4-2022

Weather-Sensitive Walmart Sales Modeling

Stella Rodriguez
Grand Valley State University

Weather-Sensitive Walmart Sales Modeling

Stella Rodriguez

Advisor: Dr. John Gabrosek
Grand Valley State University

Contents

1 Introduction

Walmart has approximately 10,500 stores in 24 countries and employs almost 1.6 million people in the United States[1]. In different regions the stores deal with varying weather conditions affecting sales. Since stores want to make sure to keep items in stock and not overstock, it would be valuable to be able to predict the sales of weather-sensitive items during different weather conditions.

Walmart held a competition on Kaggle (an online community of data scientists), which provided data on various Walmart store locations, item sales, and weather conditions. The data included several data sets containing a years worth of sales data on 111 products sold at 45 Walmart stores, which are covered by 20 weather stations (some stores may share a weather station). The goal for this project is to use the data sets available on the competition page on Kaggle to create a model to predict the sales of weather-sensitive products. The final model will also be interpreted to understand which variables from the data affect the sales of items.

As mentioned in the Kaggle competition description, some of the 111 products "may be a similar item but have a different id in different stores"[4]. As no further explanation has been provided, for the purpose of this project this will be interpreted that there may be products similar to each other but not exact (e.g. different types of milk) that have different item id numbers.

2 Data Management

2.1 Combining Data Sets

For this project there were five total data sets. The data set Stores contains each Walmart store, labeled 1 through 45 for all 45 stores. Stores also contains the size of each store and a *Type* (A, B, or C). When investigated, each *Type* contains a different average *Size*. Type A has an average *Size* of 177,273 square feet, which corresponds to the large Walmart Supercenters' average size. Type B has an average *Size* of 101,053 square feet, which corresponds to Walmart Discount Stores' average size. Type C has an average *Size* of 40,542 square feet, which corresponds to the smaller Walmart Neighborhood Markets average size[3].

The data set Weather contains the date, each station number, and various weather variables for each day for two years. The data set Key contains each store number and their corresponding weather station. The data set Train contains the date, store number, item number, and quantity of units sold. The remaining data set is Test, which contains the date and store number for a different range of dates than Train.

The four data sets, Train, Stores, Weather, and Key must be combined to have information for each date to be complete in one table. This data join is named Train_all. Refer to the Appendex for a step-by-step of the data join process.

2.2 Missing Values

In the Weather data, there are many missing values. As reported in the Weather data documentation, there are some weather variables not normally reported for some weather stations. In particular, weather station 5 only reports temperature variables for one month and weather station 8 does not report various weather variables consistently until a year into the data. These two weather stations are missing entire weather season's worth of data.

To replace these weather stations' missing values, the remaining weather stations are compared to station 5's and station 8's remaining data to find similar weather stations. This is due to having no information on store location or weather station location. The weather on dates where station 5 and station 8 do report weather information are compared to every other weather station and are found to be similar to two other weather stations. For example, both station 8 and station 4 frequently report rain and thunderstorms on the same weeks (when station 8's weather is available). Similarly, station 5 and station 2 share very similar average, min, and max temperatures for the dates when Station 5 does report the temperature variables. Weather station 5's missing values are replaced with station 2 and weather station 8's missing values are replaced with station 4.

There are also other missing values that occur in random order. For these scattered missing values, a moving average is used. This means that a missing value is replaced by an average of values from two days before and two days after that missing value. For example, if station 1 is missing the max temperature for 1/10/2012, then the mean of the max temperatures for the two days before and the two days after 1/10/2012 for station 1 will replace that missing value.

2.3 Creating New Variables

To improve predictions of units sold, new variables are created. The variable *weekday* is created using date, to represent the day of the week. The variable *holiday* is also created using *date*. *Holiday* is a variable which contains only true or false. *Holiday* is true if the date is within a 5 day range of major holidays that potentially have greater amounts of items sold. The holidays used for this variable are:Valentines Day, Easter, Memorial Day, Independence Day, Labor Day, Halloween, Thanksgiving, Black Friday, Christmas, and New Years Day.

