Final Project: Residential Power Usage in Houston, Tx

Emily Sutton, Luc Ginestet-Araki, Mathew Chan, Wyatt Garrett

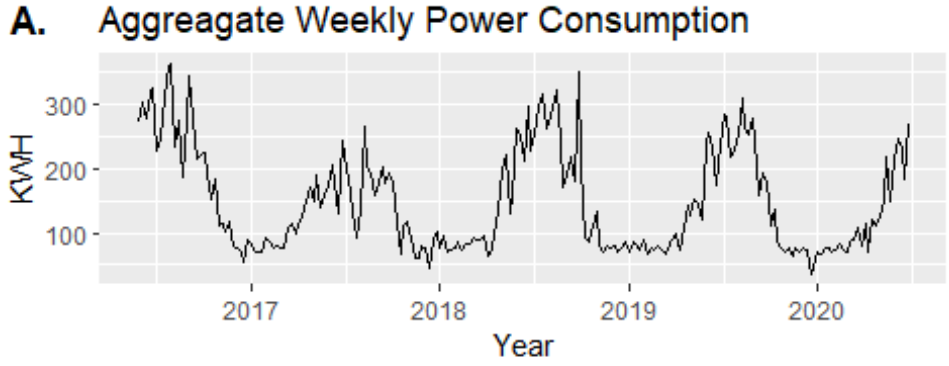
Introduction

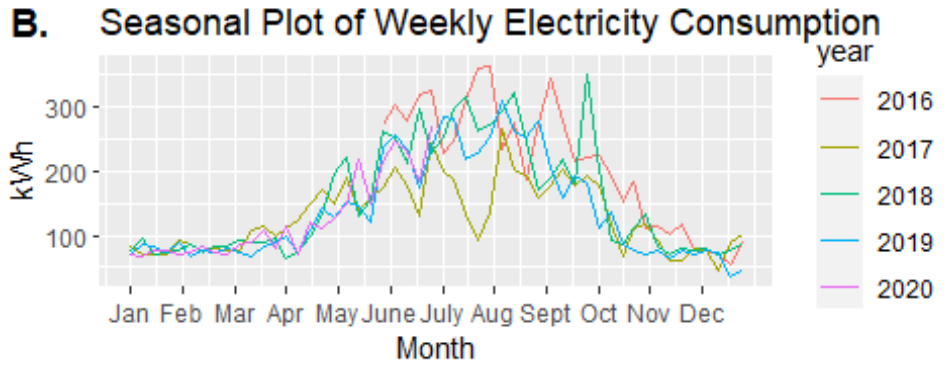
This project studies residential power usage in Houston Texas, analyzing time series data between June 2016 and August 2020. This report is tailored towards power grid suppliers where forecasts of power usage can allow suppliers to anticipate and meet consumer demand and properly apply surge pricing.

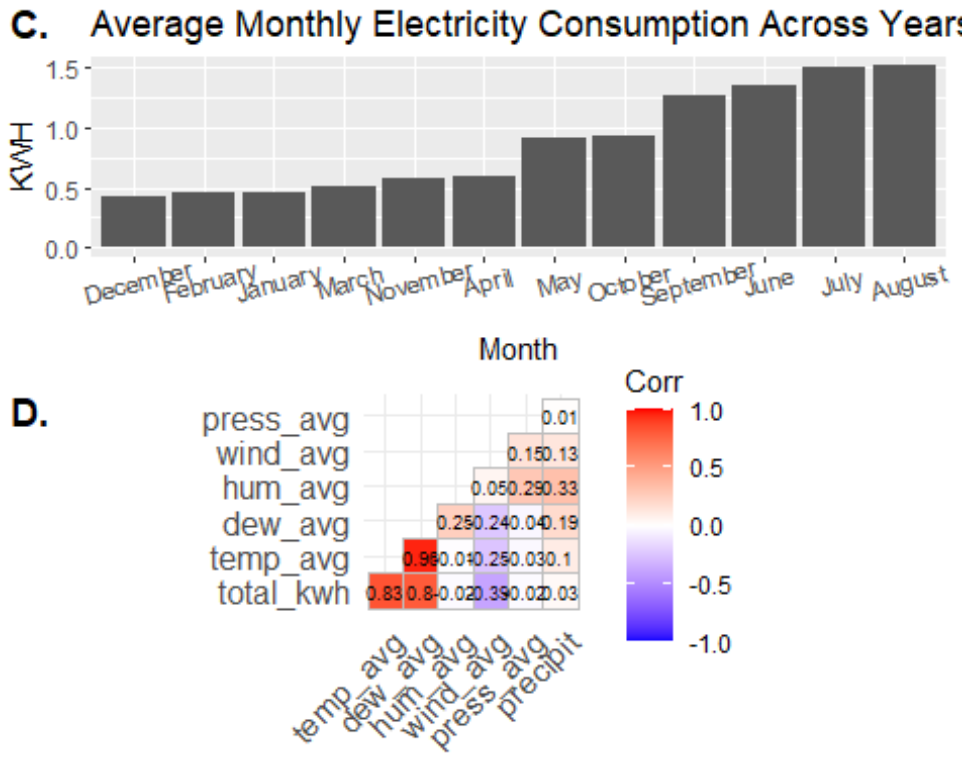
We choose this topic because of the blackouts that Texas experienced due to improper energy grid management. We want to build a forecasting model to help power suppliers anticipate power usage and adapt for the future.

