Predicting Divvy Stocking Requirements

The Problem:

Divvy is a Chicago based bike sharing company with 585 stations across the city. Divvy is a great way to get around Chicago, but getting from point A to point B smoothly relies on an available bike at your desired starting point, and an empty slot to check out your bike at your destination. If a customer looks to start their journey, and finds out there aren't any bikes available at the closest station, this will sour their experience as they now have to travel to another station or find another mode of transport. The same is true if a customer plans to end their trip at a particular location and finds there are not any empty slots available to dock their bike. They now have to find another location within their trip time window, and backtrack to their desired destination.

By analyzing historical trip data we can uncover insights on proper stocking requirements for all Divvy stations. These insights will be instrumental in helping Divvy's operations team stay in touch with customer needs. Properly stocked stations are essential for retaining customers and growing the Divvy brand.

The Data:

The 2017 Divvy bike sharing data is available on the Divvy website in two main parts; trip data, and bike station data. The trip data contains a unique trip id for each trip taken and contains information on the timing of trips, stations used, and what type of user took the ride. The two user types refer to a “customer” who purchased a 24 hour pass, and a “subscriber” who purchased an annual membership. For Subscribers there is gender and birth year information as well. In the station data the main piece of information that we’re interested in is the station capacity (how many docking slots are available at each station). There is also information on the location of all the bike stations and the date they went online as well.

The trip data for 2017 came in four csv files, one for each quarter. These were loaded into dataframes individually so that the shape and the column names could be looked at closer. Once the formats of the individual data frames were aligned, they were concatenated and then merged with station data. The station data for Q3 and Q4 of 2017 was selected to provide the most up to date station info.

Two merges were performed in total. One matching the station id in the station file to the starting station id in the trip file, and the second matching the station id in the station file to the ending station id in the trip file. This will allow us to view both the information for the starting and ending stations on the same line for each trip.

After the merge duplicate columns and rows were dropped, and data types were changed to appropriately represent numbers, categories, and date time objects. During the inspection of the data I noticed a fair number of riders whose birth years listed would make them over 100 years old. As people of this age are not likely to be riding a bike around Chicago it did not seem that all of that information was accurate. This may be caused from users entering incorrect birth

years while signing up, but because of the inaccuracies and that this info is only available for subscribers, it will not be very helpful in grouping all riders by age. Because this project's main focus is on station usage and proper stocking requirements, the birth_year column was dropped from the data set as well.

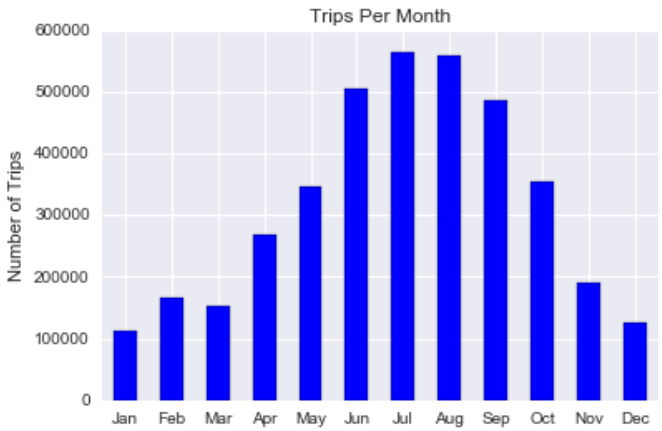
While checking for outliers in the data I found "Dependent" as value for user type. Customer and Subscriber were the only expected user types, but after grouping the data by user type, it was clear that there were a very small number of rides marked as "Dependent" in this category as well. Looking closer I noticed some those rides had gender information and some did not. Because the gender information was only tracked for subscribers, the dependents with that information were changed to "subscriber," and those that didn't were changed to "customer" to avoid any confusion down the line.

