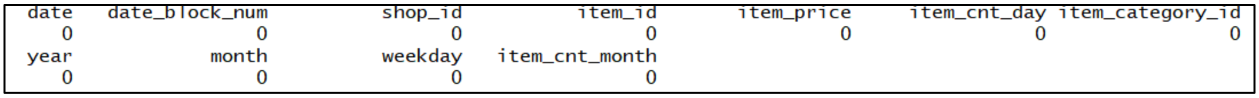
Midterm Project
Michael Pallante

The problem that will be explored in this project is the total sales for one of the largest Russian software firms, 1C Company. 1C Company has provided us with a complex data set in which we can see the total sales of the company for every item and every shop from January 2013 to October 2015. This data set contains 2,935,849 observations of sales data. There is also accompanying data for items sold, categories of items, and shop information. The significance of this immense amount of data is that we can use it to predict the future sales for this company based on a variety of factors. This type of information and these predictions can help the company down the line if sales trends are identified and ideas for improving sales, revenue, and profitability can be discovered. The objective for this project, a Kaggle competition titled "Predict Future Sales," is specifically to predict the sales for every product and every store for 1C Company in the next month, which is November 2015.

Before the modeling process was done, an exploratory data analysis was performed. After examining all pieces of data, we will be able to identify a modeling approach. For the purposes of this project and this exploratory data analysis, we are focusing on the sales_train dataset, which contains the referenced 2,935,849 observations mentioned previously, and the item data set. I merged the two datasets together into a dataset named salesdata, and removed any unneeded data sets. I also converted the date column from character format to date format, as well as created multiple date related columns including year, month, day, and weekday. Additionally, from examining the sample submission document for the competition, we notice that Kaggle wants us to predict the sales in terms of months. The variable they use is item_cnt_month, which is not currently existing in our dataset. That necessitates us to create this variable within our dataset. We are ready to continue proceeding through our exploratory data analysis after the addition of this variable to our sales dataset.

Now, we must check to see if this dataset contains any number of missing values that could potentially influence our models. As seen in Figure 1.1 below, we are fortunate to see that this dataset contains no missing observations and we can use the dataset as is.

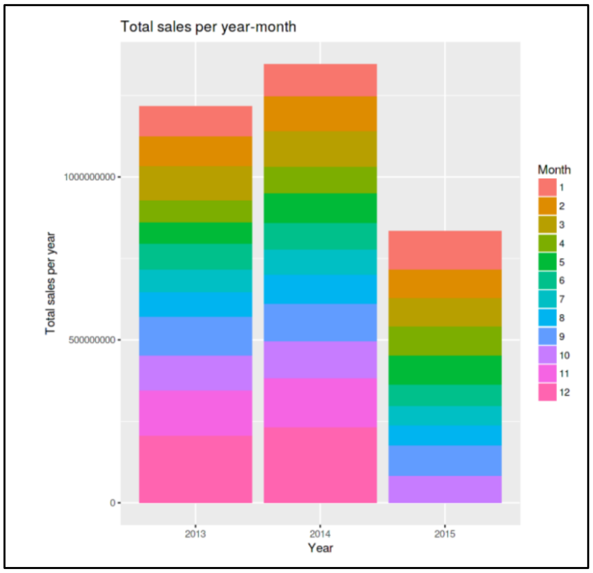
Figure 1.1: Missing Values Test



```
    date   date_block_num          shop_id          item_id       item_price     item_cnt_day item_category_id
       0                0                0                0                0                0                0
    year            month          weekday   item_cnt_month
       0                0                0                0
```

Next, I believed it was a strong idea to observe the relationships between total sales and time of sales. Perhaps we can get a picture of when items sell the most and if there is any trends. In Figure 1.2 below, the total sales per year by month is examined.

Figure 1.2: Total Sales Per Year, by Month



We can see from Figure 1.2 that sales appear to be stronger at the beginning of the year as opposed to the end of the year, with December as the exception, which could suggest sales go up for the holiday season. To get a more defined look at this sales information, we can look at Figure 1.3 below, which shows the total sales by month.