2.4 List of Variables

Variable	Description
units	The quantity sold of an item on a given day
date	The day of sales or weather
item_nbr	An id representing one of the 111 products
store_nbr	An id representing one of the 45 stores
station_nbr	An id representing one of 20 weather stations
tmax	Maximum temperature in degrees Fahrenheit
tmin	Minimum temperature in degrees Fahrenheit
tavg	Average temperature in degrees Fahrenheit
dewpoint	Average dewpoint in degrees Fahrenheit
wetbulb	Average wet bulb in degrees Fahrenheit
heat	Heating Degree Days: difference between 65 degrees F and daily temperature mean
cool	Cooling Degree Days: difference between daily temperature mean and 65 degrees F
codesum	Weather phenomena (e.g. TS: Thunderstorm, SN: Snow, RA: Rain)
preciptotal	Total water-equivalent precipitation in inches (rainfall & melted snow)
stnpressure	Average station pressure in inches of Hg
sealevel	Average sea level pressure in inches of Hg
resultspeed	Resultant wind speed in miles per hour
resultdir	Resultant direction of wind in degrees
avgspeed	Average wind speed in miles per hour
type	Size of store, A: Superstore, B: Discount Store, C: Neighborhood Market
weekday	The given date's corresponding day of the week
holiday	True/False dependent on if given date is in a five-day range of a holiday

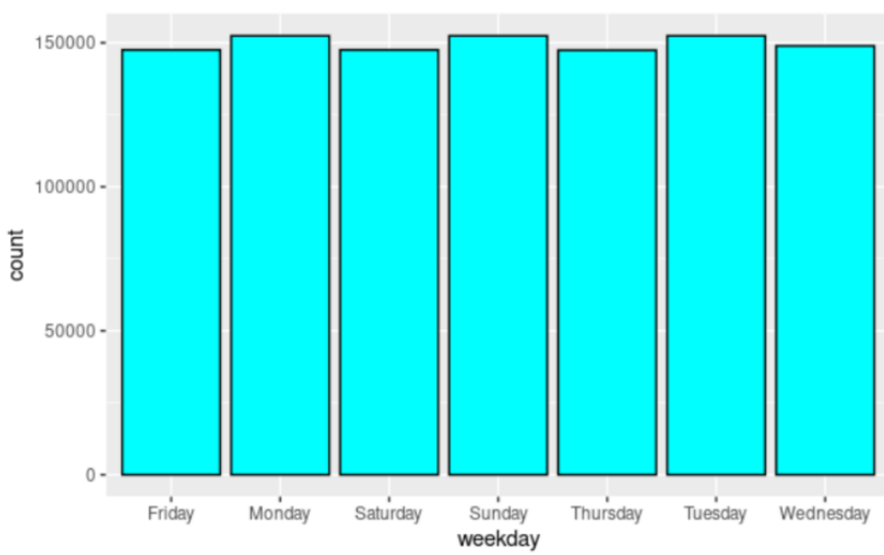
3 Univariate EDA

Univariate Exploratory Data Analysis is used to begin investigating the variables to decide which variables will be used to create the model. The objective of univariate analysis is to explore the distribution of each variable individually. To begin, frequency tables and bar graphs are created for all of the categorical (non-numeric) variables. The categorical variables are checked so that one category does not contain most of observations. *Weekday* has observations almost evenly spread into each category as seen in Table 1 and Figure 1.

Table 1: Frequency Table of weekday

weekday						
Friday	Monday	Saturday	Sunday	Thursday	Tuesday	Wednesday
147519	152403	147519	152403	147408	152403	148920

Figure 1: Bar Graph of weekday



The other categorical variables, *Type* in Table 2 and *holiday* in Table 3 show a decent spread of observations between their categories.

Table 2: Frequency Table of Type

Type		
A	B	C
520992	385725	141858

Table 3: Frequency Table of holiday

```
holiday
  TRUE  FALSE
103119 945456
```

Therefore, the categorical variables *weekday*, *Type*, and *holiday* will continue to be considered for creating the model.