Data Analysis

Our power data came from TRIEAGLE ENERGY LP, The Woodlands, Texas 77393. The historical weather data for Houston, Texas was extracted from "www.wunderground.com" The data required extensive cleaning. The dates in the two data sets would flip formatting on the 13th of the month, which required complicated cleaning.







A. The data shows strong seasonality. From this graph it appears there may downward trend to the data.

B. There appears to be strong seasonality across years.

C. August is the month with the highest power consumption followed closely by the other summer months. Early fall (September) power usage was lower than late spring power usage (May).

D. Power consumption is most strongly correlated with temperature, which is logical. Dew and power consumption are strongly correlated as well which is surprising. This may be due to dew's very strong correlation with temperature.

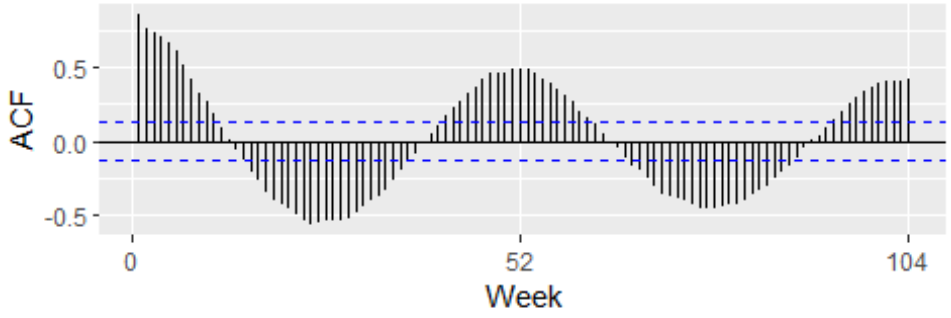
Data Manipulation

We found that working with daily data presented a host of problems when creating models. These included ARIMA or ETS models not supporting the lags necessary for the data, and other complications. As a result we choose to use an aggregate of weekly power consumption. We use this weekly data set for the rest of our data analysis and the rest of our forecasting models and will refer to it as the "time series" or "the data".

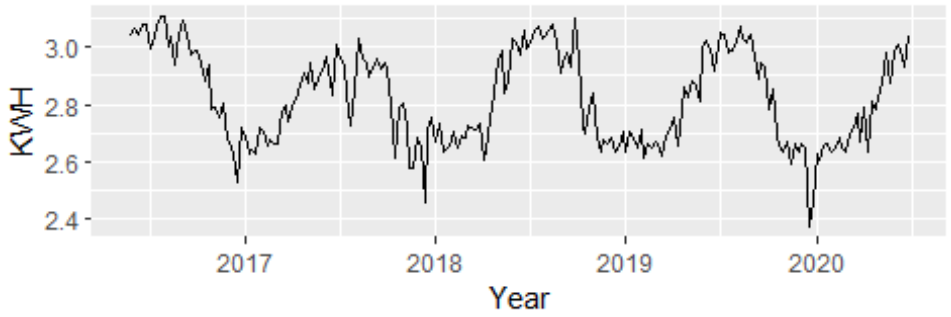
```
## Time Series:
## Start = c(2016, 22)
## End = c(2016, 27)
## Frequency = 52
##      total_kwh
## [1,]   273.316
```

```
## [2,]   304.302
## [3,]   277.739
## [4,]   319.443
## [5,]   324.557
## [6,]   229.126
```

E. Autocorrelations of time series by lag



F. Box-Cox Transformed Time Series



Autocorrelation between days of the time series:

E. Lag one was the highest of any of the lags and had the highest autocorrelation of 0.845. The data shows lags with significant differences from zero, which means that past data can have significant prediction power on the future. The ACF plot exhibits clear seasonality and because the autocorrelations are higher at smaller lags the data may exhibit some trend (it appears to decay to zero very slowly). The data appears non-stationary.

Exploring transforming the data:

We could use a lambda of -0.2466745 in a Box-Cox transformation to stabilize the seasonal variation. Since the variance in the seasonality of the original data is decreasing over time the negative sign of the calculated lambda makes sense. However we won't initially transform the data. For the basic naive and seasonal forecasts we will explore the data without transformation. In our ETS function the multiplicative damping may solve this issue.

F. The transformed data appears to have constant variation across time.

Should we use differencing?