Looking closer at the station capacity I found that the minimum capacity value was 0. As this threw up a red flag, I looked closer and found that two of the stations on the Q3 & Q4 stations file were listed with 0 capacity. This could have been because those stations were closed down in the second half of 2017, but because the trip data is available for all of 2017, it was important to find the correct capacity of those stations when the rides were completed. Our best option was to load in the Q1 & Q2 station file for 2017, and use that to find the correct capacities for those stations and update that information in the combined data frame.

Continuing the search for any outliers, the trip durations were then inspected. The trip data from Divvy's website claimed that any trips shorter than one minute or longer than 24 hours were excluded. I found this to be the case and there weren't any trips included outside of those parameters.

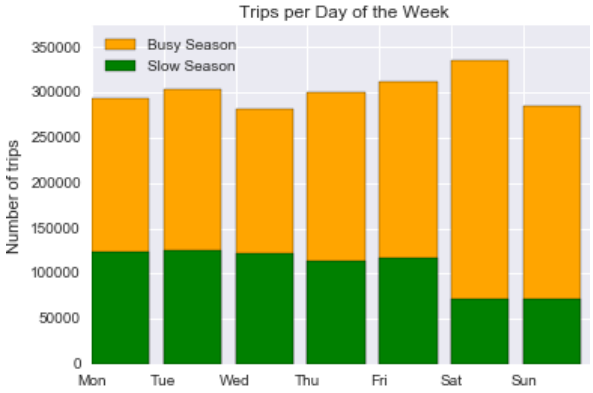
Exploratory Analysis:

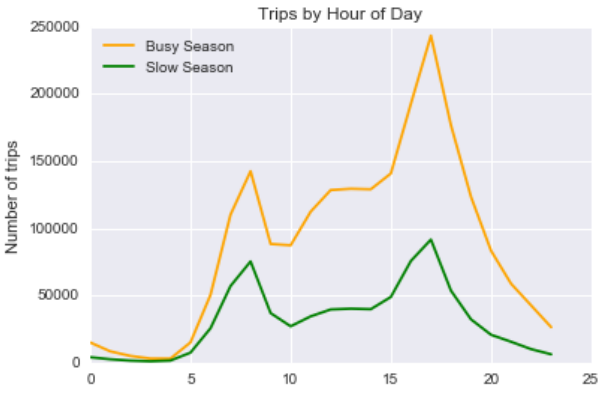
In order to make sure bike stations are properly stocked, I first looked at when Divvy riders are using the service. Looking at the number of bike trips taken in 2017 by month, it may not be surprising to see that the most trips are completed in the summer months, June through September. You'll notice that the slow season is found in the winter months, and the middle tier we see as April, May, and October mark the changing of the seasons.



The consistent number of rides in the winter months might suggest that not all of these trips are solely recreational rides on nice days around the city, but are part of the customer's daily routines or commutes. To explore this further, I took a look at a breakdown of trips by day of the week and

different parts of the day. I found that overall Mondays and Tuesdays had the highest trip counts in 2017, and that there is a spike in the number of trips around 8am and 5pm, prime commuting hours. Although those statistics reflect trip information for the year as a whole, it's important to see how those numbers fluctuate in busy and slow seasons. In the summer months (below to the left in orange) we can see that Friday and Saturday are the busiest days, while in the winter months (green) Tuesday is the busiest day. We also can see that Saturday and Sunday both have lower trip counts than every weekday during the winter. Despite Saturday being the busiest day in the high season, the average trip numbers per day were not found to be statistically significant for weekends compared to weekdays during that time.

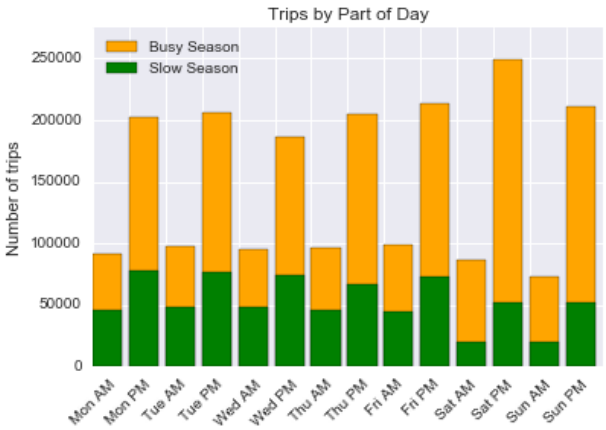




I took a closer look at the number of trips by hour of the day (above to the right) as well. You'll see that most of the trips occur around 8am and 5pm in the slow months as too.

