Colin Michael

DSC 630-T301 Predictive Analytics

Course Project: Milestone 3

I have confirmed that I will be doing my project on gasoline prices in the United States from 1995 to 2021. I gathered this dataset from Kaggle.com. It includes weekly gas prices for 13 different types of data.

I am fascinated by gas prices because of how greatly they impact America. Gas prices are often the subject of negative medium coverage for how rapidly they rise. Millions of Americans are forced to pay rising gas prices, which can place an extreme financial burden on people. Gas prices also impact shipping costs for businesses. This can cause significant increases in the total prices of goods.

My dataset from Kaggle is in csv format. I will import this data into Jupyter with the pandas.read\_csv function. The dataset includes 1,361 rows of data, one for each week from 1995 to 2021. Each row features 13 columns of gas prices. The data columns are gas prices for: all grades all formulations, all grades conventional, all grades reformulated, regular all formulations, regular conventional, regular reformulated, midgrade all formulations, midgrade conventional, and midgrade reformulated. This provides a comprehensive picture of average gas prices across different types.

Evaluating your data and determining your goal is the first step towards selecting a data science model type. My data set is numerical data over a period. My main goal is to use that data to predict future values. My first idea is to do a time series model. A time series is a series of data points ordered in time. In time series models, time is usually the independent variable, and the goal is to forecast the future (BuiltIn.com).

There are a couple of key factors to consider when running a time series model. The first is seasonality, which refers to periodic fluctuations in data. It will be interesting to see if gas prices are impacted by seasonality. My data set should do a nice job at revealing these trends.

Next is stationarity, which refers to a time series where statistical properties do not change over time. A data set is stationary if the mean and variance are constant (BuiltIn.com). Most time series models feature stationary data. I will perform a Dickey-Fuller statistical test on my data to determine if the time series is stationary.

I will begin my time series model by using an exponential smoothing test. This model calculates a moving average with the data and gives more importance to more recent data compared to older values. I believe that with inflation I will need to use this type of model, instead of a simple moving average model. I would also like to make a seasonal autoregressive integrated moving average model (SARIMA) time series. SARIMA applies seasonality to the following three components: autoregressive (a model that uses the dependent relationship between an observation and several lagged observations), integrated (the use of differencing raw observations to make a time series stationary), and moving average (using dependency between an observation and a residual error from a moving average model) (WisdomGeek.com).

I hope to use these models to better understand gas price fluctuations and predict future values. I believe that the SARIMA model will be more accurate than the simple exponential smoothing model because it considers seasonality. I believe there will be significant trends in data over the different seasons due to changes in demand over the seasons. For example, people typically travel more around the holiday season, so I’d expect gas prices to see peaks around the December-January months.

Using data ethically is arguably the most important part of data science. It is crucial that we as data scientists do not use data in harmful ways. One potential concern with my data model would be that it would reveal trends that would allow data companies to gouge their prices. Price gouging is when retailers rapidly increase the prices of goods. In May of 2022, the United States House of Representatives passed a bill to allow the Federal Trade Commission to investigate energy companies for price gouging (CNN.com). Price gouging inflicts great harm to consumers, and I will be careful to not structure my data model to encourage such malicious behavior.

My contingency plan if the exponential smoothing and SARIMA models does not work out would be to run a Prophet time series model. Prophet is an open-source software created by Facebook’s data science team and is available on R and Python (Facebook.com). It works extremely well with weekly data and is effective at revealing seasonal effects in data. The main appeal to Prophet is that it is considered easier to use and allows more clarity into seasonality in data.

I am very excited to dive into these models and the data. I have never run a time series model independently, so this should be a great learning experience. There are a lot of resources online and samples of code on time series models. I believe these will be a great resource for me as I run the exponential smoothing, SARIMA, and potentially the Prophet time series models.

Milestone 3: Preliminary Analysis:

I began my preliminary analysis by importing my Excel csv data into Python through the pandas package. I initially downloaded the csv from Kaggle.com. Once I loaded my csv data into a pandas data frame, I ran the function “df.shape” to get an idea of how my data frame is structured. It has 1,361 rows and 14 columns, which is expected. Next, I ran the function “df.isnull().values.any()” to identify any null values in my data frame. Luckily, there are no null values present. Last, I wanted to verify that my data set has no duplicate date values, because I am trying to analyze gas prices over time. I ran the function “df[‘Date’].is\_unique” and returned the value “True”, which verifies that there are no duplicate date values in my data frame. I feel comfortable using my data set after conducting these high-level data checks.

