Clustering real data with ten customers without iterations

1 Clutering approach

1. *Compute quantiles of distributions across each hour of day*
2. *Compute JS distance between households for each hour of day*
3. *Total distance between households computed as sum of JS distances for all hours*
4. *Cluster using this distance with hierarchical clustering algorithm (method "complete")*

2 Plots of raw data

Five households are considered for the following analysis from the SGSC data set. The data sets contain three columns `customer_id`, `reading_datetime` and `general_supply_kwh`. They consist of half-hourly data from 2012-2014. Figure 1 shows how the raw time plots for these five households look like. Since all the dataset looks squeezed on this linear time scale, Figure 2 shows how the raw plot looks for 2 months (Sept 2013 and Oct 2013).

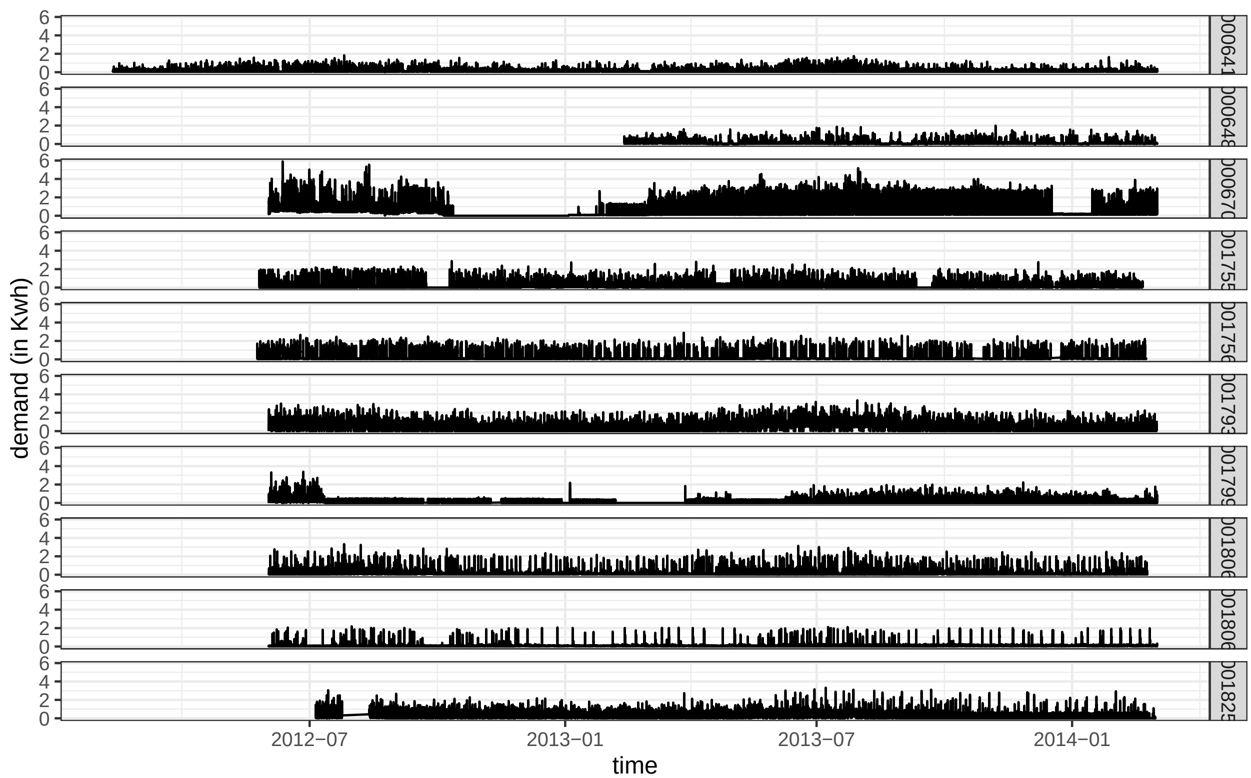


Figure 1: The raw time plots for demand shown for the entire observation period between 2012-2014 for 5 households (facets). The data is squeezed stopping us from seeing any behavioral patterns.