As 5pm was the highest trafficked hour I wanted to see if this could be attributed to it being a popular commuting hour. To test this I took a two sample t-test to determine if the average number of rides in the 5pm hour for weekdays was equal to the number of rides at the 5pm hour on the weekends. The test resulted in a very small p-value of 2.44e-15 letting us know that we could reject the hypothesis that the averages were the same. Using bootstrap replicates I determined the probability of a getting a higher average ride count at 5pm on the weekdays than the weekends was near 100%. The weekdays are a big contributor to the spike at 5pm.

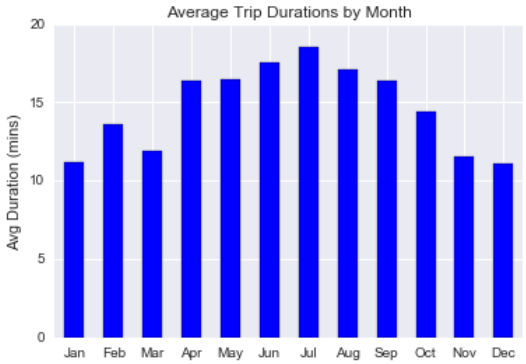
Looking closer at the number of rides by hour, it seemed that the overall number of rides in the afternoon seemed to be higher than the morning hours. I took a look at the breakdown by day of the week to try to get more insight.

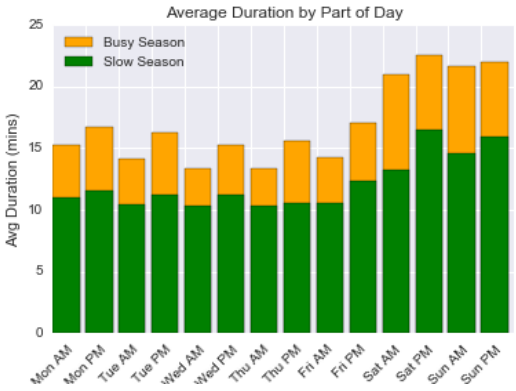


It turns out that in both the high and low seasons there are more trips in the PM hours than AM hours on any given day of the week.

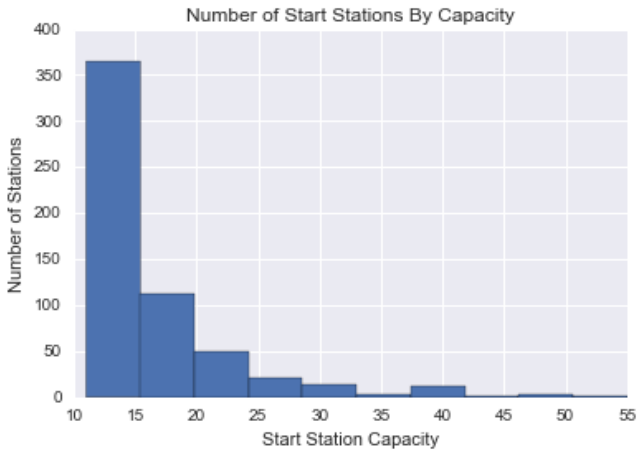
As the number of trips alone doesn't give us a full picture on how Divvy riders are using the service, I took a look at trip durations as well. I found that the average trip duration is 15.92 minutes and the majority of trips last between 5 and 15 mins. We see longer trip durations in the summer months than the winter, and

longer trips on the weekends than weekdays. I also found that the longest trip durations seem to come at 1am and 2pm throughout the year (20.07 minutes and 19.56 minutes respectively), outside of the typical commuting hours. Trip durations during the prime commuting hours 8am and 5pm are 14.05 minutes on average.





