appendix

References

Fanaee-T, Hadi. (2013). Bike Sharing. UCI Machine Learning Repository. https://doi.org/10.24432/C5W894.

1.Skilling, T. (2021). *Is a humidity of zero percent possible? WGN*. Available at: https://wgntv.com/weather/weather-blog/is-a-humidity-of-zero-percent-possible/#:~:text=Given%20the%20Earth's%20present%20climate,humidity%20of%20exactly%20zero%20percent. (Accessed: 27 April 2024).

Walker, J. (2020). *Elements of Statistical Modeling for Experimental Biology*. https://www.middleprofessor.com/files/applied-biostatistics_bookdown/_book/generalized-linear-models-i-count-data.html.

Team Responsibilities/Roles

Group members worked together throughout the duration of the project. We meet once a week to discuss progress and provide insight on issues we faced. No specific task was assigned per member and the presentation and report were divided evenly among group members.

Code

```
library(ggplot2)
library(dplyr)
library(reshape2)
library(car)
library(MASS)
library(performance)
day <- read.csv("day.csv")
day$season <- as.factor(day$season)
day$mnth <- as.factor(day$mnth)
day$weekday <- as.factor(day$weekday)
day$weathersit <- as.factor(day$weathersit)
day <- day[-c(69,668),]

#Negative Binomial Models
model_cnt <- glm.nb(cnt ~ . - registered - casual - dteday - instant,
                    data = day)
model_cnt2 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant,
                     data = day) #Bad vif
model_cnt3 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday- instant -
season,
                     data = day) #Good vif
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                     data = day) #Better vif

model_cnt5 <- glm.nb(cnt ~ . - registered - casual - dteday - temp - workingday - instant - mnth,
                     data = day)  #Better AIC
day_wm <- day %>% filter(day$mnth == 6 |
                           day$mnth == 7  |
                           day$mnth == 8 |
```

```
                day$mnth == 10 |
                day$mnth == 11 |
                day$mnth == 12) #Is Removing half of mnths bad, Prob...

model_cnt6 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant -
        season, data = day_wm)
model_cnt7 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth
        - holiday, data = day)

#Poisson Models
pois_cnt <- glm(cnt ~ . - registered - casual - dteday - instant,
            family = "poisson",
            data = day) #multicollinearity issue with temp and atemp
pois_cnt2 <- glm(cnt ~ . - registered - casual - dteday - atemp - workingday - instant,
            family = "poisson",
            data = day) #still has issue with multicollinearity between season and mnth
pois_cnt3 <- glm(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - season,
             family = "poisson",
             data = day) #WORSE AIC
                    # has better RMSE than without mnth
                    # mulitcollinearity problem with mnth and temp?
pois_cnt4 <- glm(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
             family = "poisson",
             data = day) #just gets worse

#Residual Plot
residualPlot(model_cnt4, type = "deviance", quadratic = F)
residualPlot(model_cnt4, type = "rstudent", quadratic = F)
residualPlot(model_cnt4, type = "pearson", quadratic = F)

#Influence Plot
influenceIndexPlot(model_cnt4,
                vars=c("Cook", "Studentized", "hat"))
influenceIndexPlot(model_cnt4)

#FOR RESIDUAL BARPLOTS
fill_model <- model_cnt4

#Standardized Residuals
stdres(fill_model)
```

```
barplot(height = stdres(fill_model), names.arg = 1:length(stdres(fill_model)),
        main = "Standardized Residuals", xlab = "Index",
       ylab = "Standardized Resid", ylim=c(-5,5))