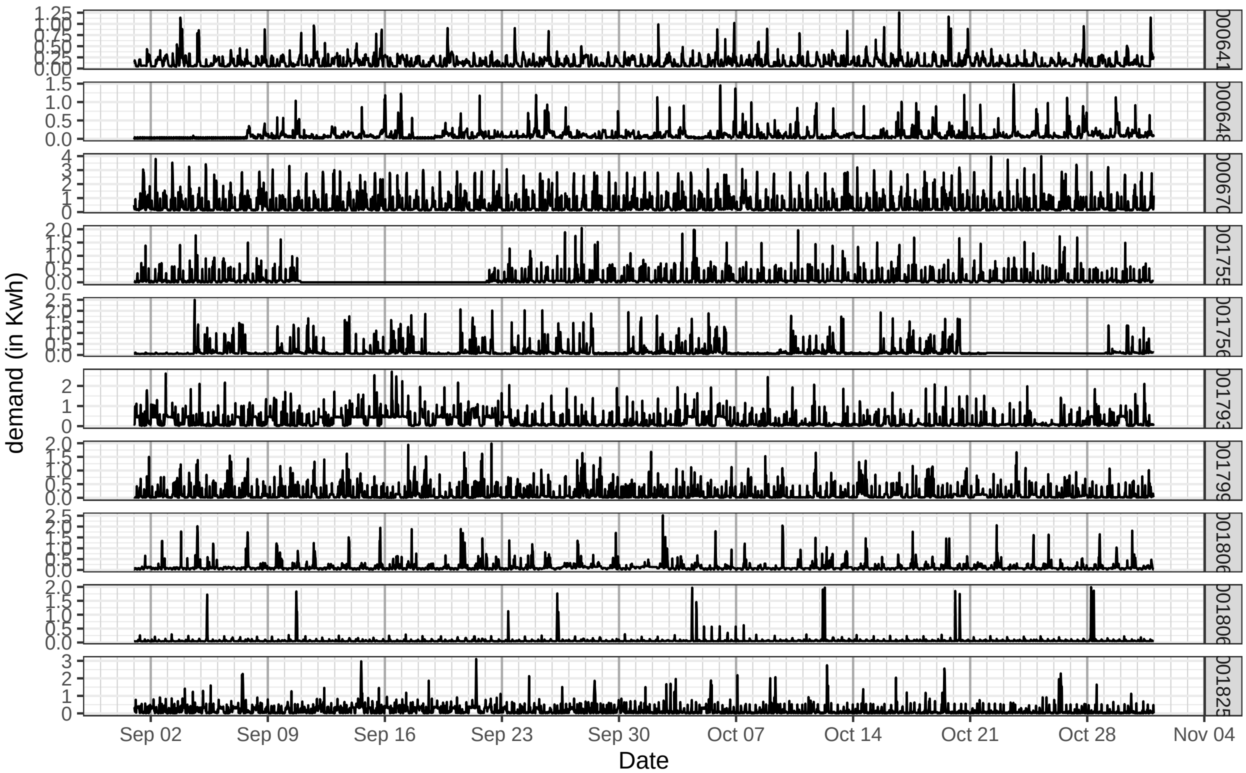


Figure 2: The raw plots for ten customers shown for Sept 2013-Oct 2013. The data is zoomed in and the y-scales made free to emphasize weekly, hourly or any behavior/patterns that they might have. There is some daily and weekly pattern in all households except the 4th one, that has spikes which seem to occur at irregular intervals.