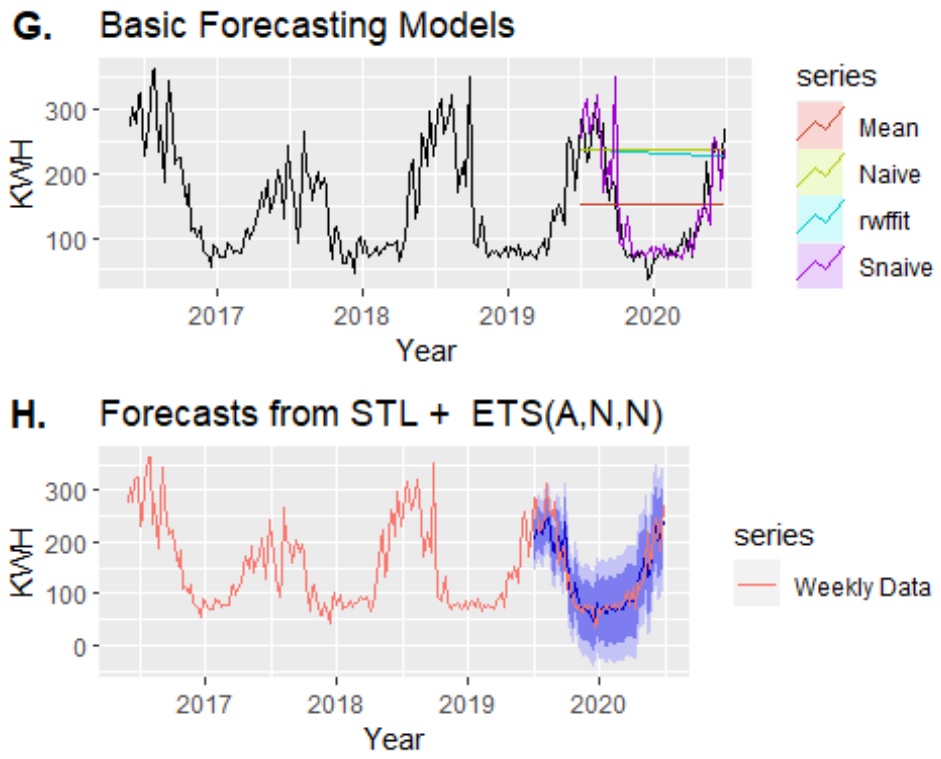
Using a Kwiatkowski_Phillips-Schmidt-Shin test yields a test statistic of 0.2635 This test statistic is lower than the five percent critical value of 0.463 so we accept (can't reject) the null that the data is stationary. Using an augmented Dickey-Fuller test we arrive at -3.8832 which is lower than the test statistic of 0.01 and we accept (can't reject) the null hypothesis that the time series is a random walk. The differences in these two tests is likely due to the Dickey-Fuller test being a unit root test and the KPSS test being a stationary test, and likely means that the data does not give enough observations. Using nsdiffs we found that differencing was necessary to create stationary data.

From our data analysis we expect that models that incorporate seasonality and stabilize for variation in seasonality will perform best. We also expect a model with zero or one differences to perform well.

Forecasting Model Selection

Basic Forecasting Models:

We first applied basic forecasting models, comparing mean, naive, drift and seasonal naive methods. The Seasonal Naive Model had the highest accuracy in the test set. This is expected due to the very apparent seasonality of the data. The graph of the four models are seen in Exhibit G.



G. Of the basic models the seasonal naive method clearly fits the model the best. However there was significant information left in the residuals, so we continued exploring for a better model.

ETS models:

Exploring ETS models with the data was impossible due to the high frequency of the data (52). As a result we chose to use a Seasonal and Trend decomposition using Loess + ETS model.

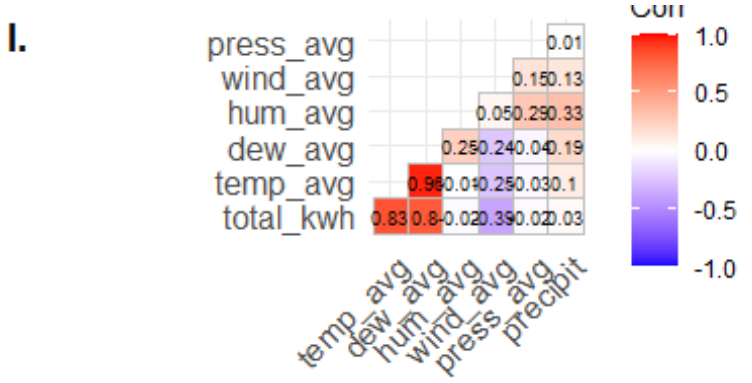
H. The blue STL forecast appears to fit the model fairly well. The STL + ETS model had a lower RMSE than the seasonal naive method (as well as a lower MAE, MAPE, and MASE). While the STL + ETS model was more accurate than the seasonal naive method, the model still did not account for all of the information available . The STL + ETS model failed the Ljung-Box test which means that the data exhibits serial correlation in the residuals that has not been included in the model. Our search for a model continued.