I next wanted to create some data visualizations to help better understand my data. I am aiming to see trends over time, so I began by creating a line chart with different gas prices: *df.plot(x = ‘Date’, ‘y = [‘A1’, ‘R1’, ‘M1’, ‘P1]), plt.title(‘Gas Prices’), plt.xlabel(‘Years’), plt.legend()*

*Chart

Description automatically generated*

The plot shows the weekly average prices of all grades, regular grade, midgrade, and premium gasoline from 1995 to 2021. All four gasoline prices follow the same trends: major dip from 2001 to 2002, steady growth from 2003 to 2008, massive drop from 2008 to 2009, prices rebounding from 2009 to 2011, high fluctuations from 2011 to 2014, another massive drop from 2014 to 2015, steady growth from 2015 to 2019, and a slight from 2019 to 2020.

I then wanted to get a better understanding of the volatility of gas prices in each given year from 1995 to 2021. I created a box plot of the “All Grades All Formulations Gasoline Prices” (A1) against each year of the data set: *df=pd.DataFrame(np.random.randint(50,1000,1361).reshape(-1,1),*

*index=pd.date\_range('1995-01-01','2021-01-24',freq='W'),*

*columns=['A1'])*

*df.reset\_index(inplace=True)*

*df.columns = ['Date','A1']*

*df['year'] = df['Date'].dt.strftime('%y'), fig, ax = plt.subplots()*

*fig.set\_size\_inches((12,4))*

*sns.boxplot(x='year',y='A1',data=df,ax=ax)*

*plt.show():*

Chart

Description automatically generated

The years from 2006 to 2013 had high volatility within the years, but the periods of 1995-1998 and 2020-21 had low volatility.

This preliminary data analysis reinforces my confidence in the focus of my research project. First, I feel very confident in the quality of the data I am analyzing. There are no missing weeks of data, no missing data in my set, and it is all formatted consistently. The line charts and box plot I created identify some very interesting trends. There are some very sharp changes in gas prices year to year, reflected by the line charts. Also, there is significant volatility of gas prices within the years studied, illustrated by the box plot. I am highly fascinated to gain more insight into forecasting future trends in the data by conducting moving average tests, seasonal autoregressive integrated moving average model (SAMIRA), and Prophet time series tests. This will allow me to predict future trends in gas prices, as opposed to just illustrating existing trends which I have done with the line and box plots.

**Milestone 4**:

My starting dataset has 1,361 rows and 14 columns. The first column is ‘Date’, with one row of data for each week from 1/02/1995 to 1/25/2021. Each column after gives a different metric for gas prices. I decided to focus my first time series model on one metric, so I dropped every column except ‘Date’ and ‘A1’ (A1 measures all grades all formulations retail gasoline prices). I chose to focus on ‘A1’ because it provides the best summary of different types of gasoline. I then converted my ‘Date’ column to ‘datetime’ format and set the data frame index to be ‘Date.’ I did these steps to make the data frame more compatible with the time series coding: *df = pd.read\_csv("/Users/colinmichael/Desktop/Data Science/DSC 630/usgas.csv")*

*df['Date']= pd.to\_datetime(df['Date'])*

*df = df[['Date', 'A1']]*

*df.set\_index("Date", inplace=True).*

I chose the SARIMAX time series model to predict future gas prices. The SARIMAX model stands for ‘Seasonal Autoregressive Integrated Moving Average and takes seasonality into account when predicting future values. I believe this will be useful towards predicting gas prices because seasonality impacts demand for gasoline. I ran the following code to fit my data frame into a SARIMAX model and produces a data summary: *model = sm.tsa.statespace.SARIMAX(df["A1"], order=(1, 1, 1), seasonal\_order=(0, 1, 1, 52)) results = model.fit() print(results.summary())*

The results.summary() function showed that my analysis is statistically significant, with p-values under .05. I then wanted to use the model to predict gas prices for the next year. I ran the following code to do this: *n = 52, pred = results.get\_prediction(start=df.index[-1], end=df.index[-1] + pd.DateOffset(weeks=n), dynamic=False), pred\_mean = pred.predicted\_mean, pred\_conf = pred.conf\_int().*

Next, I plotted the predicted values against the existing data from 1995-2021: *plt.plot(df.index, df["A1"], color='blue', label='Observed')*

*plt.plot(pred\_mean.index, pred\_mean, color='red', label='Predicted')*

*plt.legend()*

*plt.xlabel('Date')*

*plt.ylabel('Price')*

*plt.show()*

Chart, histogram

Description automatically generated

My results show that gas prices are predicted to increase throughout the early half of 2021, and then start to decline at the end of 2021 into 2022. This visualization helps show how the SARIMAX’s predictions follow the general trend of the data. The next steps in my project will be conducting different types of time series models and comparing results. I also am planning on using a subset of the prices, for example 1995-2019, to train each model, and will compare the predicted results against the actual results. This will give me a better idea of which models more accurately forecast gas prices.