After taking some time to understand Divvy riders behavior, the next step was to take a closer look at the network of stations and their capacities. I found that most trips are coming from and ending at stations with capacities of 10-25. As the max capacity for a station in the city is 55, I needed to take a closer look at why it doesn't seem traffic is correlating with a higher station capacity. Looking at the distribution of capacities across all stations I found that an overwhelming majority of stations fall between a capacity of 10-15. This helps explain why much of the traffic from a high level perspective seems to be connected to lower capacity stations. Only 7 out of 585 stations have a capacity over 40.

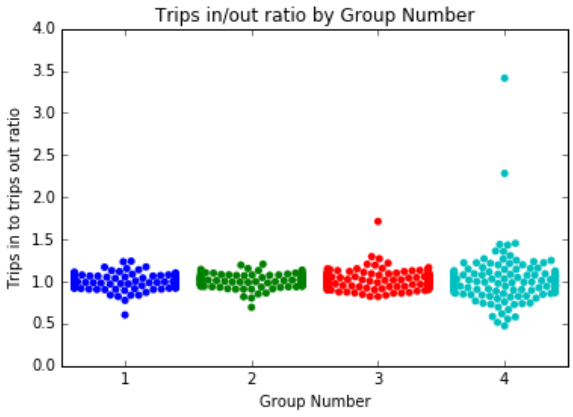


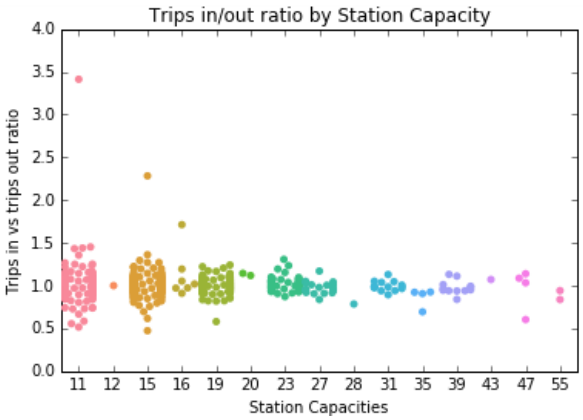
To look at this a different way I grouped all of the stations into 4 equal tiers based on their number of trips for the year. Highly trafficked stations were marked as group 1 and the lowest trafficked stations were marked as group 4. Looking at the average capacity for each group, the average capacities were higher in each group as you move up to the higher trafficked tiers. The average capacities can be found below:

Average Start Station Capacity	Average End Station Capacity
Group 1 27.5882739203	Group 1 25.9218887618
Group 2 17.6941599153	Group 2 21.2396421509
Group 3 15.2839083074	Group 3 18.6819500757
Group 4 13.644185503	Group 4 15.8927581717

Next I wanted to investigate if there was a large change in which stations were used in busy and slow seasons. I found that 15 of the top 25 start stations and 15 of the top 25 end stations remained constant in busy to slow seasons. When I ran the same analysis for the top 50 start and end stations, I discovered 33 start stations and 31 end stations remained constant from busy to slow seasons. As about 60% of top stations remain constant in summer and winter months, it further enforces the theory that a large portion of Divvy trips are completed as a part of a daily routine that extends year round.

An important aspect I wanted to dive deeper into was the ratio of bikes that come in and out of each of the stations. To investigate this I calculated the in/out ratio for each station by dividing the total number of rides ending at each station by the total number of rides starting at each station for 2017. I found that neither number of rides, or station capacity had a significant impact on the average ratio as the average was very similar across all traffic group levels and station capacities. You will notice the variance does seem to be greater in the lowest traffic group and the smallest station capacities.





The in/out ratio was not affected by the high or low season either. A two sample t-test did not provide sufficient evidence to reject the null hypothesis that the means were the same in both seasons.

Insights gathered:

The summary of insights that were uncovered regarding Divvy stations and Divvy customer behavior are provided below:

- The busiest time for Divvy in terms of number of trips is June through September. Their slowest times fall in January - March, and November - December.
- The number of trips are higher in the PM hours regardless of season or day of the week.
- The highest trafficked hour of the day is at 5pm, and the weekday trips at that time are consistently higher than weekend trips at that time. This lends itself to the belief that this spike is attributed to 5pm being a popular commuting hour.