Figure 1.3: Total Sales by Month

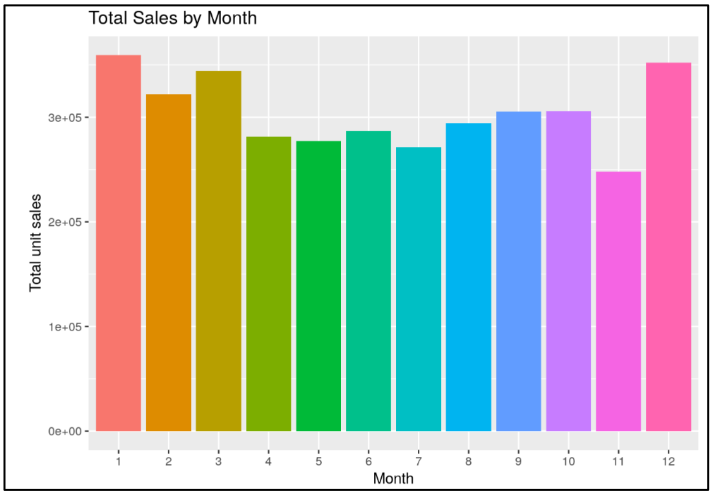


Figure 1.3 confirms what we gathered from Figure 1.2. However, it becomes more apparent that sales in November are the lowest among all months. Sales in the months prior to November are not all that great either. Sales in the few months after November appear to trend up. It is more obvious to us now why 1C Company would want us to predict future November sales, as they would like to see reasoning behind why their sales dip the most in that

specific month. Figure 1.4 below gives us another different look at the sales per month, showing the percent of items sold per month.

Figure 1.4: Percent of Items Sold Per Month

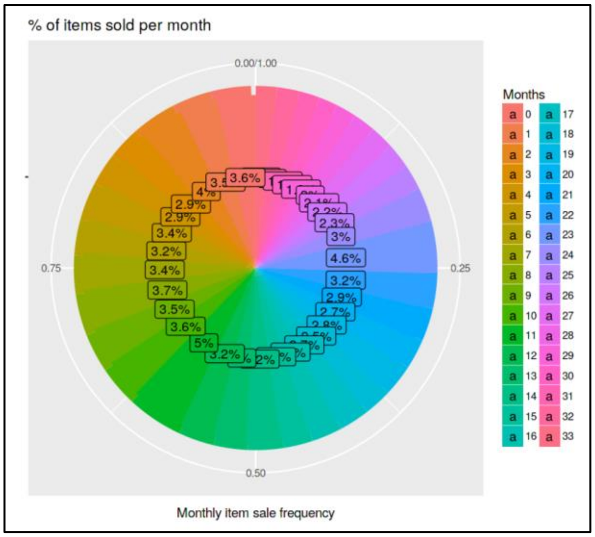
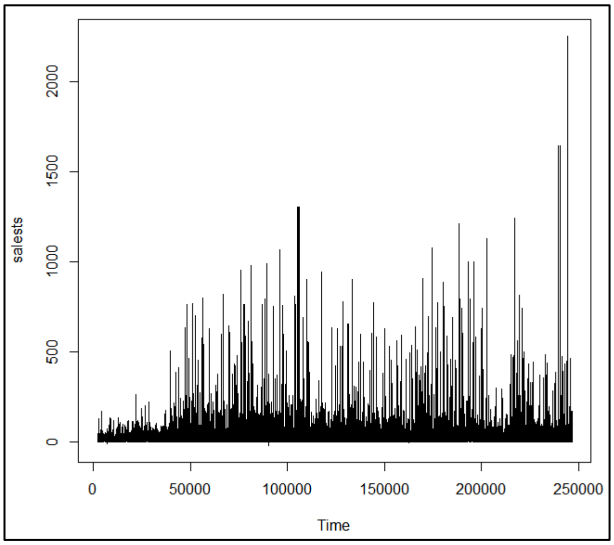


Figure 1.4 also confirms the beliefs we have from the previous two plots. A clearer picture of what to model has been derived from this overview of sales data based on timing of sales, and it is what our modeling process will focus mostly on. It has also been concluded that use two different time series modeling approaches to predict November 2015 sales, including ARIMA and ETS models. We will use these models to forecast those future sales. The salesdata dataset was created into a time series, and the plot of this time series can be seen in Figure 1.5 below. This plot does show seasonality with no dominating trends. This also marks the conclusion of our exploratory data analysis, and we can proceed to the model building phase.

