MSADS 506 Final Project Modeling

Tommy Barron

2025-11-29

```
# Load packages
library(readxl)
library(fpp3)
```

```
## Registered S3 method overwritten by 'tsibble':
##   method               from
##   as_tibble.grouped_df dplyr
```

```
## -- Attaching packages --------------------------------------------- fpp3 1.0.2 --
```

```
## v tibble      3.2.1     v tsibble     1.1.6
## v dplyr       1.1.4     v tsibbledata 0.4.1
## v tidyr       1.3.1     v feasts      0.4.2
## v lubridate   1.9.4     v fable       0.4.1
## v ggplot2     3.5.1
```

```
## -- Conflicts ------------------------------------------------ fpp3_conflicts --
## x lubridate::date()    masks base::date()
## x dplyr::filter()      masks stats::filter()
## x tsibble::intersect() masks base::intersect()
## x tsibble::interval()  masks lubridate::interval()
## x dplyr::lag()         masks stats::lag()
## x tsibble::setdiff()   masks base::setdiff()
## x tsibble::union()     masks base::union()
```

```
library(dplyr)
library(tsibble)
library(lubridate)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats 1.0.0     v readr   2.1.5
## v purrr   1.0.4     v stringr 1.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()     masks stats::filter()
## x tsibble::interval() masks lubridate::interval()
## x dplyr::lag()        masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(gt)
```

```r
# Load data
df <- read_xlsx("SDGE-ELEC-2022to2025.xlsx")
head(df)
```

```
## # A tibble: 6 x 8
##   ZipCode Month  Year CustomerClass Combined TotalCustomers TotalkWh AveragekWh
##     <dbl> <dbl> <dbl> <chr>         <chr>              <dbl>    <dbl>      <dbl>
## 1   91901    10  2022 A             Y                      0        0          0
## 2   91901    10  2022 C             Y                      0        0          0
## 3   91901    10  2022 I             Y                      0        0          0
## 4   91901    10  2022 R             Y                   8132  3602486        443
## 5   91902    10  2022 A             Y                      0        0          0
## 6   91902    10  2022 C             Y                      0        0          0
```