abline(h=3, col = "Red", lwd=2)
abline(h=-3, col = "Red", lwd=2)

#Studentized Residuals
studres(fill_model)
barplot(height = studres(fill_model), names.arg = 1:length(studres(fill_model)),
       main = "Studentized Residuals", xlab = "Index",
       ylab = "Studentized Residuals", ylim=c(-5,5))
abline(h=3, col = "Red", lwd=2)
abline(h=-3, col = "Red", lwd=2)

#R-Student Residuals
rstudent(fill_model)
barplot(height = rstudent(fill_model), names.arg = 1:length(rstudent(fill_model)),
       main = "R-Student Residuals", xlab = "Index",
       ylab = "R-Student Residuals", ylim=c(-10,10))
cor.level <- 0.05/(2*length(rstudent(fill_model)))
cor.qt <- qt(cor.level, 711, lower.tail=F)
abline(h= cor.qt, col = "Red", lwd=2)
abline(h= -cor.qt, col = "Red", lwd=2)

#dfbeta Plots
dfbetaPlots(model_cnt4)

#QQ Plot
par(mfrow=c(1,2))
hist(studres(model_cnt4),
    breaks=20,
    freq=F,
    col="cornflowerblue",
    cex.axis=1.5,
    cex.lab=1.5,
    cex.main=2)
qqPlot(model_cnt4)

# Compares Performance of all Models
library(performance)
```

```
compare_performance(model_cnt,
                    model_cnt2,
                    model_cnt3,
                    model_cnt4,
                    model_cnt6,
                    model_cnt7,
                    pois_cnt,
                    pois_cnt2,
                    pois_cnt3,
                    pois_cnt4)

#VIF of Model
vif(model_cnt4)

#Remove Influential Points
day4 <- day[-as.numeric(row.names(data.frame(summary(influence.measures(model_cnt4))))), ]

#QQPlot for NB and Poisson
hist(studres(model_cnt4),
    breaks=20,
    freq=F,
    col="cornflowerblue",
    cex.axis=1.5,
    cex.lab=1.5,
    cex.main=2)
res <- residuals(model_cnt4, type="deviance")
abline(h=0, lty=2)
qqnorm(res)
qqline(res)

#COMPARE BEFORE AND AFTER REMOVING POINTS, RESIDUALS
par(mfrow = c(2,2))
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                     data = day)
residualPlot(model_cnt4, type = "deviance", quadratic = F)
residualPlot(model_cnt4, type = "rstudent", quadratic = F)
day4 <- day[-as.numeric(row.names(data.frame(summary(influence.measures(model_cnt4))))), ]
model_cnt41 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant -
          mnth, data = day4)
residualPlot(model_cnt41, type = "deviance", quadratic = F)
```

```
residualPlot(model_cnt41, type = "rstudent", quadratic = F)


#COMPARE BEFORE AND AFTER REMOVING POINTS, QQPLOTS
par(mfrow = c(1,2))
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                     data = day)
hist(studres(model_cnt4),
    breaks=20,
    freq=F,
    col="cornflowerblue",
    cex.axis=1.5,
    cex.lab=1.5,
    cex.main=2,
    xlab = "Studentized Residuals",
    main = "Studentized Residuals")
res <- residuals(model_cnt4, type="deviance")
qqnorm(res)
qqline(res)
day4 <- day[-as.numeric(row.names(data.frame(summary(influence.measures(model_cnt4))))), ]
day4 <- day4[-668, ]
day4 <- day4[-69,]
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                     data = day4)
hist(studres(model_cnt4),
    breaks=20,
    freq=F,
    col="cornflowerblue",
    cex.axis=1.5,
    cex.lab=1.5,
    cex.main=2,
    xlab = "Studentized Residuals",
    main = "Studentized Residuals")
res <- residuals(model_cnt4, type="deviance")
qqnorm(res)
qqline(res)

#BEFORE AND AFTER REMOVING POINTS, RESIDUAL BARPLOTS
par(mfrow = c(3,1))
fill_model <- model_cnt4
```

```
#Standardized Residuals
stdres(fill_model)
barplot(height = stdres(fill_model), names.arg = 1:length(stdres(fill_model)),
      main = "Standardized Residuals", xlab = "Index",
      ylab = "Standardized Resid", ylim=c(-5,5))
abline(h=3, col = "Red", lwd=2)
abline(h=-3, col = "Red", lwd=2)

#Studentized Residuals