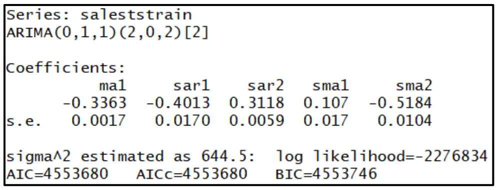
Figure 1.5: 1C Company Sales Time Series Plot



To improve model accuracy, the salesdata dataset was split into three different training sets. These three different training sets also focused on specific months in regards to sales. This was viewed as the best course of action with the salesdata dataset being so large, as well as us knowing information that may be best indicators of future sales. The first training set was built on sales data from the month of focus, November, and the prior month, October. The second training set was built on sales data from only the two prior months, September and October, and excluding November. The third training set was built on sales data from the month of focus, November, and the next month, December. The hope with splitting the training sets up this way was to see if the sales from the specific months selected were the best indicators of the sales in the month November. A total of six models will be built using these training sets, including three ARIMA models and three ETS models. So, essentially three of each model type are being created using each of the three respective training sets.

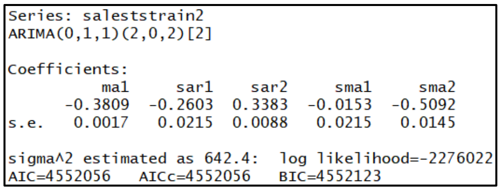
The first model type that was built was an ARIMA model. This was achieved in R using the auto.arima command, which automatically selects the ARIMA model. The ARIMA model that was selected for all three training sets was the ARIMA(0,1,1)(2,0,2)[2] model. The formulation of these models can be seen in Figure 2.1, Figure 2.2, and Figure 2.3 below. The forecasts for these models can be seen in Figure 2.4, Figure 2.5, and Figure 2.6 below.

Figure 2.1: ARIMA Model 1 Formulation



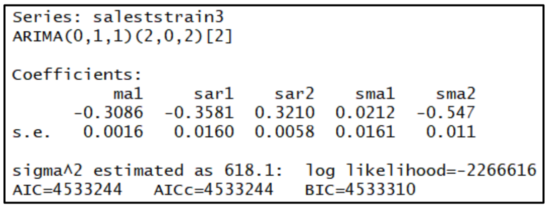
```
Series: salesstrain
ARIMA(0,1,1)(2,0,2)[2]

Coefficients:
          ma1     sar1    sar2   sma1     sma2
      -0.3363  -0.4013  0.3118  0.107  -0.5184
s.e.   0.0017   0.0170  0.0059  0.017   0.0104

sigma^2 estimated as 644.5:  log likelihood=-2276834
AIC=4553680   AICc=4553680   BIC=4553746
```

Figure 2.2: ARIMA Model 2 Formulation



```
Series: salesstrain2
ARIMA(0,1,1)(2,0,2)[2]

Coefficients:
          ma1     sar1    sar2     sma1     sma2
      -0.3809  -0.2603  0.3383  -0.0153  -0.5092
s.e.   0.0017   0.0215  0.0088   0.0215   0.0145

sigma^2 estimated as 642.4:  log likelihood=-2276022
AIC=4552056   AICc=4552056   BIC=4552123
```

Figure 2.3: ARIMA Model 3 Formulation



```
Series: salesstrain3
ARIMA(0,1,1)(2,0,2)[2]

Coefficients:
          ma1     sar1    sar2    sma1    sma2
      -0.3086  -0.3581  0.3210  0.0212  -0.547
s.e.   0.0016   0.0160  0.0058  0.0161   0.011

sigma^2 estimated as 618.1:  log likelihood=-2266616
AIC=4533244   AICc=4533244   BIC=4533310
```