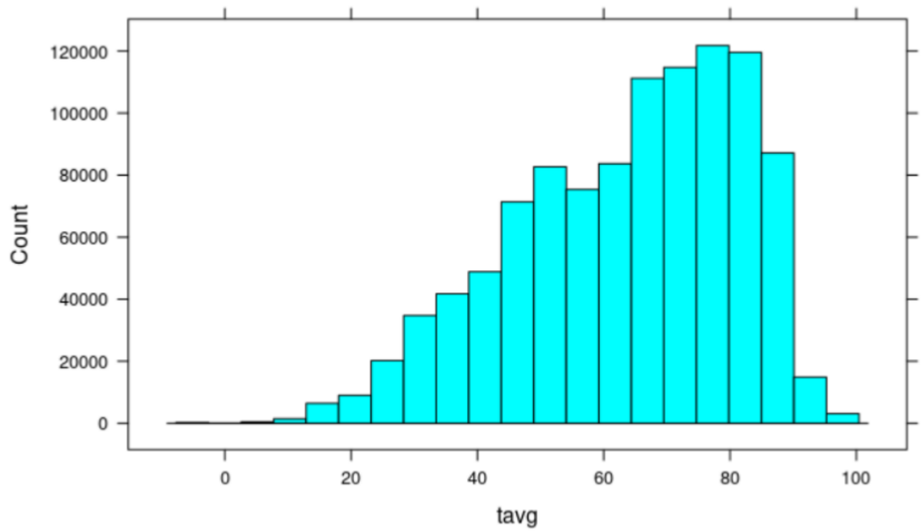
There is one categorical variable that will be regrouped before creating the model. *codesum* reports weather types, and since various weather can occur in one day (e.g. rain and wind), in Train_all *codesum* has 449 different combinations. For *codesum* to be used in the model, the categories will be regrouped into a much smaller number of categories.

For each of the quantitative variables (variables that represent a measurable quantity) a five-number summary and histogram were used. Variables are checked for outliers and checked for values that do not make sense in context of the variable. Table 4 shows summary statistics for average temperature, *tavg*.

Table 4: Summary Statistics for tavg

min <dbl>	Q1 <dbl>	median <dbl>	Q3 <dbl>	max <dbl>	mean <dbl>	sd <dbl>	n <int>
-3	51	67	78	100	63.52222	17.80778	1048575

Figure 2: Histogram of tavg



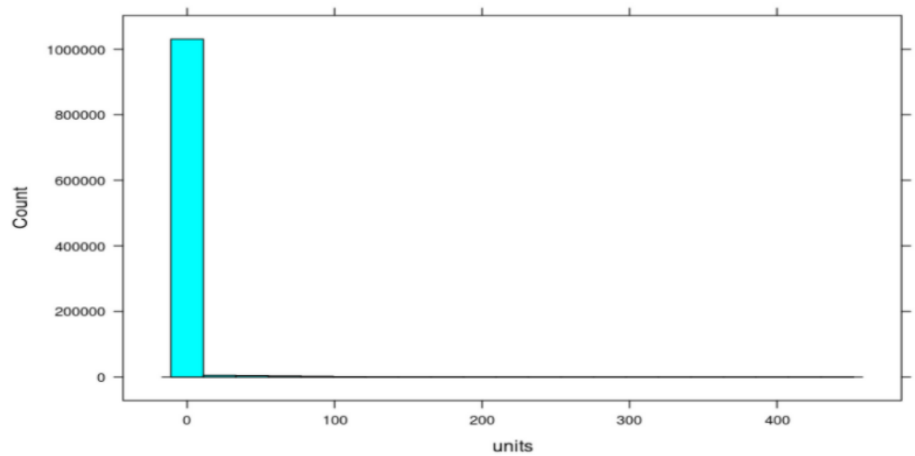
The distribution of the average temperature makes sense with context, only a few values near the extreme temperatures below zero and a majority of values from 60 to 80 degree Fahrenheit. The rest of the quantitative variables are check for major outliers and values that make sense in their context. The variables that have no issues, like *tavg* are kept to explore during Bivariate EDA.

The response variable *units* is also checked for a normal distribution. Table 5 and Figure 3 shows that most of the distribution is near zero.

