GLA Task - Exploratory analysis in R

Code

As dataset is large, I did some initial data exploration to demonstrate proof of concept using the first 1 million rows.

```
library(data.table)
library(dplyr)
library(prophet)
```

Read in and prepare data

```
# read in first csv file of 1 million rows
csv_name = 'C:/Users/User/Documents/my_code_files/R_codes/gla_interview_task
_011017/smart_meter_data/separate_csvs/Power-Networks-LCL-June2015(withAcorn
Gps)v2_2.csv'
mydata <- fread(csv_name, drop = c("Acorn", "Acorn_grouped"))
```

```
Bumped column 4 to type character on data row 1932, field contains 'Null'. C
oercing previously read values in this column from logical, integer or numer
ic back to character which may not be lossless; e.g., if '00' and '000' occu
rred before they will now be just '0', and there may be inconsistencies wit
h treatment of ',,' and ',NA,' too (if they occurred in this column before t
he bump). If this matters please rerun and set 'colClasses' to 'character' f
or this column. Please note that column type detection uses a sample of 1,00
0 rows (100 rows at 10 points) so hopefully this message should be very rar
e. If reporting to datatable-help, please rerun and include the output from
verbose=TRUE.
```

```
head(mydata)
```

```
# Compute daily consumption per household
daily_kwh_per_household <- mydata %>%
  # use households on standard tariff only
  filter(stdorToU=="Std") %>%
  # rename variables for ease of reference
  rename(kwh = 'KWH/hh (per half hour)') %>%
  # extract date from datetime
  mutate(
    dt = as.POSIXct(paste(DateTime)),
    day = as.Date(strftime(dt, format = "%D"), "%m/%d/%y")) %>%
  # compute total daily consumption for each household
  group_by(day, LCLid) %>%
  summarise(total_kwh = sum(as.numeric(kwh)))
```

```
NAs introduced by coercionNAs introduced by coercionNAs introduced by coerci
onNAs introduced by coercionNAs introduced by coercionNAs introduced by coer
cionNAs introduced by coercionNAs introduced by coercionNAs introduced by co
ercionNAs introduced by coercionNAs introduced by coercionNAs introduced by
coercionNAs introduced by coercionNAs introduced by coercionNAs introduced b
y coercionNAs introduced by coercionNAs introduced by coercionNAs introduce
d by coercionNAs introduced by coercionNAs introduced by coercionNAs introdu
ced by coercionNAs introduced by coercionNAs introduced by coercionNAs intro
duced by coercionNAs introduced by coercionNAs introduced by coercionNAs int
roduced by coercion
```

```
str(daily_kwh_per_household)
```

```
Classes 'grouped_df', 'tbl_df', 'tbl' and 'data.frame': 20861 obs. of  3 var
iables:
 $ day      : Date, format: "2011-12-07" "2011-12-08" "2011-12-08" ...
 $ LCLid    : chr  "MAC000037" "MAC000037" "MAC000038" "MAC000039" ...
 $ total_kwh: num  0.014 0.445 7.611 10.188 15.707 ...
 - attr(*, "vars")= chr "day"
 - attr(*, "drop")= logi TRUE
```