Regression Models:

We began our survey of regression models by exploring the correlations between the variables seen in Exhibit I (and earlier in Exhibit B). We noted that average dew, humidity and pressure were all strongly correlated with temperature. Including all these variables into a model could lead to imperfect multicolinearity.

We first used a backwards step-wise regression model on all available variables. With the inclusion of of date and dummy variables for month the backwards step-wise regression was still outperformed by a time series regression model.

We found that a parsimonious time series regression model that included the the average temperature for the week, the average pressure, and the average precipitation was the mot accurate model that avoided multicolinearity. Including any other variable increased the standard error of all regression terms.



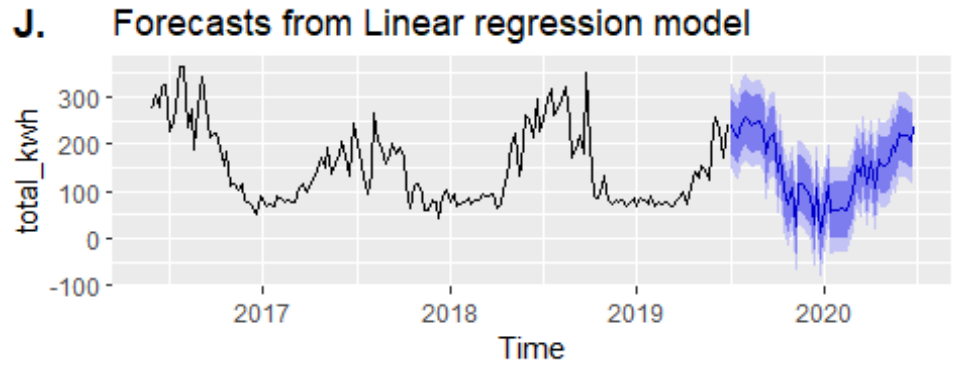
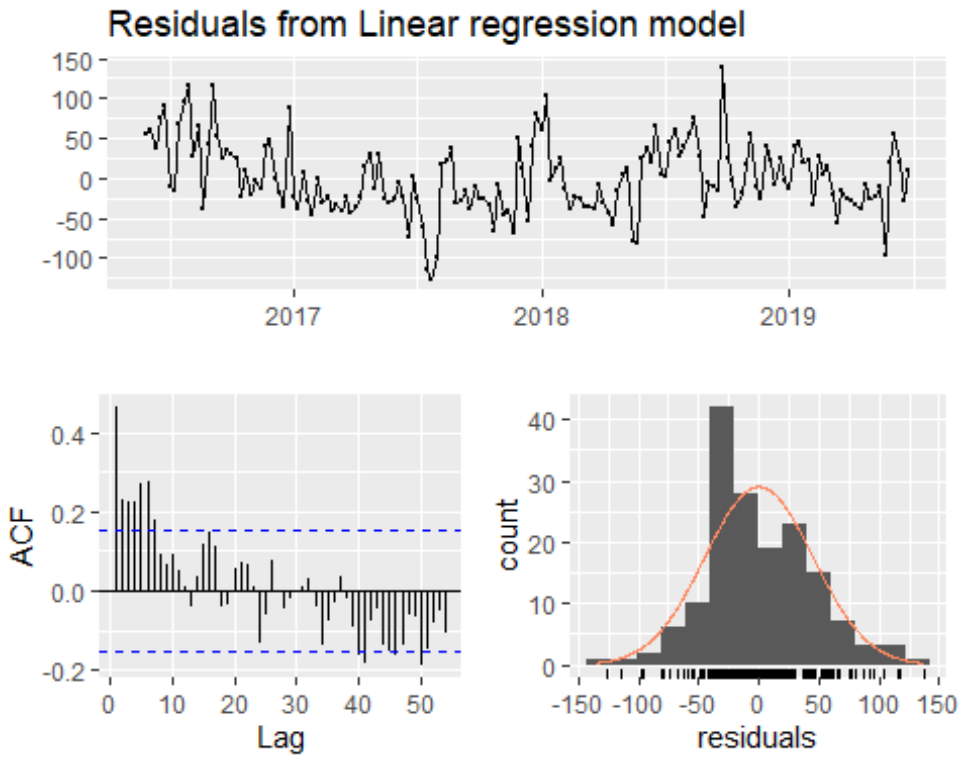


Exhibit J. The regression model appears to model seasonality well and but only explains roughly 70% of the variation in the data.

```
##
## Call:
## tslm(formula = total_kwh ~ temp_avg + wind_avg + precipit, data =
training_ts)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -125.902  -29.030   -8.045   27.146  138.028
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -183.8405    33.1368  -5.548 1.20e-07 ***
## temp_avg       0.8049     0.0489  16.460  < 2e-16 ***
## wind_avg      -1.1993     0.2994  -4.006 9.51e-05 ***
## precipit      -0.4475     1.7405  -0.257    0.797
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 45.31 on 157 degrees of freedom
## Multiple R-squared:  0.6971, Adjusted R-squared:  0.6913
## F-statistic: 120.5 on 3 and 157 DF,  p-value: < 2.2e-16
```