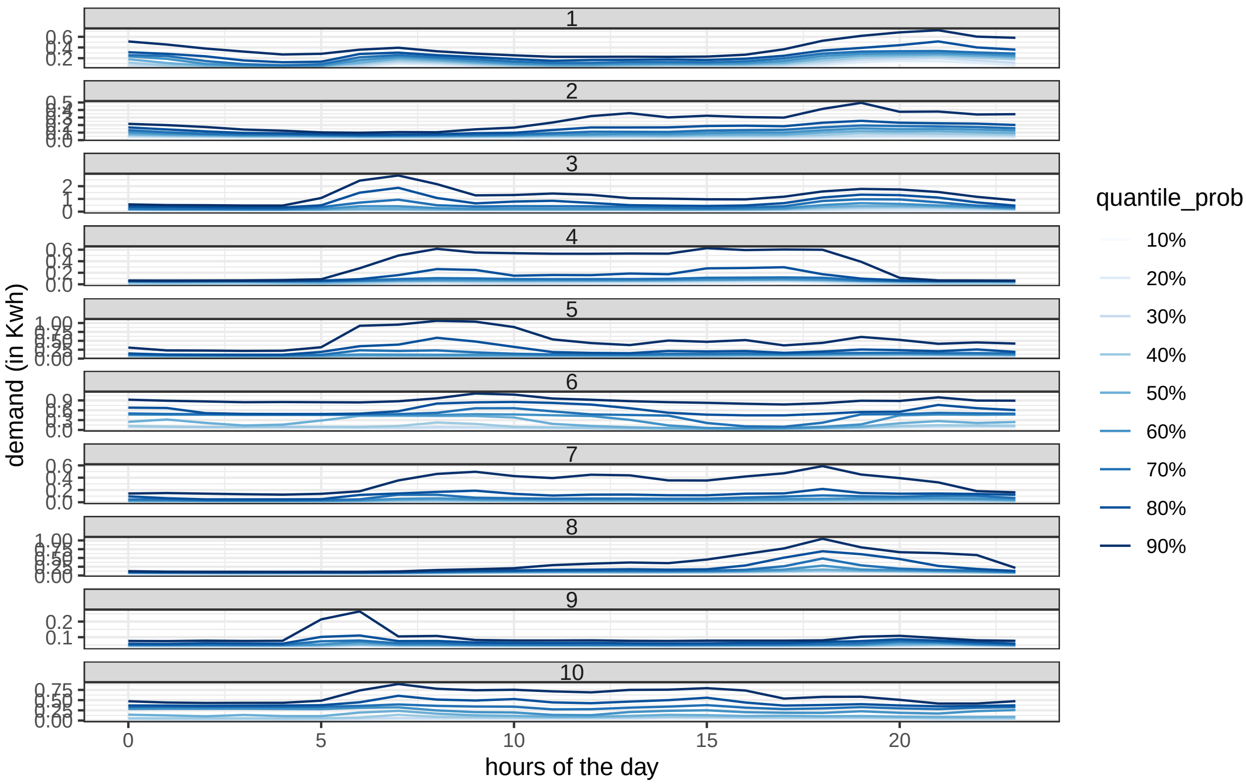
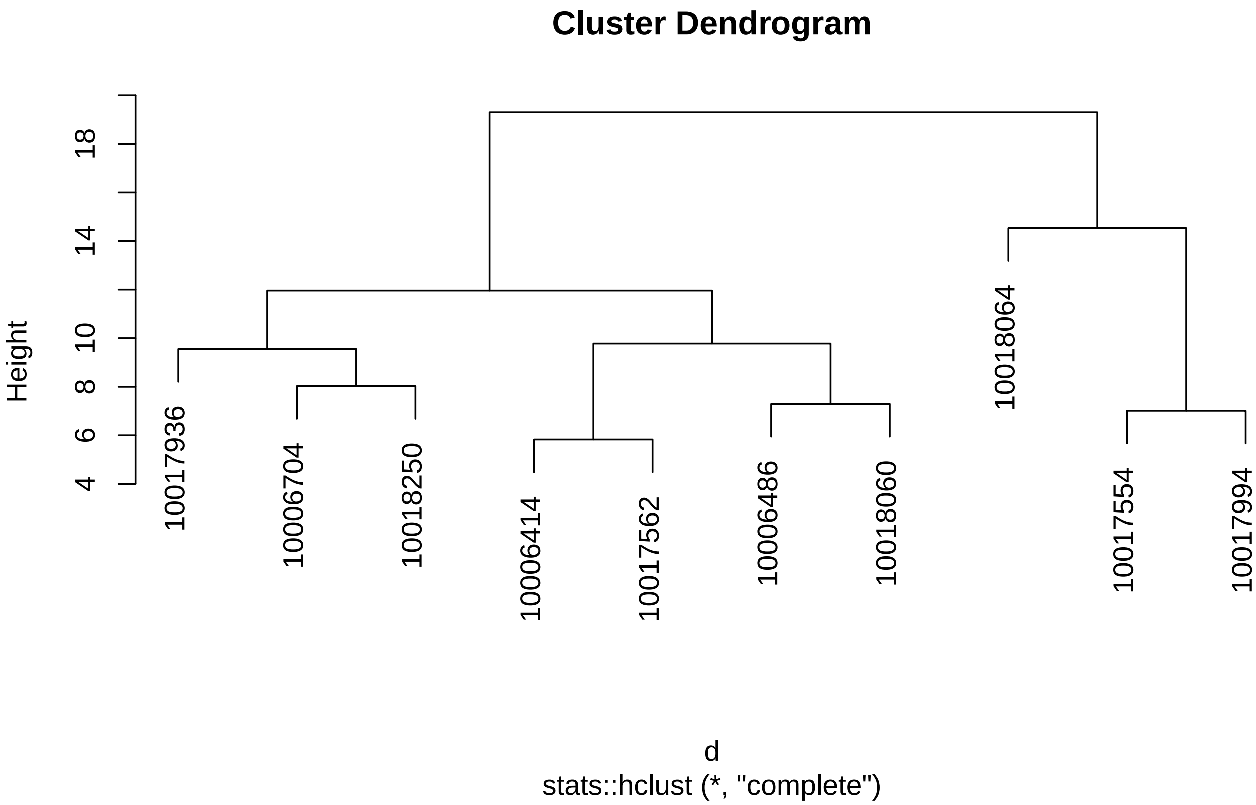