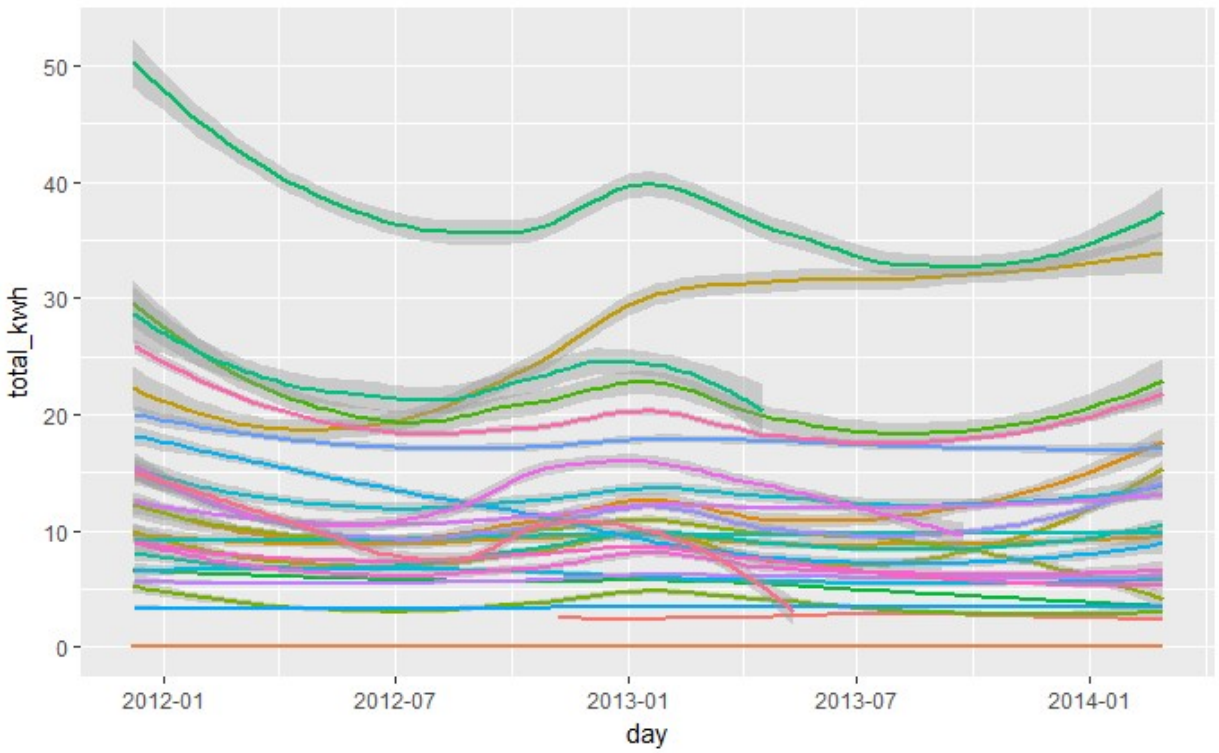
```
length(unique(daily_kwh_per_household$LCLid)) # 27 households
```

```
[1] 27
```

```
# Save as CSV
# fwrite(daily_kwh_per_household, file = "daily_kwh_per_household_csv2_2.cs
v")
```

plots - by household

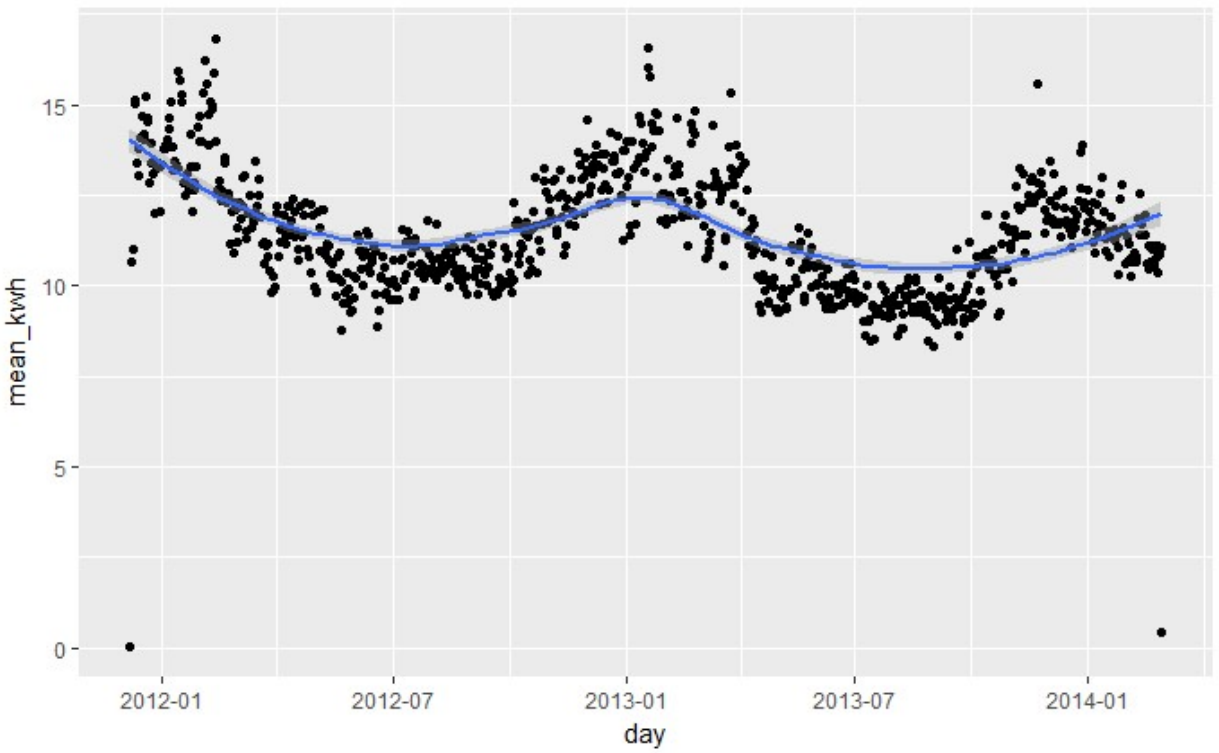
```
# plots
# plot by household
ggplot(data = daily_kwh_per_household,
mapping = aes(x = day, y = total_kwh)) +
  #geom_point(aes(colour = LCLid), show.legend = FALSE) +
  geom_smooth(aes(colour = LCLid), show.legend = FALSE)
```