Data Preparation

Region Assignment

```r
# This section assigns the region according to the county of San Diego and the United States Postal Ser
df$region <- recode(df$ZipCode,
                    "92007" = "North",
                    "92008" = "North",
                    "92009" = "North",
                    "92010" = "North",
                    "92011" = "North",
                    "92014" = "North",
                    "92024" = "North",
                    "92054" = "North",
                    "92055" = "North",
                    "92056" = "North",
                    "92057" = "North",
                    "92058" = "North",
                    "92067" = "North",
                    "92075" = "North",
                    "92081" = "North",
                    "92083" = "North",
                    "92084" = "North",
                    "92091" = "North",
                    "92672" = "North",
                    "92003" = "North",
                    "92004" = "North",
                    "92025" = "North",
                    "92026" = "North",
                    "92027" = "North",
                    "92028" = "North",
                    "92029" = "North",
                    "92036" = "North",
                    "92059" = "North",
                    "92060" = "North",
```

```
                    "92061" = "North",
                    "92064" = "North",
                    "92065" = "North",
                    "92066" = "North",
                    "92069" = "North",
                    "92070" = "North",
                    "92078" = "North",
                    "92082" = "North",
                    "92086" = "North",
                    "92096" = "North",
                    "92127" = "North",
                    "92128" = "North",
                    "92129" = "North",
                    "92259" = "North",
                    "92536" = "North",
                    "92037" = "North",
                    "92093" = "North",
                    "92106" = "North",
                    "92107" = "North",
                    "92108" = "North",
                    "92109" = "North",
                    "92110" = "North",
                    "92111" = "North",
                    "92117" = "North",
                    "92119" = "North",
                    "92120" = "North",
                    "92121" = "North",
                    "92122" = "North",
                    "92123" = "North",
                    "92124" = "North",
                    "92126" = "North",
                    "92130" = "North",
                    "92131" = "North",
                    "92140" = "North",
                    "92145" = "North",
                    "92161" = "North",
                    "91901" = "East",
                    "91905" = "East",
                    "91906" = "East",
                    "91916" = "East",
                    "91917" = "East",
                    "91931" = "East",
                    "91934" = "East",
                    "91935" = "East",
                    "91941" = "East",
                    "91942" = "East",
                    "91945" = "East",
                    "91948" = "East",
                    "91962" = "East",
                    "91963" = "East",
                    "91977" = "East",
                    "91978" = "East",
                    "91980" = "East",
```

```
                    "92019" = "East",
                    "92020" = "East",
                    "92021" = "East",
                    "92040" = "East",
                    "92071" = "East",
                    "92101" = "Central",
                    "92102" = "Central",
                    "92103" = "Central",
                    "92104" = "Central",
                    "92015" = "Central",
                    "92113" = "Central",
                    "92114" = "Central",
                    "92115" = "Central",
                    "92116" = "Central",
                    "92134" = "Central",
                    "92136" = "Central",
                    "92139" = "Central",
                    "92182" = "Central",
                    "91902" = "South",
                    "91910" = "South",
                    "91911" = "South",
                    "91913" = "South",
                    "91914" = "South",
                    "91915" = "South",
                    "91932" = "South",
                    "91950" = "South",
                    "92118" = "South",
                    "92135" = "South",
                    "92154" = "South",
                    "92155" = "South",
                    "92173" = "South",
                    "91912" = "South",
                    "92049" = "North",
                    "92068" = "North",
                    "92072" = "North",
                    "92079" = "North",
                    "92085" = "North",
                    "92092" = "North",
                    "92105" = "Central",
                    "92112" = "Central",
                    "92132" = "Central",
                    "92152" = "Central",
                    "92158" = "Central",
                    "92179" = "South",
                    "92199" = "Central")
```

```
## Warning: Unreplaced values treated as NA as `.x` is not compatible.
## Please specify replacements exhaustively or supply `.default`.
```

```
df[is.na(df$region), ]
```

```
## # A tibble: 2,695 x 9
##    ZipCode Month  Year CustomerClass Combined TotalCustomers TotalkWh AveragekWh
```

```
##       <dbl> <dbl> <dbl> <chr>         <chr>              <dbl>    <dbl>      <dbl>
##  1   92045    10  2022 C             Y                      0        0          0
##  2   92045    10  2022 I             Y                      0        0          0
##  3   92045    10  2022 R             Y                      0        0          0
##  4   92624    10  2022 A             Y                      0        0          0
##  5   92624    10  2022 C             Y                      0        0          0
##  6   92624    10  2022 I             Y                      0        0          0
##  7   92624    10  2022 R             N                   2862  1337589        467
##  8   92625    10  2022 A             Y                      0        0          0
##  9   92629    10  2022 A             Y                      0        0          0
## 10   92629    10  2022 C             Y                      0        0          0
## # i 2,685 more rows
## # i 1 more variable: region <chr>
```

```
unique(df$ZipCode[is.na(df$region)])
```

```
##  [1] 92045 92624 92625 92629 92649 92651 92653 92654 92656 92673 92674 92675
## [13] 92676 92677 92679 92688 92690 92691 92692 92693 92694
```

```
# Removed ZipCodes:
# 92045 - No where on the map
# 92624 - In Orange County
# 92625 - In Orange County
# 92629 - In Orange County
# 92649 - In Orange County
# 92651 - In Orange County
# 92653 - In Orange County
# 92654 - In Orange County
# 92656 - In Orange County
# 92673 - In Orange County
# 92674 - In Orange County
# 92675 - In Orange County
# 92676 - In Orange County
# 92677 - In Orange County
# 92679 - In Orange County
# 92688 - In Orange County
# 92690 - In Orange County
# 92691 - In Orange County
# 92692 - In Orange County
# 92693 - In Orange County
# 92694 - In Orange County

remove_zips <- c("92045", "92624", "92625", "92629", "92649", "92651", "92653", "92654", "92656", "9267

df_clean <- df |>
  filter(!ZipCode %in% remove_zips)
```

```
# Check if there are any missing ZipCodes with no assigned region
unique(df_clean$ZipCode[is.na(df_clean$region)])
```

```
## numeric(0)
```

Class Adjustment

```
# This section filters out the A and I CustomerClass because they do not have any contributing kWh
filter_class <- c("A", "I")

df_clean <- df_clean |>
  filter(!CustomerClass %in% filter_class)
```

```
head(df_clean)
```

```
## # A tibble: 6 x 9
##   ZipCode Month  Year CustomerClass Combined TotalCustomers TotalkWh AveragekWh
##     <dbl> <dbl> <dbl> <chr>         <chr>              <dbl>    <dbl>      <dbl>
## 1   91901    10  2022 C             Y                      0        0          0
## 2   91901    10  2022 R             Y                   8132  3602486        443
## 3   91902    10  2022 C             Y                      0        0          0
## 4   91902    10  2022 R             N                   4487  1951812        435
## 5   91905    10  2022 C             Y                      0        0          0
## 6   91905    10  2022 R             Y                    273   109986        403
## # i 1 more variable: region <chr>
```