Table 5: Summary Statistics for units

min <dbl>	Q1 <dbl>	median <dbl>	Q3 <dbl>	max <dbl>	mean <dbl>	sd <dbl>	n <int>
0	0	0	0	441	1.135121	10.58188	1048575

Figure 3: Histogram of units



Units contains an excess amount of zeros observed. In the data documentation it is mentioned that the sales data does not differentiate between stock and demand. 0 units sold can mean that the item was in stock but none were purchased or that the product was out of stock. This causes the over-abundance of zeros. This is taken into consideration when building a model.

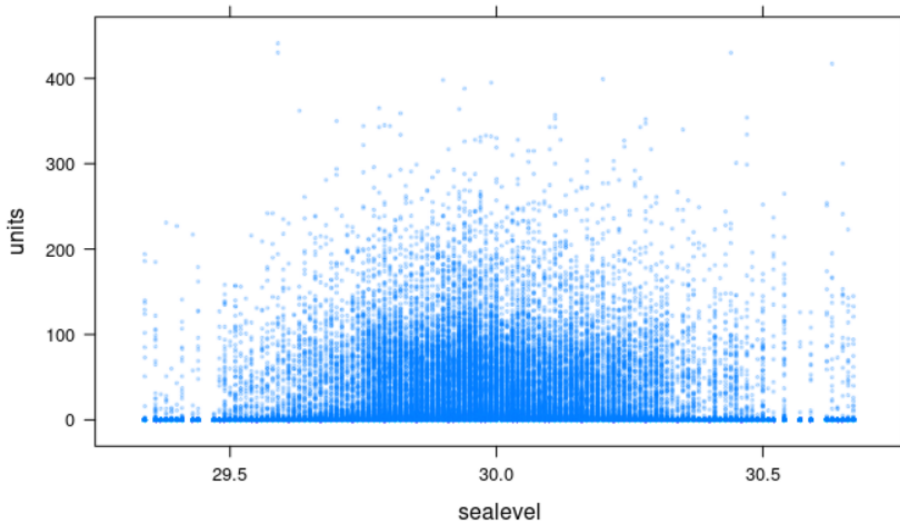
4 Bivariate EDA

During Bivariate exploratory data analysis, the relationships between the potential predictor variables and the response variable *units* are explored.

Scatterplots and correlation coefficients are calculated for all of the potential quantitative predictors with *units*. A correlation coefficient is a measure of the strength of the linear relationship between the two variables. The values range between -1 and 1. A correlation coefficient closer to -1 or 1 (farther from 0) signifies a stronger linear relationship. We will remove variables where there is a very weak relationship between the variable and *units*.

Looking at *sealevel*, we obtain the scatterplot Figure 4.

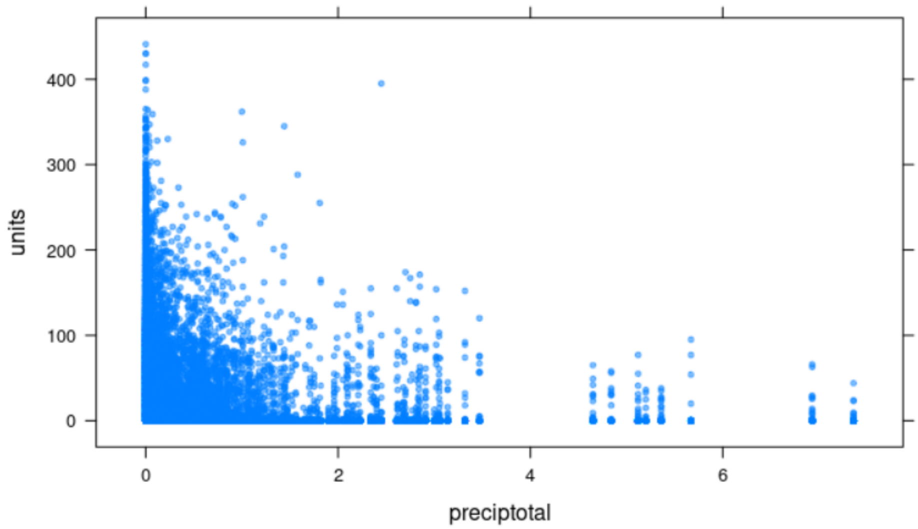
Figure 4: Scatterplot of units by sealevel



The correlation between *sealevel* and *units* is 0.0003, which is very small, so *sealevel* is removed from the potential predictors. One other variable, *cool*, which has a correlation coefficient smaller than *sealevel*, is also removed for not having a strong enough relationship with *units*.