Get mean per day across households

```
mpd <- summarise(daily_kwh_per_household, mean_kwh = mean(total_kwh))
head(mpd)
```

```
# plot means
ggplot(data = mpd,
       mapping = aes(x = day, y = mean_kwh)) +
  geom_point(show.legend = FALSE) +
  geom_smooth(show.legend = FALSE)
```



Prepare df for time series analysis

```
mdkwh <- select(ungroup(mpd), ds = day, y = mean_kwh)
head(mdkwh)
```

Fit additive regression model

```
arm <- prophet(mdkwh, growth = "linear",
               yearly.seasonality=TRUE,
               weekly.seasonality=FALSE)
```

```
Disabling daily seasonality. Run prophet with daily.seasonality=TRUE to over
ride this.
```

```
Initial log joint probability = -185.96
Optimization terminated normally:
  Convergence detected: relative gradient magnitude is below tolerance
```

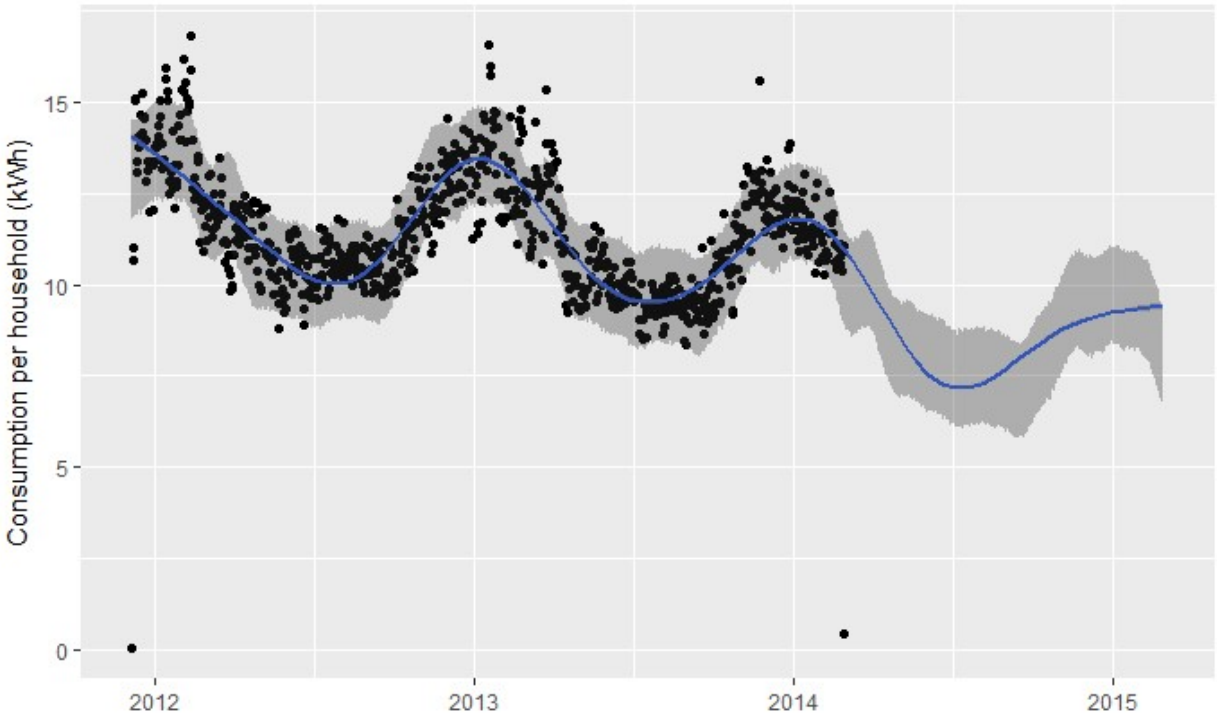
predict consumption for the next year

```
# first make list of dates to predict for
future <- make_future_dataframe(arm, periods = 365)
# make the forecast
forecast <- predict(arm, future)
tail(forecast[c('ds', 'yhat', 'yhat_lower', 'yhat_upper')])
```