- The peak number of rides in the morning is at 8am, and also aligns with a popular commuting hour on the weekdays.
- The number of trips are higher on the weekend days in the busy season while the number of trips are higher on the weekdays in the slow season.
- Although the busiest day of the busy season lies on a Saturday, there is not a significant difference in the average ride count on a weekend day compared to a weekday during that time.
- The average trip duration is 15.92 minutes, and this tends to be higher on weekend days than weekdays throughout the year. The trip durations tend to be longer in the busy season as well.
- 60% of the top starting and ending stations in 2017 remained constant in both busy and slow seasons.
- 62% of Divvy stations have capacities between 10-15, but the average station capacity is found to be higher for higher trafficked stations.
- The ratio of trips in to trips out of the station is not significantly affected by the number of rides going through the station or the station capacity.

Predictive Modeling:

A lot of the insights gathered so far revolve around how Divvy's usage fluctuates throughout the week and throughout the year. As the usage differs in this way, I'd like to predict station stocking requirements by the day of the week and month of the year. For example, if Divvy wanted to know what the stocking requirements should be for station x on a Monday in August, the model would be able to provide the answer.

For this model I would like to use the ratio of number of trips in to number of trips out for each station on a given day as the target variable for the station's stocking requirements. I believe this will be more valuable because ratio can relate to the number of empty and filled slots each station should have. The total number of rides associated with the station alone on will not give us the same insight.

With the goal of a supervised model in mind, I began this portion of the analysis by organizing the trip data by station, day of the week, and month of the year so I could calculate the trips in to trips out ratio. The number of rides in and number of rides out were found for each station by day of the week and month of the year, but looking closer I found that not every day of the week was accounted for each month for every station. The reason behind this was that not every station had a ride come in or out on every day of the year. This trip information we'd like to have marked as 0 was not available in the current grouping. By unstacking the day of the week, and the month of the year to the top of the dataframe I was then able to then view the ride counts for every day of the week for every month. Those days without trip numbers in them showed up as

NaN, and I was able to fill them in with 0s, and restack the day of the week and month of the year variables.

The ratio could then be calculated using complete number of rides in and number of rides out information. A few things need to be addressed regarding this ratio however. When the rides out (the denominator) is 0 for this ratio, it produces a NaN value. To solve this, when rides out is 0, the ratio is set equal to the numerator (rides in). So if 12 rides came in and 0 rides went out the ratio would be 12, not 0 or undefined. This more accurately represents the reality of that particular day. The next instance of concern would be when the number of rides in for the day (the numerator) is 0. As 0 rides in and 1 ride out, and 0 rides in and 12 rides out would need to be accounted for differently we can't have them both with a ratio of 0. In these cases the ratio is set to 1/rides out, this will more accurately reflect the reality as well. The last potential concern would be days with 0 rides in and 0 rides out. This should not read as 0 either, as an even number of rides in and rides out needs to be represented by a 1. In these cases, the ratio was calculated as a 1.

After the ratios were calculated, the following variables were then added to the newly organized dataframe; average trip duration, station capacity, station latitude, station longitude, station's online date, and group number. The group numbers were assigned in previous analysis and are based on traffic to each station. 1 representing a high traffic station, and 4 representing a low traffic station.

Using longitude and latitude alone would not be a big help while building a model because each set of coordinates will be unique. Two stations right next to each other would not be seen as related by the model as the coordinates would differ. To better group the bike stations by location, I split stations into two different groups, north and south. These were split based on the median latitude. I also split the stations into four territories to further narrow down their location. These four territories are marked as 1 (northwest), 2 (northeast), 3 (southwest), and 4 (southeast). The north and south dividing line was again the median latitude, and the east to west dividing line was the median longitude for the stations.