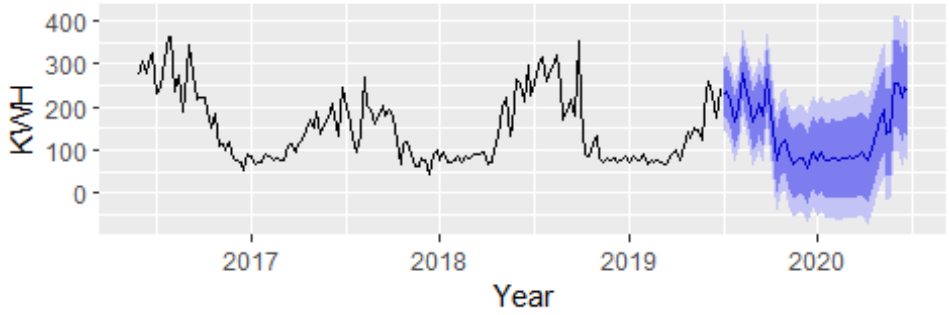

```
##
##  Breusch-Godfrey test for serial correlation of order up to 32
##
## data:  Residuals from Linear regression model
## LM test = 61.442, df = 32, p-value = 0.001327
```

The regression model still did not account for all of the available information. There appear to be many spikes in the residuals from the linear model and the ACF graph shows greater than 5% of the lags are significantly autocorrelated. The model's failure of the Breusch-Godfrey test confirms that information remains in the residuals. We continued our search.

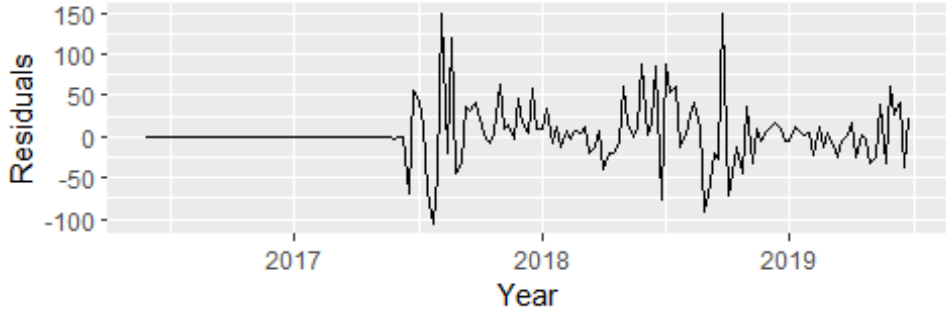
ARIMA Models:

We began by finding an ARIMA model using auto.arima.

K. Forecast from ARIMA (2,1,1)(1,1,0)[52]



L. Residuals from ARIMA(2,1,1)(1,1,0)[52]



K. Using the auto.arima function we found that a SARIMA model describing power consumption combined a first order Auto-Regressive model with one degree of first differencing and a 1st order Moving Average model. In addition the model has a seasonal first order auto-regressive component and one order of seasonal differencing with a 52 week cycle.

We suspected from our earlier exploration of stationarity that first differencing would be required. We also were not surprised to see a seasonal difference that included an annual cycle.

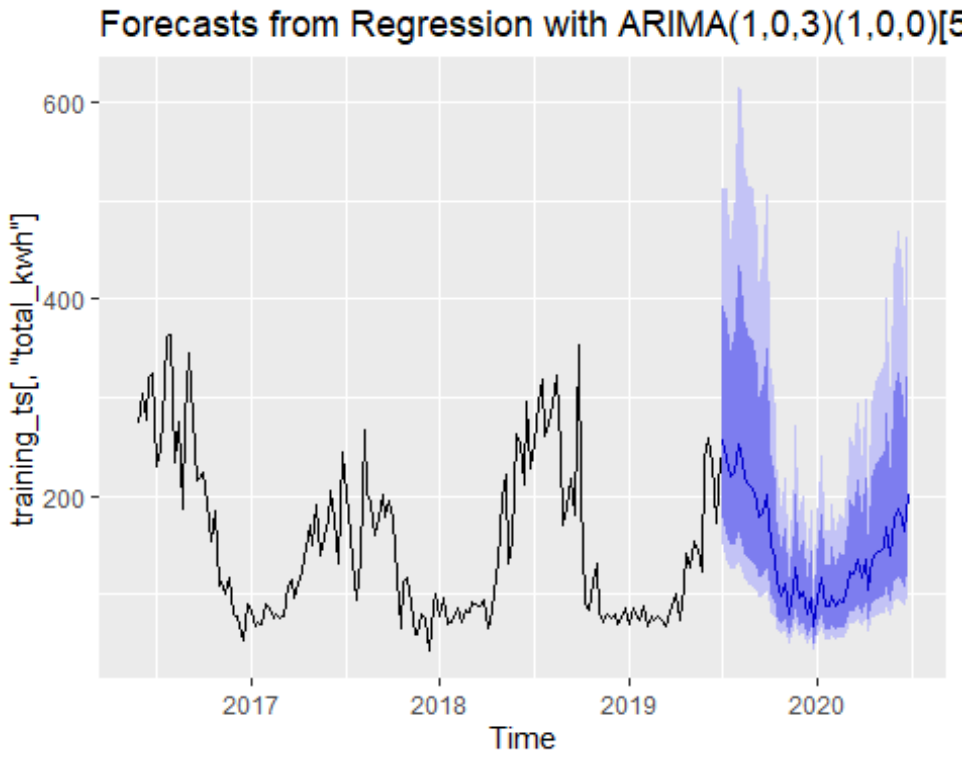
L. The residuals of the SARIMA model were not calculated for the first 52 time stamps used in the seasonal differencing. The model showed a spike in the 63 week, and again in the 122 second week for two August residual spikes.