simple plot with raw data plus forecast 1y into future

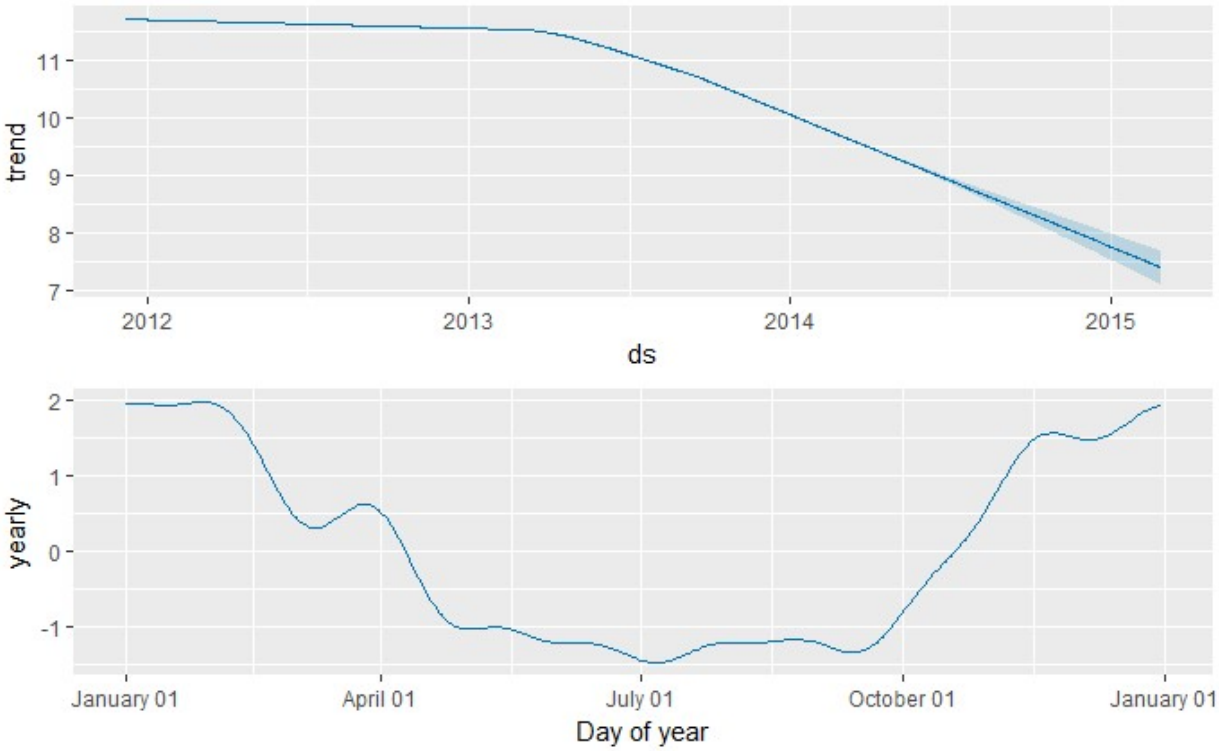
```
ggplot(data = arm$history, mapping = aes(x = ds, y = y)) +
  geom_point() +
  geom_smooth(data = forecast, mapping = aes(x = ds, y = yhat)) +
  geom_ribbon(data = forecast, mapping = aes(x = ds, y = yhat,
                                             ymin = yhat_lower,
                                             ymax = yhat_upper), alpha = 1/
3) +
  ylab("Consumption per household (kWh)") +
  xlab("")
```

```
Ignoring unknown aesthetics: y
```



plot seasonal components

```
prophet_plot_components(arm, forecast)
```


trend
11
10
9
8
7
2012
2013
2014
2015
ds
yearly
2
1
0
-1
January 01
April 01
July 01
October 01
January 01
Day of year