Figure 2.4: ARIMA Model 1 Forecast

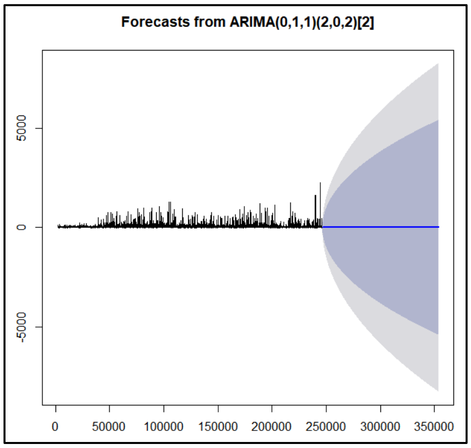


Figure 2.5: ARIMA Model 2 Forecast

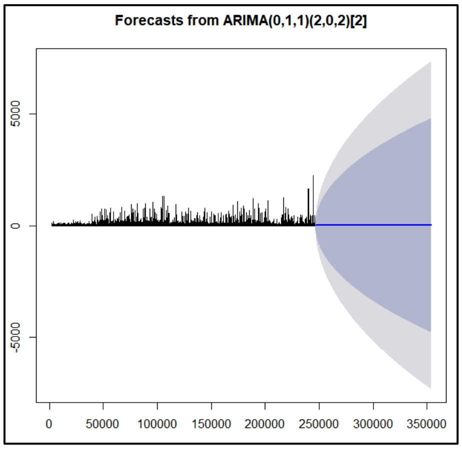
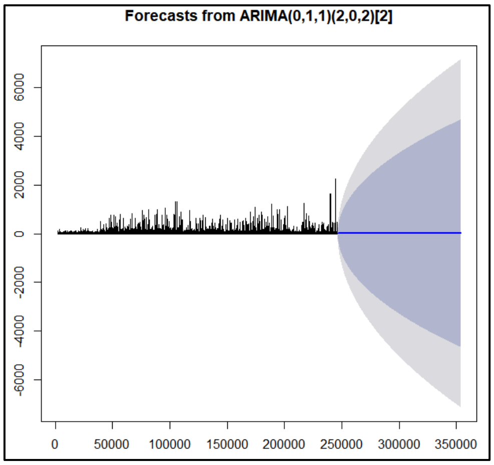
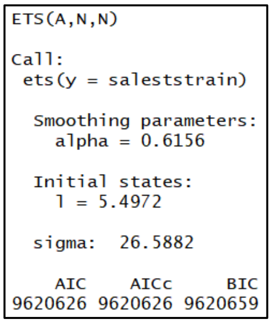


Figure 2.6: ARIMA Model 3 Forecast



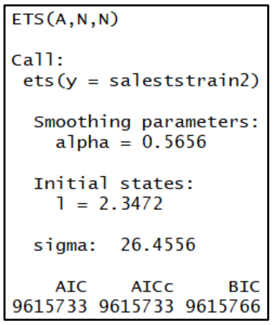
The second model type that was built was an ETS model. This was achieved in R using the ets command, which automatically selects the ETS model. The ETS model that was selected for all three training sets was the ETS(A,N,N) model. The formulation of these models can be seen in Figure 3.1, Figure 3.2, and Figure 3.3 below. The forecasts for these models can be seen in Figure 3.4, Figure 3.5, and Figure 3.6 below.

Figure 3.1: ETS Model 1 Formulation



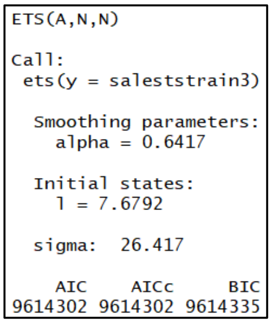
```
ETS(A,N,N)

Call:
 ets(y = salestrain)

  Smoothing parameters:
    alpha = 0.6156

  Initial states:
    l = 5.4972

  sigma:  26.5882

    AIC     AICc      BIC
9620626 9620626 9620659
```

Figure 3.2: ETS Model 2 Formulation



```
ETS(A,N,N)

Call:
 ets(y = salestrain2)

  Smoothing parameters:
    alpha = 0.5656

  Initial states:
    l = 2.3472

  sigma:  26.4556

    AIC     AICc      BIC
9615733 9615733 9615766
```