An example of a stronger relationship between a potential predictor and *units* is shown by Figure 5.

Figure 5: Scatterplot of units by preciptotal



Notice that on the horizontal axis, as *preciptotal* increases, *units*, on the vertical axis, decreases.

Colinearity between the quantitative predictors is also taken into consideration. This is where two predictors have a strong relationship which means they have similar information about the response variable. It is unnecessary to keep both variables in the model. If two predictors have a correlation coefficient greater than 0.8, then we will keep the predictor that is more highly correlated with *units*. After reviewing the correlations between predictors, the variables *tmin*, *tavg*, *tmax*, *wetbulb*, *resultdir*, *resultspeed*, and *heat* are removed from potential predictors.

5 Modeling

5.1 Variable Regrouping

Before modeling, there is a categorical variable that needs to be reorganized. As mentioned in the univariate EDA, *codesum* will be regrouped into a smaller number of categories. *Codeum* will be regrouped into three possible categories:extreme weather, minor weather, and moderate weather. Any *codesum* value that includes the reported weather of tornado/waterspout, funnel cloud, duststorm, sandstorm, thunderstorm, heavy fog (less than .25 miles visibility), or volcanic ash will be categorized as extreme weather. All other weather possibilities that do not include extreme weather (e.g. mist, rain), are categorized as minor weather. Weather that is not adverse is labeled as “Moderate” in *codesum*, so Moderate weather will be labeled as such.

For the categorical variables to be used in the model, they must be transformed into dummy variables, which take the value of 1 or 0 to indicate the presence or absence of the category. The variable *weekend* is transformed into *DW.wkend*, where a one indicates that it is a weekend (Saturday or Sunday) and a zero indicates a weekday.

The *codesum* variable is transformed into *DCS.ex* and *DCS.min*. *DCS.ex* is equal to one if there is extreme weather present and a zero otherwise. *DCS.min* is equal to one if there is minor weather present and a zero otherwise. For example, if there is extreme weather, then $DCS.ex = 1$ and $DCS.min = 0$. If there is minor weather, then $DCS.ex = 0$ and $DCS.min = 1$. In the last case, if there is moderate weather, then $DCS.ex = 0$ and $DCS.min = 0$.

The variable *Type* is transformed into *DType.A* and *DType.B*. *DType.A* is equal to one if the store is an A sized Superstore and equal to zero otherwise. *DType.B* is equal to one if the store is a B sized Discount store and equal to zero otherwise. For example, if the store is a Superstore ($Type = A$), then $DType.A = 1$ and $DType.B = 0$. If the store is a Discount Store ($Type = B$), then $DType.A = 0$ and $DType.B = 1$. If the store is a Neighborhood Market ($Type = C$), then $DType.A = 0$ and $DType.B = 0$

5.2 Model Building

Due to the excess zeros in the response variable *units*, zero-inflated models are considered for modeling this data. Zero-inflated models like Zero-inflated Poisson or Zero-Inflated Negative Binomial (ZINB) take into account count data with excess zeros[2]. After further research into zero-inflated models, ZINB is found to be the best fit for this data. ZINB fits data better when variance is larger than the mean, as was indicated in Table 3, where the mean for *units* was 1.14 and the standard deviation for *units* was 10.58 [5]. A ZINB model is composed of two parts. The first part of the model accounts for excess zeros by increasing the probability of zero units sold; the second part models the units sold that aren’t part of the excess zeros. In other words, the second part models the count of units sold.

ZINB used with this data is shown below.

$$E(n_{\text{units sold}} = k) = P(\text{not in stock}) * 0 + P(\text{in stock}) * E(y = k|\text{in stock})$$

The left side equals the number of units of an item sold equals k. The first term on the right side equals the probability of an item not being in stock multiplied by zero. This is added to the second term on the right side which is the probability of the item being in stock multiplied by the number of units sold given that the item is in stock.