Modeling

Tsibble Creation

```
# This groups the date and region with the customer class C and R
# Afterwards, it creates the date column for creating a TSIBBLE
df_clean1 <- df_clean |>
  group_by(region, Year, Month, CustomerClass) |>
  summarize(
    total_count = sum(TotalkWh, na.rm = TRUE),
    .groups = "drop"
  ) |>
  mutate(
    Date = yearmonth(paste(Year, Month, "1", sep = "-"))
  )

# This data frame contains only the R CustomerClass
df_r <- df_clean1 |>
  filter(!CustomerClass %in% c("R"))

# This data frame contains only the C CustomerClass
df_c <- df_clean1 |>
  filter(!CustomerClass %in% c("C"))

# Create Tsibble for R and C
df_r_tsibble <- df_r |>
  as_tsibble(
    index = Date,
    key = region
```

```
  ) |>
  fill_gaps(total_count = 0)

df_c_tsibble <- df_c |>
  as_tsibble(
    index = Date,
    key = region
  )

# This section combines the C and R CustomerClass
df_combined <- df_clean |>
  group_by(region, Year, Month) |>
  summarize(
    total_count = sum(TotalkWh, na.rm = TRUE),
    .groups = "drop"
  ) |>
  mutate(
    Date = yearmonth(paste(Year, Month, "1", sep = "-"))
  )

df_combined_tsibble <- df_combined |>
  as_tsibble(
    index = Date,
    key = region
  ) |>
  fill_gaps(total_count = 0)
```

Modeling and Validation for R CustomerClass

```
# This section creates models for the R CustomerClass and Forecasts 12 months
trn_cutoff <- yearmonth(df_r_tsibble$Date |>
                          max() |>
                          as_date() - years(1))

df_r_trn <- df_r_tsibble |>
  filter(Date <= trn_cutoff)

df_r_trn_fit <- df_r_trn |>
  model(
    tslm_logv = TSLM(log(total_count) ~ trend()),
    tslm_log1p = TSLM(log1p(total_count) ~ trend()),
    tslm_sqrt = TSLM(sqrt(total_count) ~ trend()),
    tslm_v = TSLM(total_count ~ trend()),
    arima = ARIMA(total_count),
    ets = ETS(total_count)
  )
```

```
## Warning: 4 errors (1 unique) encountered for tslm_logv
## [4] NA/NaN/Inf in 'y'
```

.model	region	.type	ME	RMSE	MAE	MPE	MAPE	MASE	RMS
tslm_v	Central	Test	453,137.2	1,791,312.6	1,213,444.9	-15.5	37.1	0.7	
ets	Central	Test	472,098.8	1,834,221.9	1,239,729.3	-15.5	37.7	0.8	
tslm_sqrt	Central	Test	740,177.0	1,906,028.0	1,422,372.9	-10.0	39.1	0.9	
arima	Central	Test	783,100.0	1,937,714.7	1,462,597.5	-9.3	39.6	0.9	
tslm_log1p	Central	Test	-1,441,081.6	2,936,133.8	2,363,619.5	-61.6	72.6	1.4	
arima	East	Test	2,545,463.0	3,865,124.0	3,577,903.2	6.8	10.5	0.3	
ets	East	Test	-3,162,744.5	4,296,830.8	3,241,522.1	-10.3	10.5	0.3	
arima	South	Test	122,591.0	4,480,082.6	3,760,369.9	-4.0	18.3	0.4	
ets	South	Test	-1,040,784.2	4,597,754.1	3,881,313.1	-9.4	19.8	0.4	
tslm_sqrt	South	Test	-1,840,663.4	5,137,733.9	4,290,236.7	-13.5	22.3	0.4	
tslm_v	South	Test	-2,943,984.4	5,710,821.1	4,579,243.3	-18.7	24.5	0.5	
tslm_v	East	Test	6,173,097.7	6,909,644.7	6,251,700.1	17.6	17.9	0.5	
tslm_sqrt	East	Test	6,668,642.8	7,339,069.3	6,668,642.8	19.1	19.1	0.5	
tslm_log1p	East	Test	-1,234,958.1	7,639,268.7	6,887,099.4	-4.6	20.2	0.6	
tslm_log1p	South	Test	-9,042,794.7	9,813,150.1	9,042,794.7	-43.7	43.7	0.9	
ets	North	Test	-2,421,239.9	12,814,472.1	11,009,020.5	-7.0	16.9	0.3	
arima	North	Test	-3,747,759.9	13,129,889.5	11,410,717.9	-9.0	17.8	0.3	
tslm_log1p	North	Test	17,900,631.3	30,718,613.4	22,695,431.1	21.9	29.7	0.5	
tslm_sqrt	North	Test	26,362,125.7	31,313,699.5	26,362,125.7	36.1	36.1	0.6	
tslm_v	North	Test	35,702,191.2	41,135,736.5	35,702,191.2	49.6	49.6	0.9	
tslm_logv	Central	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
tslm_logv	East	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
tslm_logv	North	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
tslm_logv	South	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na

```
df_r_trn_fc <- df_r_trn_fit |>
  forecast(h = "1 year")

df_r_trn_fc |>
  accuracy(df_r_tsibble) |>
  arrange(RMSE) |>
  gt() |>
  fmt_number(decimals = 1)
```