studres(fill_model)
barplot(height = studres(fill_model), names.arg = 1:length(studres(fill_model)),
      main = "Studentized Residuals", xlab = "Index",
      ylab = "Studentized Residuals", ylim=c(-5,5))
abline(h=3, col = "Red", lwd=2)
abline(h=-3, col = "Red", lwd=2)

#R-Student Residuals
rstudent(fill_model)
barplot(height = rstudent(fill_model), names.arg = 1:length(rstudent(fill_model)),
      main = "R-Student Residuals", xlab = "Index",
      ylab = "R-Student Residuals", ylim=c(-10,10))
cor.level <- 0.05/(2*length(rstudent(fill_model)))
cor.qt <- qt(cor.level, 711, lower.tail=F)
abline(h= cor.qt, col = "Red", lwd=2)
abline(h= -cor.qt, col = "Red", lwd=2)
day4 <- day[-as.numeric(row.names(data.frame(summary(influence.measures(model_cnt4))))), ]
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                  data = day4)
fill_model <- model_cnt4

#Standardized Residuals
stdres(fill_model)
barplot(height = stdres(fill_model), names.arg = 1:length(stdres(fill_model)),
      main = "Standardized Residuals", xlab = "Index",
      ylab = "Standardized Resid", ylim=c(-5,5))
abline(h=3, col = "Red", lwd=2)
abline(h=-3, col = "Red", lwd=2)

#Studentized Residuals
```

```
studres(fill_model)
barplot(height = studres(fill_model), names.arg = 1:length(studres(fill_model)),
      main = "Studentized Residuals", xlab = "Index",
      ylab = "Studentized Residuals", ylim=c(-5,5))
abline(h=3, col = "Red", lwd=2)
abline(h=-3, col = "Red", lwd=2)

#R-Student Residuals
rstudent(fill_model)
barplot(height = rstudent(fill_model), names.arg = 1:length(rstudent(fill_model)),
      main = "R-Student Residuals", xlab = "Index",
      ylab = "R-Student Residuals", ylim=c(-10,10))
cor.level <- 0.05/(2*length(rstudent(fill_model)))
cor.qt <- qt(cor.level, 21, lower.tail=F)
abline(h= cor.qt, col = "Red", lwd=2)
abline(h= -cor.qt, col = "Red", lwd=2)


#Before and After, Influence Plots
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                data = day)
influenceIndexPlot(model_cnt4,
               vars=c("Cook", "Studentized", "hat"))
day4 <- day[-as.numeric(row.names(data.frame(summary(influence.measures(model_cnt4))))), ]
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                data = day4)
influenceIndexPlot(model_cnt4,
               vars=c("Cook", "Studentized", "hat"))

#COMPARE BEFORE AND AFTER REMOVING POINTS, RESIDUALS
par(mfrow = c(2,2))
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                data = day)
residualPlot(model_cnt4, type = "deviance", quadratic = F)
residualPlot(model_cnt4, type = "rstudent", quadratic = F)
day4 <- day[-as.numeric(row.names(data.frame(summary(influence.measures(model_cnt4))))), ]
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - atemp - workingday - instant - mnth,
                data = day4)
residualPlot(model_cnt4, type = "deviance", quadratic = F, id = TRUE)
residualPlot(model_cnt4, type = "rstudent", quadratic = F, id = TRUE)
```

```
#TRANSFORMATION
model_cnt4 <- glm.nb(cnt ~ . - registered - casual - dteday - instant - atemp - workingday -
        mnth, data = day)
summary(model_cnt4)

plot(day$temp, residuals(model_cnt4, type="pearson"), xlab = "Temperature",
    ylab = "Pearson Residuals")
plot(day$hum, residuals(model_cnt4, type="pearson"), xlab = "Humidity",
    ylab = "Pearson Residuals")
plot(day$windspeed, residuals(model_cnt4, type="pearson"), xlab = "Windspeed",
    ylab = "Pearson Residuals")

day7 <- day
day7$temp.sq <- day$temp^2
model_cnt11 <- glm.nb(cnt ~ . - registered - casual - dteday - instant - atemp - workingday -
        mnth, data = day7)
summary(model_cnt11)

residualPlot(model_cnt11, type = "pearson", quadratic = F)

anova(model_cnt4, model_cnt9)
```