The probability uses the logistic model:

$$\frac{\lambda}{1+\lambda},$$

where

$$\lambda = e^{C_0 + C_1 Z_1 + C_2 Z_2 + \ldots + C_n Z_n}$$

and the count is of units sold uses the negative binomial model:

$$\frac{\Gamma(y+\theta)}{\Gamma(\theta)\Gamma(y+1)} \left(\frac{1}{1+\theta^{-1}\mu}\right)^{\theta} \left(\frac{\theta^{-1}\mu}{1+\theta^{-1}\mu}\right)^{y}$$

where

$$y = 0 \text{ or } y = 1, 2, 3, \ldots \text{ and}$$
$$\mu = e^{B_0 + B_1 X_1 + B_2 X_2 + \ldots + B_n X_n}.$$

Now that the type of model is chosen, the variables must be fit to the model. The first arrangement of variables into the model's two different components is based on the assumptions of whether the variables would better explain if an item was in or out of stock or explain the number of units sold. For the first model, the variables used in the probability of an item being in stock are: *DW.wkend*, *DType.A*, *DType.B*. The variables used in the count model are: *store_nbr*, *item_nbr*, *station_nbr*, *dewpoint*, *preciptotal*, *avgspeed*, *holiday*, *DCS.ex*, and *DCS.min*.

To find the best model, different combinations of the variables will be made into models and tested against the previous model using the Vuong test. The Vuong test compares the fit of the two models and suggests the better one[5]. To test the current model, a second model is created changing only one variable. The change can be a variable moved from one part of two components of the model to the other, a variable removed from the model altogether, or a variable added back into the model after being removed. Table 6 is a result of a Vuong test run on two models using this data, where model1 is found to be the preferred model over model2.

Table 6: Result of Vuong Test

	Vuong z-statistic <dbl>	H_A <chr>	p-value <chr>
Raw	2.0873609	model1 > model2	0.018428
AIC-corrected	1.8527385	model1 > model2	0.031960
BIC-corrected	0.4610822	model1 > model2	0.322370

After several models are created and tested against each other, the final model is created. After creating the final model, when tested against other similar models, the final one is always found to be the preferred model using the Vuong test. Also, each variable's significance in the model is checked using the p-value in the output (significant if less than 0.05).

The following is the output of the final model, including the coefficients. Notice that all the p-values of the final model, listed on the far right of each variable as $\Pr(>|z|)$, are less than 0.05.

Figure 6: Final Model Coefficients

```
Call:
zeroinfl(formula = units ~ store_nbr + item_nbr + station_nbr + avgspeed +
    DCS.min + Dw.wkend | DCS.ex + preciptotal + Dtype.A + Dtype.B, data = train_all,
    dist = "negbin")

Pearson residuals:
     Min       1Q   Median       3Q      Max
-0.11835 -0.09606 -0.09431 -0.09257 85.49605

Count model coefficients (negbin with log link):
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.8550283  0.0422044  91.342  < 2e-16 ***
store_nbr    0.0120680  0.0008160  14.789  < 2e-16 ***
item_nbr    -0.0348295  0.0005139 -67.780  < 2e-16 ***
station_nbr  0.0148984  0.0018451   8.075 6.77e-16 ***
avgspeed     0.0313044  0.0025449  12.301  < 2e-16 ***
DCS.min     -0.0947273  0.0211601  -4.477 7.58e-06 ***
Dw.wkend     0.2427590  0.0212744  11.411  < 2e-16 ***
Log(theta)  -1.2923020  0.0247415 -52.232  < 2e-16 ***

Zero-inflation model coefficients (binomial with logit link):
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.21369    0.02121 151.506  < 2e-16 ***
DCS.ex      -0.03603    0.01812  -1.988 0.046798 *
preciptotal -0.05716    0.01318  -4.336 1.45e-05 ***
Dtype.A     -0.06629    0.01905  -3.480 0.000502 ***
Dtype.B     -0.11489    0.01970  -5.831 5.52e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Theta = 0.2746
Number of iterations in BFGS optimization: 21
Log-likelihood: -2.586e+05 on 13 Df
```

The final model includes the following variables in the count model *store_nbr*, *item_nbr*, *station_nbr*, *avgspeed*, *DCS.min*, and *DW.wkend*. The logistic model includes the variables: *DCS.ex*, *preciptotal*, *DType.A*, and *DType.B*. The coefficients given from the output can now be substituted into the final model.