Modeling and Validation for C CustomerClass

```
# This section creates models for the C CustomerClass and Forecasts 12 months
trn_cutoff <- yearmonth(df_c_tsibble$Date |>
                          max() |>
                          as_date() - years(1))

df_c_trn <- df_c_tsibble |>
  filter(Date <= trn_cutoff)
```

```
df_c_trn_fit <- df_c_trn |>
  model(
    tslm_logv = TSLM(log(total_count) ~ trend()),
    tslm_log1p = TSLM(log1p(total_count) ~ trend()),
    tslm_sqrt = TSLM(sqrt(total_count) ~ trend()),
    tslm_v = TSLM(total_count ~ trend()),
    arima = ARIMA(total_count),
    ets = ETS(total_count)
  )
```

```
## Warning: 4 errors (1 unique) encountered for arima
## [4] .data contains implicit gaps in time. You should check your data and convert implicit gaps into e
```

```
## Warning: 4 errors (1 unique) encountered for ets
## [4] .data contains implicit gaps in time. You should check your data and convert implicit gaps into e
```

```
df_c_trn_fc <- df_c_trn_fit |>
  forecast(h = "1 year")

df_c_trn_fc |>
  accuracy(df_c_tsibble) |>
  arrange(RMSE) |>
  gt() |>
  fmt_number(decimals = 1)
```

Modeling and Validation for Combined CustomerClass (C and R)

```
# This section creates models for the combined C and R CustomerClass and Forecasts 12 months
trn_cutoff <- yearmonth(df_combined_tsibble$Date |>
                          max() |>
                          as_date() - years(1))

df_combined_trn <- df_combined_tsibble |>
  filter(Date <= trn_cutoff)

df_combined_trn_fit <- df_combined_trn |>
  model(
    tslm_logv = TSLM(log(total_count) ~ trend()),
    tslm_log1p = TSLM(log1p(total_count) ~ trend()),
    tslm_sqrt = TSLM(sqrt(total_count) ~ trend()),
    tslm_v = TSLM(total_count ~ trend()),
    arima = ARIMA(total_count),
    ets = ETS(total_count)
  )
```

```
## Warning: 4 errors (1 unique) encountered for tslm_logv
## [4] NA/NaN/Inf in 'y'
```