The last thing I wanted to look at before feature selection was grouping the stations by their capacity. In previous analysis we saw that the mean ratio didn't differ greatly between station capacities, but the variance was higher in the lower capacity stations. Segmenting the stations will allow models to be created for each group that will ideally better fit each groups intricacies. Grouping the stations by capacities will also help in predicting the ratios for new stations. If we grouped stations by traffic level, for example, there wouldn't be any trip information for a brand new station to help determine what model should be used. The stations were segmented into 4 groups by capacity 1 containing the lowest capacity stations and 4 containing the highest. These designations were recorded in dataframe under the cap_group column.

Once this information was gathered together in the new data frame, I used Lasso regularization to provide some insight on my feature selection. Looking at the Lasso coefficients, none of the

variables had strong coefficients on their own.Because this was the case, all of the predictor variables (day of the week, month of the year, average duration, station capacity, group, year online, month online, section, and territory) would be used in the models to keep as much predictive power as possible. With a smaller number of variables we don't have to worry about the dimensions being too high in this case.

As we're looking to predict a continuous variable (the trips in to trips out ratio) I started with the linear regression model. Before training this model though, I needed to change the territory, section, group, day of the week, and month of the year variables to categorical data. Although the data are numbers, they represent different categories, and don't relate to each other by their size. Next, dummy columns were added to turn the categorical data into a series of 1's and 0's depending on what category they belonged to. This will allow us to use the categorical information in the regression model.

After splitting the data into training and testing sets, and training the linear regression model, it became clear that model was not performing well. After trying ridge regularization as well, I found a similar result of less than 1% accuracy. Looking back at scatter plots and correlation coefficients for these variables it was not surprising to find that they don't have a strong linear relationship with the calculated ratio.

To help solve the main goal of predicting stocking requirements for the bike stations I grouped the ratios into 5 different classes. These classes were based on the value of the ratio as the ratios near one another are going to have very similar stocking recommendations. For example whether the ratio is .8 or 1.2, you still have approximately the same amount of bikes coming in vs leaving a station on the particular day, and you would ideally have that station stocked with equal number of filled and empty slots. Once these stocking classes were created, this became a classification problem instead of regression.

Before working with different models, I wanted to have a benchmark to test them against. The benchmark used was the average ratio for each station for all of 2017. I wanted to test if these models would be more effective than putting them in a particular stock class based on this average. Looking at all of the stations together, the accuracy of our benchmark in predicting the correct stocking class was 68%. Separating the stations into the four groups by capacity mentioned previously, the accuracy of the benchmark was 61% for cap_group 1, 64% for cap_group 2, 75% for cap_group 3, and 81% for cap_group 4.

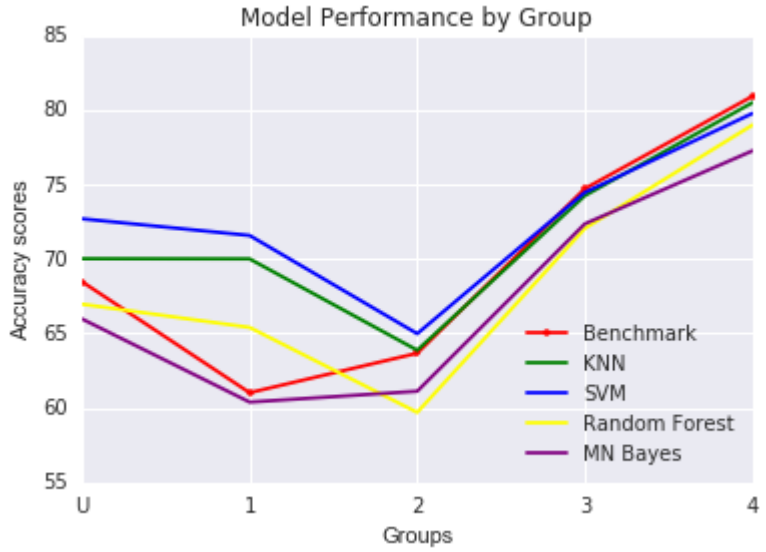
With these benchmarks in mind, I looked at 4 models while searching for the most accurate predictions. KNeighbors, Random Forest, SVM, and Multinomial Naive Bayes. Although often used with text data, the bayes model was included as it can be of help in finding a dependent categorical variable that has more than 2 possible categories.