Figure 3.3: ETS Model 3 Formulation



```
ETS(A,N,N)

Call:
 ets(y = salestrain3)

  Smoothing parameters:
    alpha = 0.6417

  Initial states:
    l = 7.6792

  sigma:  26.417

    AIC     AICc      BIC
9614302 9614302 9614335
```

Figure 3.4: ETS Model 1 Forecast

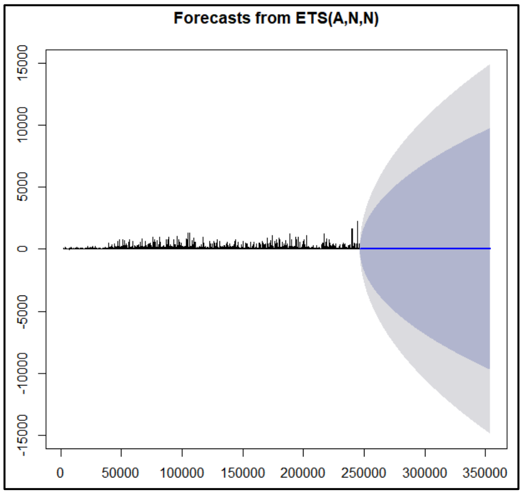


Figure 3.5: ETS Model 2 Forecast

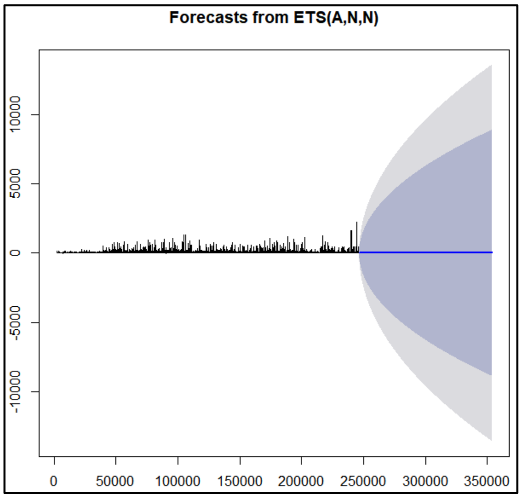
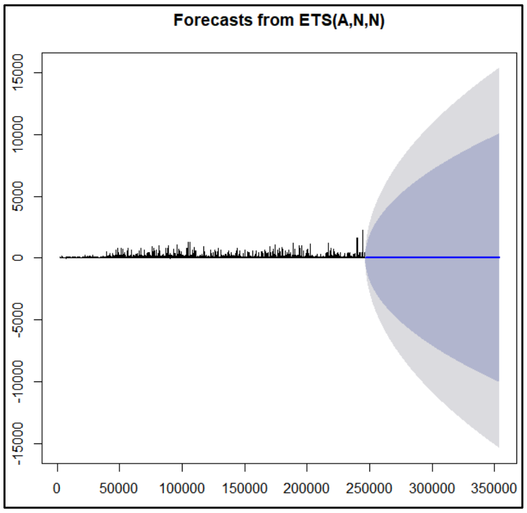


Figure 3.6: ETS Model 3 Forecast



After building both model types, the forecasts were measured against the test set for accuracy. The predictions for both model types were then prepared in the same format as the sample submission document. This allowed for six total submissions into the Kaggle competition: three ARIMA models and three ETS models. The performance of each model after submission, in terms of our RMSE evaluation metric, can be seen in Figure 4.1 below. The lower the RMSE, the better the model performed.

Figure 4.1: Model Accuracy and Performance, Post-Submission to Kaggle

Model	Root Mean Squared Error (RMSE)
ETS 3	1.45601
ETS	1.73076
ETS 2	2.75299
ARIMA	3.40179
ARIMA 3	4.50389
ARIMA 2	5.03274