.model	region	.type	ME	RMSE	MAE	MPE	MAPE	MASE	RMS
tslm_log1p	South	Test	-695,684.8	2,919,320.5	2,511,867.9	-6.6	14.7	0.6	
tslm_logv	South	Test	-695,687.9	2,919,321.1	2,511,867.8	-6.6	14.7	0.6	
tslm_sqrt	South	Test	2,094,665.6	3,670,035.8	2,730,759.4	9.2	13.6	0.6	
tslm_v	South	Test	2,253,657.5	3,791,585.4	2,755,700.9	10.1	13.6	0.6	
tslm_v	Central	Test	-3,328,360.2	4,502,869.1	3,930,860.0	-19.1	21.2	0.6	
tslm_sqrt	Central	Test	-5,257,943.7	5,963,808.0	5,257,943.7	-28.2	28.2	0.8	
tslm_logv	East	Test	329,193.0	10,079,323.8	8,392,921.2	-6.0	23.2	0.4	
tslm_log1p	East	Test	329,195.1	10,079,323.9	8,392,920.9	-6.0	23.2	0.4	
tslm_log1p	Central	Test	-9,921,070.7	10,480,428.8	9,921,070.7	-50.1	50.1	1.4	
tslm_logv	Central	Test	-9,921,071.7	10,480,429.9	9,921,071.7	-50.1	50.1	1.4	
tslm_sqrt	East	Test	6,109,185.3	12,045,577.2	8,698,175.9	10.6	20.4	0.5	
tslm_v	East	Test	9,087,904.3	14,132,096.8	9,759,614.5	19.0	21.8	0.5	
tslm_logv	North	Test	7,443,445.6	20,234,305.5	14,193,122.8	5.3	16.0	0.5	
tslm_log1p	North	Test	7,443,447.8	20,234,306.3	14,193,122.9	5.3	16.0	0.5	
tslm_sqrt	North	Test	21,921,521.8	29,452,855.6	22,120,207.6	24.2	24.6	0.7	
tslm_v	North	Test	32,667,563.6	39,402,066.1	32,667,563.6	38.0	38.0	1.0	
arima	Central	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
arima	East	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
arima	North	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
arima	South	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
ets	Central	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
ets	East	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
ets	North	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na
ets	South	Test	NaN	NaN	NaN	NaN	NaN	NaN	Na

```
df_combined_trn_fc <- df_combined_trn_fit |>
  forecast(h = "1 year")

df_combined_trn_fc |>
  accuracy(df_combined_tsibble) |>
  arrange(RMSE) |>
  gt() |>
  fmt_number(decimals = 1)
```

.model	region	.type	ME	RMSE	MAE	MPE	MAPE	MASE	RMS
tslm_sqrt	Central	Test	205,325.1	4,323,873.0	3,464,018.3	-2.3	13.0	0.4	
tslm_v	Central	Test	-792,771.0	4,430,916.2	3,600,664.1	-6.2	14.1	0.4	
tslm_v	South	Test	1,067,932.2	4,814,139.1	3,752,759.7	1.4	9.1	0.3	
tslm_sqrt	South	Test	3,099,704.1	5,512,512.5	4,363,946.4	6.5	10.3	0.3	
ets	Central	Test	-548,404.8	5,644,161.1	4,513,361.4	-6.3	17.3	0.5	
ets	South	Test	-5,522,864.4	7,027,382.9	6,048,924.5	-15.0	16.0	0.4	
arima	South	Test	5,727,726.9	7,189,501.7	6,005,738.5	13.2	14.0	0.4	
tslm_log1p	South	Test	-1,704,025.6	7,434,102.4	5,539,747.2	-4.8	13.8	0.4	
tslm_log1p	Central	Test	-6,968,421.7	10,639,421.9	9,495,404.8	-32.8	39.5	1.1	
arima	Central	Test	9,175,861.9	10,758,821.9	9,175,861.9	31.4	31.4	1.1	
arima	East	Test	-255,238.4	12,357,899.6	10,107,269.1	-3.2	14.5	0.3	
ets	East	Test	-19,209,845.8	22,742,077.1	19,902,945.9	-30.9	31.6	0.6	
tslm_sqrt	East	Test	19,025,584.9	23,195,862.4	19,107,687.6	24.5	24.7	0.6	
tslm_v	East	Test	19,711,400.3	24,012,180.9	19,711,400.3	25.5	25.5	0.6	
tslm_log1p	East	Test	12,939,298.8	24,916,101.3	17,128,616.2	15.1	21.5	0.5	
ets	North	Test	-7,110,717.2	27,963,480.6	24,231,834.6	-8.0	16.8	0.3	
arima	North	Test	-12,337,125.5	29,725,386.7	26,845,038.8	-11.7	19.0	0.3	
tslm_log1p	North	Test	41,944,754.7	67,371,367.6	49,365,760.5	24.7	30.1	0.6	
tslm_sqrt	North	Test	59,373,760.0	69,002,640.7	59,373,760.0	37.9	37.9	0.7	
tslm_v	North	Test	78,258,510.3	88,767,926.1	78,258,510.3	50.6	50.6	1.0	
tslm_logv	Central	Test	NaN	NaN	NaN	NaN	NaN	NaN	N
tslm_logv	East	Test	NaN	NaN	NaN	NaN	NaN	NaN	N
tslm_logv	North	Test	NaN	NaN	NaN	NaN	NaN	NaN	N
tslm_logv	South	Test	NaN	NaN	NaN	NaN	NaN	NaN	N