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(2,1,1)(1,1,0)[52]
## Q* = 52.897, df = 28, p-value = 0.003018
##
## Model df: 4.   Total lags used: 32
```

The SARIMA model narrowly failed the Ljung-Box test, meaning that there agian is information left in the data not captured by the model.

Dynamic Regression Models:

We began our exploration of dynamic regression models with a model that combined an ARIMA with our earlier regression model.



The dynamic regression model produced a regression and SARIMA model that combined a first order Auto-Regressive model with a 3rd order Moving Average model. In addition the model has a seasonal first order auto-regressive component.

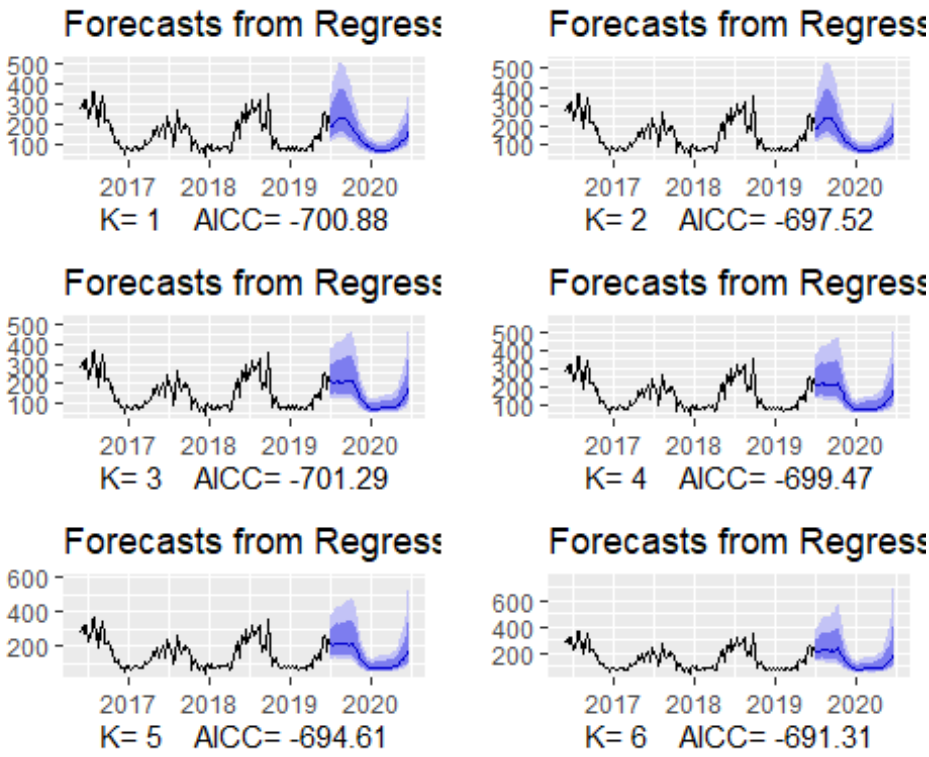
```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(1,0,3)(1,0,0)[52] errors
## Q* = 45.986, df = 23, p-value = 0.003003
##
## Model df: 9.   Total lags used: 32
```

The model again failed the Ljung box test indicating that there is information available in the data not being captured by the model.

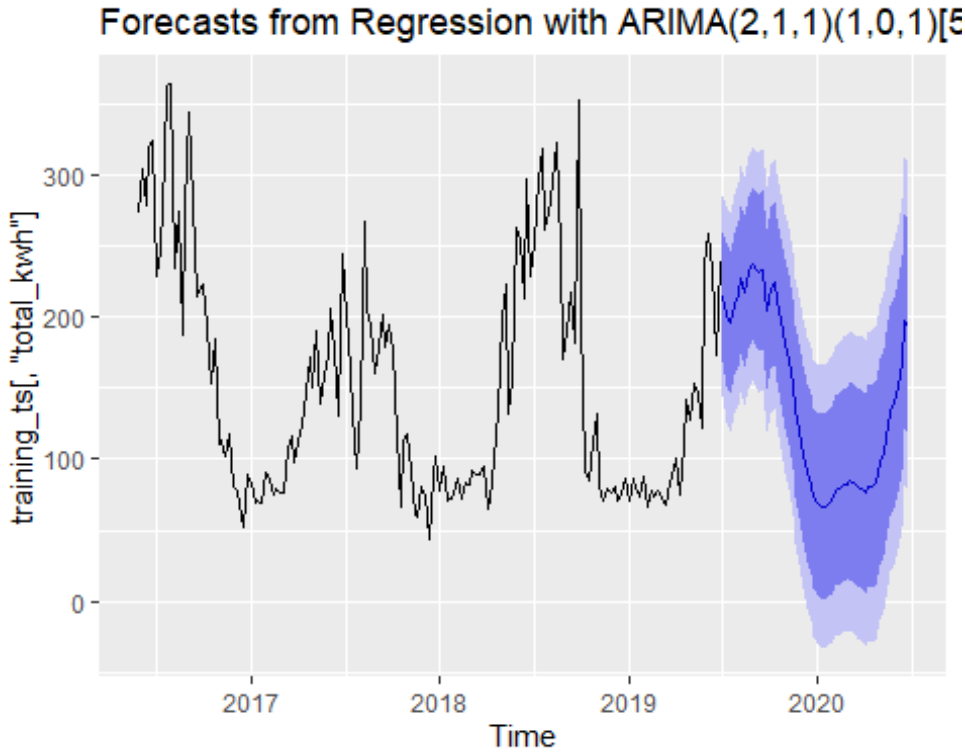
Dynamic Harmonic Regression Model:

We then moved on to our final model a dynamic harmonic regression model that combined regression and SARIMA components.

“When there are long seasonal periods, a dynamic regression with Fourier terms is often better than other models we have considered in this book.” - Robert Hyndman



The dynamic regression model produced six forecasts with different K, the number of Fourier sin and cos pairs. We selected the model with K equal to 6 because it had the lowest AICC and the best accuracy.



The dynamic harmonic regression combined a regression and SARIMA model describing power consumption. The SARIMA component combined a 2nd order Auto-Regressive model with one degree of first differencing and a 1st order Moving Average model. In addition the model has a seasonal first order auto-regressive component and 1st order seasonal moving average with a 52 week cycle.