The best model submitted to Kaggle for November 2015 sales predictions was the third ETS model I created, which had an RMSE of 1.45601. This model was the one that was fit using the training set that was built on sales data from the month of focus, November, and the next month, December. At the time of submission, this entry ranked 557 out of 732 competitors. The score proved to be a bit below average, with room for improvement. The models designed were likely limited in how they captured the sales and likely needed greater inclusion of higher correlated variables or another combination of months to predict sales. The Kaggle benchmark models outperforming all six time series models. Perhaps this suggests that creating time series models was not the best way to predict the total number of sales and that traditional modeling approaches may have fared better. In future work, I would have to reconsider using time series to make these predictions. I would also have to reconsider which months to focus on, if not including all months in future predictions. There were multiple variables with shared high correlations in each city. I learned a great deal from this assignment, however, as this was my first competition with Kaggle and also my first-ever analytics competition in general. I learned that traditional machine learning methodology can outperform time series methodology in forecasting. I also expanded my coding abilities considerably, as I was a basic R coder with no coding background prior to the MSDS program. This project was definitely the most difficult project I have ever completed, and without question the most challenging project I have completed at Northwestern University. The project felt very advanced and complex to me, but I managed to overcome it. The ability to even create submissions for this project gave me an incredible sense of accomplishment and I hope to continue to grow from it.

Literature

This section serves to briefly discuss peer reviewed journals that were used to inspire this report and relate to how this project was completed. The journal titled "A novel hybridization of artificial neural networks and ARIMA models for time series forecasting" discusses the popularity of the ARIMA approach in time series forecasting and presents advantages in linear modeling. These models identify existing linear structure in data to generate models with improved forecasting accuracy. The journal titled "Performance of state space and ARIMA models for consumer retail sales forecasting" shows the importance of forecasting in strategic and planning decision that go into retail sales companies. Some of the forecasts in this case study were derived from ARIMA models and were evaluated using RMSE just like this project. The journal titled "Time Series forecasts of international travel demand for Australia" proposes the difference in forecasts using smaller and larger time periods, and proved that the model using a smaller time period performed very well and was more accurate. This model was developed using an ARIMA model and RMSE was the evaluation metric as well. This helped me to decide on using a smaller number of months for my forecasts. The inclusion of more months were less accurate when I tried doing it that way. The journal titled "Comparison of the ARMA, ARIMA, and the autoregressive artificial neural network models in forecasting the monthly inflow of Dez dam reservoir" showed that an increased number of parameters increased the forecast accuracy and they also incorporated an ARIMA model. This explains the reason for developing multiple training sets with different parameters in each one, which helped me to find the model that performed best in my analysis. The journal titled "Time series sales forecasting for short shelf-life food products based on artificial neural networks and evolutionary computing" stresses the great importance of accurate time series sales forecasting. These forecasts accompany company efforts to increase profits, reduce costs improve customer service, and become more efficient. These reasons can set companies apart from strong competition and explains why 1C Company wants to find out how to improve sales in the month of November, which would go a long way in striving for long-term success.

References

Doganis, P., Alexandridis, A., Patrinos, P., & Sarimveis, H. (2006). Time series sales forecasting for short shelf-life food products based on artificial neural networks and evolutionary computing. *Journal of Food Engineering, 75*(2), 196-204. Retrieved May 3, 2018, from https://www.sciencedirect.com/science/article/pii/S0260877405002402

Khashei, M., & Bijari, M. (2011). A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. *Applied Soft Computing, 11*(2), 2664-2675. Retrieved May 3, 2018, from https://www.sciencedirect.com/science/article/pii/S1568494610002759

Lim, C., & McAleer, M. (2002). Time series forecasts of international travel demand for Australia. *Tourism Management, 23*(4), 389-396. Retrieved May 3, 2018, from https://www.sciencedirect.com/science/article/pii/S026151770100098X

Ramos, P., Santos, N., & Rebelo, R. (2015). Performance of state space and ARIMA models for consumer retail sales forecasting. *Robotics and Computer-Integrated Manufacturing, 34*, 151-163. Retrieved May 3, 2018, from https://www.sciencedirect.com/science/article/pii/S0736584515000137

Valipour, M., Banihabib, M., & Behbahani, S. (2013). Comparison of the ARMA, ARIMA, and the autoregressive artificial neural network models in forecasting the monthly inflow of Dez dam reservoir. *Journal of Hydrology, 476*, 433-441. Retrieved May 3, 2018, from https://www.sciencedirect.com/science/article/pii/S002216941200981X