Each of these models' hyperparameters were tuned using GridSearchCV. If the value on either end of the grid search was found to be the best parameter, another grid search was run to check the range above or below that value. Classification reports were then run on each model as well.

The accuracy results for these models can be found below:

	Not Grouped	Group 1	Group 2	Group 3	Group 4
Benchmark:	68.47	61.02	63.67	74.71	80.91
Models:					
SVM	72.69	71.57	64.98	74.46	79.76
KNN	70.01	70	63.87	74.24	80.48
Random Forest	66.96	65.4	59.7	72.04	78.98
Multinomial NB	65.97	60.39	61.12	72.35	77.25

SVM and KNeighbors performed the best across the board, and both beat the benchmark accuracy score of 68% for the ungrouped model. When the models were segmented based on on our 4 groups, they outperformed the benchmark scores for groups 1 and 2, and were in line with the benchmarks for groups 3 and 4.



Both training data scoring and test data scoring were looked at for each of the models to find if they were overfitting the training data. The Random Forest had the biggest differences between it's training and test scores. Even with the optimal parameter the training scores hovered around 98% but the testing scores did not come close. The training and testing scores were much more consistent in the other three models, and we don't have to worry about overfitting in this case with KNeighbors, SVM, and Multinomial NB.

Recommendations and continued improvements:

SVM was the highest performing model across the board, and although it was right in line with the benchmarks for capacity group 3 and capacity group 4, it did outperform the benchmarks for the first two groups. This aligns with our previous findings that the ratios were found to have a tighter shape around the mean for groups 3 and 4, while lower station capacities had a higher variance. This helps explain why the benchmark performed better in groups 3 and 4, and the models outperformed the benchmark by a greater degree in capacity group 1, the lowest capacity group. Although the benchmark accuracy score beats out the SVM score for group 4 by a slim margin, the precision, recall, and F1 scores computed for the SVM do outperform those of the benchmark. They were .7, .8, and .71 respectively compared to the benchmark scores of .67, .75, and .67. Based on the overall performance of the models, I found that SVM should be used in predicting the ratio for all groups moving forward.

To review, these methods focus on predicting a range of ratios for a station on any given day. These ranges were broken down so that all of the ratios in each range would warrant the same stocking suggestion. You'll find the 5 stock classes below along with their recommendations.

Ratio's and Recommendations:	
0.0 – 0.49	30% or more bikes docked than open slots
0.5 – 0.79	15% more bikes docked than open slots
0.8 – 1.19	Equal number of bikes docked and open slots
1.2 – 2.49	15% more open slots than bikes docked
2.5 +	30% or more open slots than bikes docked

The percentages were chosen to accommodate different station capacities while allowing each station to be prepared for excess rides in or out. The operations team at Divvy can quickly check what the current stocking status is for the station in question, and determine if it falls in line with the predicted stocking suggestion. If they find there is a difference for that particular day they will be able to make adjustments accordingly.

As predictions can always be further refined, a few other aspects can be considered for future improvement. We've seen that Divvy riders do use the service even in the heart of Chicago winter, but the number of trips is significantly lower during these inclimate times. Combining the trip data with weather data, and being able to classify the different days based on the weather conditions will allow us to gain an even better understanding in the fluctuations of usage throughout the year, and help better predict stocking requirements.

Another point to be considered is the number of times in a given day Divvy stations are able to be adjusted. If more resources are allocated to the operations team in the future, it would make sense to look at stocking requirements before the spike in traffic during the commuting hours in the morning and in the afternoon. There will be some cases where stations may need to be stocked differently than the ratio for the day as a whole. For example, many trips could leave a station downtown in a short amount of time to bring riders back home after work. When resources are available to do so, making adjustments before both of these commuting hours, can provide a better experience for Divvy's commuting customers.