```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(2,1,1)(1,0,1)[52] errors
## Q* = 30.502, df = 21, p-value = 0.08236
##
## Model df: 11.   Total lags used: 32
```

Excitingly, the dynamic harmonic regression model passed the Ljung-Box test. The model residuals were normally distributed and the ACF plot showed no significant serial correlation between lags. The plot of the residuals appears to have zero mean.

Let's now compare the dynamic harmonic regression model to all the models.

Comparing All Models:

```
## [1] "STL + ETS model accuracy"

##                      ME     RMSE      MAE         MPE     MAPE      MASE
## Training set -1.834437 30.33457 20.92964 -3.9275358 13.98242 0.4661446
## Test set      6.772887 34.99857 25.29269 -0.6277598 19.31879 0.5633186
```

```
##                    ACF1 Theil's U
## Training set 0.1964693        NA
## Test set     0.2104836  1.017338

## [1] " "

## [1] " "

## [1] "Regression model accuracy"

##                         ME     RMSE      MAE        MPE     MAPE      MASE
## Training set  1.804705e-15 44.74702 35.68175  -7.070319 28.92008 0.7947035
## Test set     -9.715182e+00 34.82503 30.12562 -14.463301 31.05722 0.6709575
##                    ACF1 Theil's U
## Training set 0.4631853        NA
## Test set     0.2101592  1.545679

## [1] "SARIMA model accuracy"

##                       ME     RMSE      MAE         MPE     MAPE      MASE
## Training set  3.6472364 34.64385 19.76461  0.1514028 13.72940 0.4401973
## Test set     -0.2850796 37.94372 28.18172 -8.7619757 23.25073 0.6276629
##                     ACF1 Theil's U
## Training set -0.0693990        NA
## Test set      0.2310004  1.290106

## [1] "Dynamic Regression Model accuracy"

##                      ME     RMSE      MAE        MPE     MAPE      MASE
## Training set  6.587226 35.66254 24.62880  -1.473791 16.45129 0.5485324
## Test set     -5.090736 34.31713 28.79237 -18.383220 27.91181 0.6412633
##                    ACF1 Theil's U
## Training set 0.1312774        NA
## Test set     0.6180204   1.32675