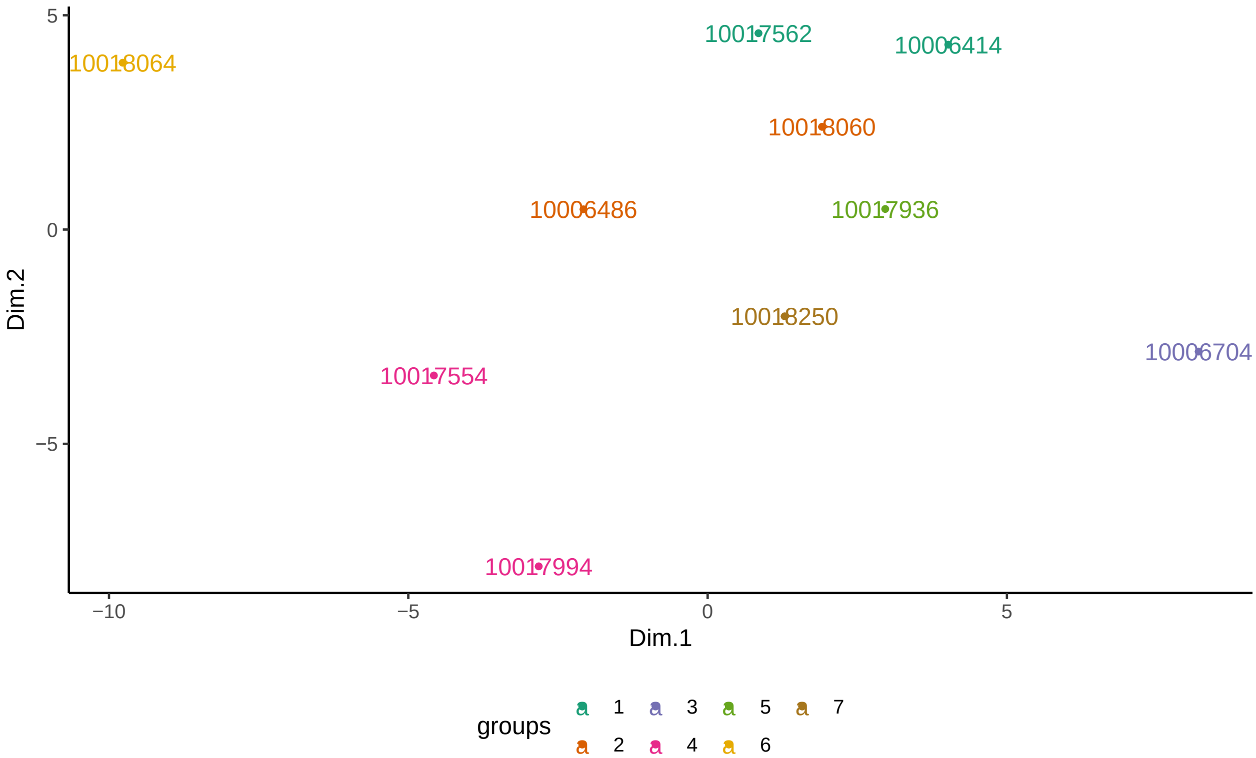