The maximum likelihood estimate of

$$\frac{\lambda}{1+\lambda},$$

the logistic model, is where

$$\lambda = e^{3.21369-0.03603DCS.ex-0.05716preciptotal-0.06629DType.A-0.11489DType.B}$$

and the negative binomial model:

$$\frac{\Gamma(y+0.2746)}{\Gamma(0.2746)\Gamma(y+1)}\left(\frac{1}{1+4.1195\mu}\right)^{0.2746}\left(\frac{4.1195\mu}{1+4.1195\mu}\right)^{y}$$

where

$$y = 0 \text{ or } y = 1, 2, 3, ... \text{ and}$$

$$\mu = e^{3.8550+0.0121 store\_nbr - 0.0348 item\_nbr + 0.0149 station\_nbr + 0.0313 avgspeed - 0.0947 DCS.min + 0.2428 DW.wkend}.$$

Recall that the combination of these two models is the full final model.

6 Model Usage

6.1 Interpreting Coefficients

Using the model output, the coefficients are interpreted.
For the probability model:
DCS.ex: The odds that the item is in stock for extreme weather is $e^{0.03127} = 1.032$ times less than the odds that the item is in stock for a moderate weather day. Thus, if the weather is extreme, the less likely that the item is in stock[6].
preciptotal: If *preciptotal* increases by one, the odds that the item is in stock decreases by a factor of $e^{0.06061} = 1.062$.
DType.A: The odds that the item is in stock for a type A Superstore is $e^{0.06162} = 1.064$ times less than the odds that the item is in stock for a small type C Neighboorhood Market store. Thus, if the store is the large Superstore, the less likely that the item is in stock.
DType.B: The odds that the item is in stock for a type B Discount Store is $e^{0.10954} = 1.116$ times less than the odds that the item is in stock for a small type C Neighborhood Market store. Thus, if the store is a Neighborhood Market store, the less likely that the item is in stock.

For the count model:
store_nbr: If *store_nbr* increases by one, the expected units sold would increase by $e^{0.0116070} = 1.012$ while holding all other variables in the model constant.
item_nbr: If *item_nbr* increases by one, the expected units sold would increase by $e^{0.0355868} = 1.036$ while holding all other variables in the model constant.

station_nbr: If *station_nbr* increases by one, the expected units sold would increase by $e^{0.0130894} = 1.013$ while holding all other variables in the model constant.
DCS.min: The expected number of units sold for minor weather is $e^{0.1148114} = 1.122$ times less than the expected number of units sold for a moderate weather day while holding all other variables in the model constant.
DW.wkend: The expected number of units sold for a weekend is $e^{0.2517515} = 1.286$ times more than the expected number of units sold for a weekday while holding all other variables in the model constant.

6.2 Examples Using Model

To use the final model, two examples will be tested. The first test will be of a high probability of an item being in stock. The values in Table 7 are used for this example.

Table 7: Variable Values for Example 1

DCS.ex	0
preciptotal	0
DType.A	0
DType.B	0
store_nbr	40
item_nbr	2
station_nbr	20
avgspeed	15
DCS.min	0
DW.wkend	1

Substituting these values into the logistic model:

$$\frac{\lambda}{1+\lambda},$$

where

$$\begin{aligned}\lambda &= e^{3.21369-0.03603(0)-0.05716(0)-0.06629(0)-0.11489(0)} \\ &= 24.87\end{aligned}$$

and into the count model:

$$\frac{\Gamma(y+0.2746)}{\Gamma(0.2746)\Gamma(y+1)}\left(\frac{1}{1+4.1195\mu}\right)^{0.2746}\left(\frac{4.1195\mu}{1+4.1195\mu}\right)^{y}$$

where

$$\begin{aligned} &y = 0 \text{ or } y = 1, 2, 3, \ldots \text{ and} \\ &\mu = e^{3.8550+0.0121(40)-0.0348(2)+0.0149(20)+0.0313(15)-0.0947(0)+0.2428(1)} \\ &\quad = 196.31 \end{aligned}$$

The final model predicts that 7.58 units are sold with the attributes in Table 7.