## [1] "Dynamic Harmonic Regression Model accuracy"

##                      ME     RMSE      MAE        MPE     MAPE      MASE
## Training set -2.632729 34.33123 24.52279  -5.504969 17.47439 0.5461714
## Test set     -1.869454 56.62904 43.69774 -17.409737 38.41926 0.9732356
##                      ACF1 Theil's U
## Training set -0.05062828        NA
## Test set      0.75617783  2.051514
```

Despite having a lower RMSE than other models on the test data set the Dynamic Harmonic Regression Model is the only model to not show signicant information left in the residuals.

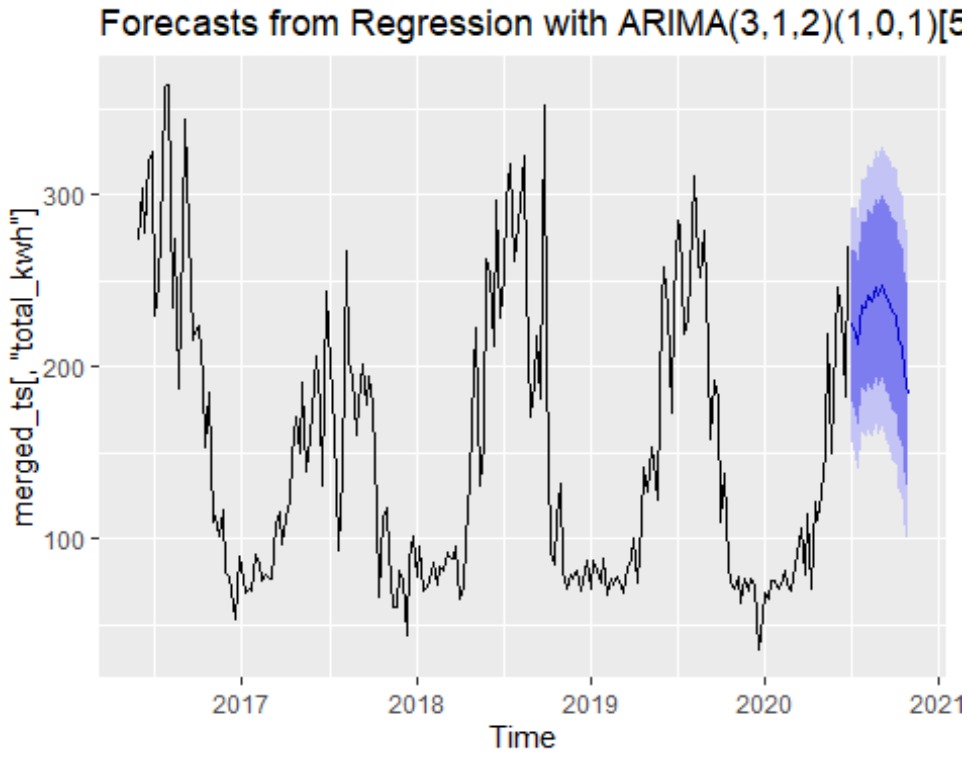
We have selected the dynamic harmonic regression model for our forecast.

Forecast:

We first began by applying dynamic harmonic regression to the entire data set to build our final forecasting model. Using a forecast of the next three months we applied the model to

the forecasted data (the forecast was built using a python script to pull weather data to simulate a forecast).

Three month forecast of the Dynamic Regression model



The point forecast itself:

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2020.500       224.6269 180.2854 268.9684 156.8125 292.4414
## 2020.519       219.7975 172.0289 267.5661 146.7418 292.8532
## 2020.538       213.6117 165.8393 261.3842 140.5501 286.6734
## 2020.558       235.8254 188.0229 283.6278 162.7178 308.9329
## 2020.577       233.9424 184.6739 283.2109 158.5927 309.2921
## 2020.596       241.4150 191.0606 291.7694 164.4046 318.4255
## 2020.615       237.5924 186.3265 288.8583 159.1879 315.9969
## 2020.635       246.4199 194.6675 298.1724 167.2714 325.5685
## 2020.654       241.1831 188.5855 293.7806 160.7421 321.6241
## 2020.673       247.3758 194.1040 300.6476 165.9036 328.8480
## 2020.692       240.6759 186.5515 294.8004 157.8997 323.4521
## 2020.712       239.2133 184.4699 293.9567 155.4905 322.9361
## 2020.731       231.5198 175.9978 287.0419 146.6062 316.4335
## 2020.750       229.3786 173.2263 285.5309 143.5010 315.2561
## 2020.769       217.3387 160.4272 274.2502 130.3001 304.3773
## 2020.788       212.1751 154.6380 269.7123 124.1797 300.1706
## 2020.808       197.4161 139.1547 255.6775 108.3130 286.5193
## 2020.827       183.8246 124.9430 242.7062  93.7729 273.8763
```

Solutions and Recommendations:

- How does the forecast help you make decisions?
  - Apply surge pricing during summer months due to seasonal increases power usage due to hot weather
  - Increase power storage capacity to meet max forecasted demand for power
  - Build out alternative energy production capacity to match demand and to take advantage of government tax credits for renewable energy projects
  - Stay aligned with 2035 goal of carbon pollution-free power sector
  - Balance reliability concerns with the increased storage capacity and traditional fossil fuel sources to supplement periods of reduced production from green energy sources
  - Anticipate a potential Carbon Tax which will proactively help protect the bottom line