Figure 3: The deciles plots for 5 households are shown for entire observation period across hours of the day. The y-scales are made free to allow us to see daily patterns for each households. The deciles for all households tend to show an increase in the morning and evening hours, although the hours differ. Except for the 90th decile, the deciles for the 4th household look pretty flat.

3 Clustering results

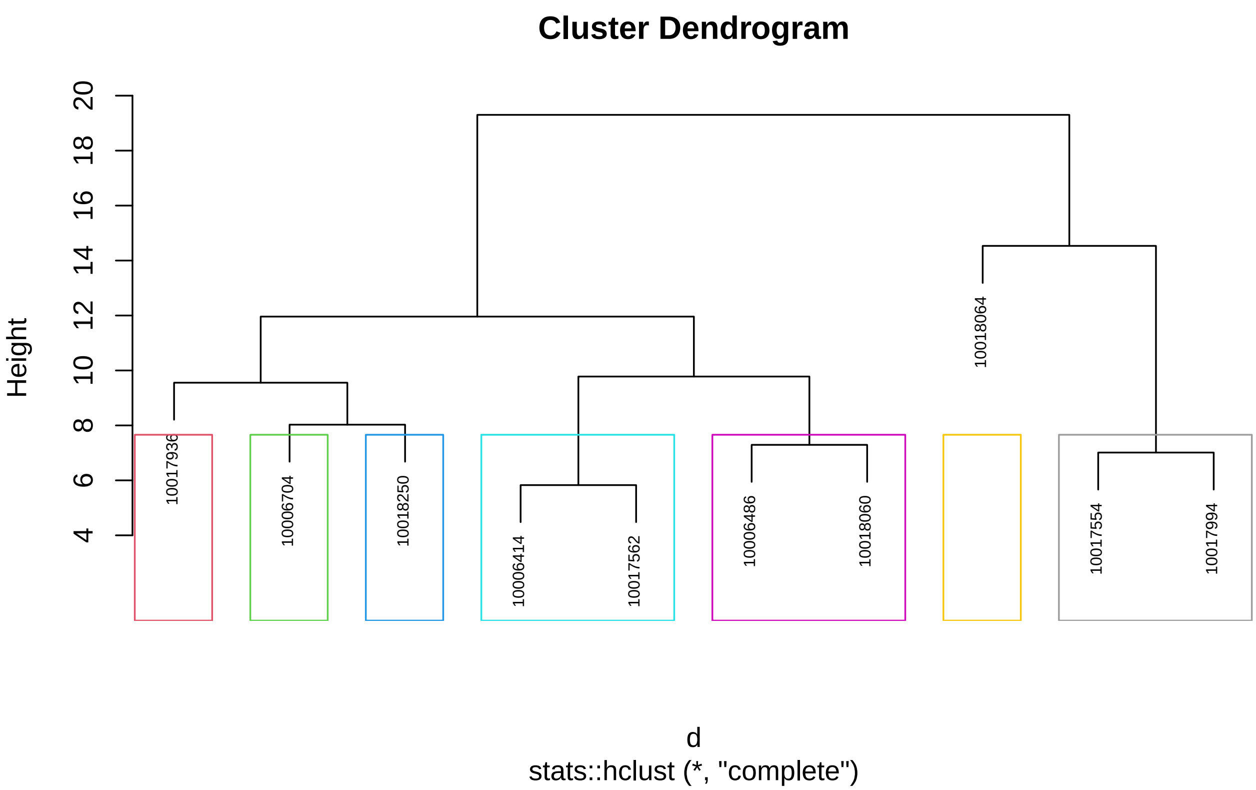
The following figures show dendograms when we are using optimal number of clusters (k) from k = fpc::nselectboot(), k = 2, 3, 4 and 5. We observe that the 4^{th} household tend to get split as different clusters as increase the number of clusters.





Optimal number of clusters as defined by the *fpc::nselectboot* is 7.

k = 7



4 Cluster characterization

The quartile deviation of different clusters are drawn. The shape of daily load curve is different for all of them. Across each cluster, customers should show different shape and within each cluster, customers should differ in size.