The second example will be of a low probability of an item being in stock. The values in Table 8 are used for this example.

Table 8: Variable Values for Example 2

DCS.ex	1
preciptotal	6
DType.A	0
DType.B	1
store_nbr	1
item_nbr	100
station_nbr	2
avgspeed	1
DCS.min	0
DW.wkend	0

Substituting these values into the logistic model:

$$\frac{\lambda}{1+\lambda},$$

where

$$\begin{aligned} \lambda &= e^{3.21369-0.03603(1)-0.05716(6)-0.06629(0)-0.11489(1)} \\ &= 15.18 \end{aligned}$$

and into the count model:

$$\frac{\Gamma(y+0.2746)}{\Gamma(0.2746)\Gamma(y+1)}\left(\frac{1}{1+4.1195\mu}\right)^{0.2746}\left(\frac{4.1195\mu}{1+4.1195\mu}\right)^{y}$$

where

$$\begin{aligned}
y &= 0 \text{ or } y = 1, 2, 3, ... \text{ and} \\
\mu &= e^{3.8550+0.0121(1)-0.0348(100)+0.0149(2)+0.0313(1)-0.0947(1)+0.2428(0)} \\
&= 1.47
\end{aligned}$$

The final model predicts that 0.096 units are sold with the attributes in Table 8.

7 Conclusion

As expected, different weather affects the expected number of units sold and the probability of items being in stock. With the final model different Walmart store locations are able to predict the expected number of units sold for their weather-sensitive items, as shown by the examples. Predicting demand allows Walmart stores to stock more or to not over-stock for different items while observing upcoming weather. Walmart can more efficiently allocate store inventory and even work with their distribution centers to prepare ahead for extreme weather.

A Appendix

In order to use all the variables for the model, the data sets Train, Store, Weather, and Key were joined together. The data sets were joined one-by-one using common variables in the following order.

- Weather and Key were joined into Weather_key by the common variable *station_nbr*.
- Weather_key and Stores were joined into Weather_stores by the common variable *store_nbr*.
- Train and Weather_Stores were joined into Train_all by the common variables *store_nbr* and *date*.

With the data join each item at all stores and every day for the dates in Train has information on the weather of that day, store type, and number of units sold.

References

[1] About. Corporate. (n.d.). Retrieved April 16, 2022, from https://corporate.walmart.com/about.

[2] Loeys, T., Moerkerke, B., De Smet, O.,& Buysse, A. (2011). The analysis of zero-inflated count data: Beyond zero-inflated Poisson regression. British Journal of Mathematical and Statistical Psychology, 65(1), 163–180. Retrieved from https://www.researchgate.net/publication/51676806_The_analysis_of_zero-inflated_count_data_Beyond_zero-inflated_Poisson_regression.

[3] Our Retail Divisions. Corporate. (2005, January 6). Retrieved April 20, 2022, from https://corporate.walmart.com/newsroom/2005/01/06/our-retail-divisions.

[4] Walmart recruiting II: Sales in Stormy Weather. Kaggle. (n.d.). Retrieved April 16, 2022, from https://www.kaggle.com/competitions/walmart-recruiting-sales-in-stormy-weather/overview.

[5] ZERO-INFLATED NEGATIVE BINOMIAL REGRESSION — R DATA ANALYSIS EXAMPLES. UCLA: Statistical Consulting Group. from https://stats.oarc.ucla.edu/r/dae/zinb/.

[6] ZERO-INFLATED NEGATIVE BINOMIAL REGRESSION — STATA ANNOTATED OUTPUT. UCLA: Statistical Consulting Group. from https://stats.oarc.ucla.edu/stata/output/zero-inflated-negative-binomial-regression/.