ATUS Clustering

Joe Marlo

May 24 2020

Contents

EDA 1

EDA 4

Transformations 5

PCA 9

Resampling the data using survey weights 10

Clustering 11

Hierarchical cluster 11

kmeans clustering 18

Model based clustering 22

```
library(tidyverse)
library(pander)
library(mclust)
library(NbClust)
options(mc.cores = parallel::detectCores())
set.seed(44)
theme_set(theme_minimal())

atus_long <- read_tsv('Data/atus.tsv')
demographics <- read_tsv('Data/demographic.tsv')
```

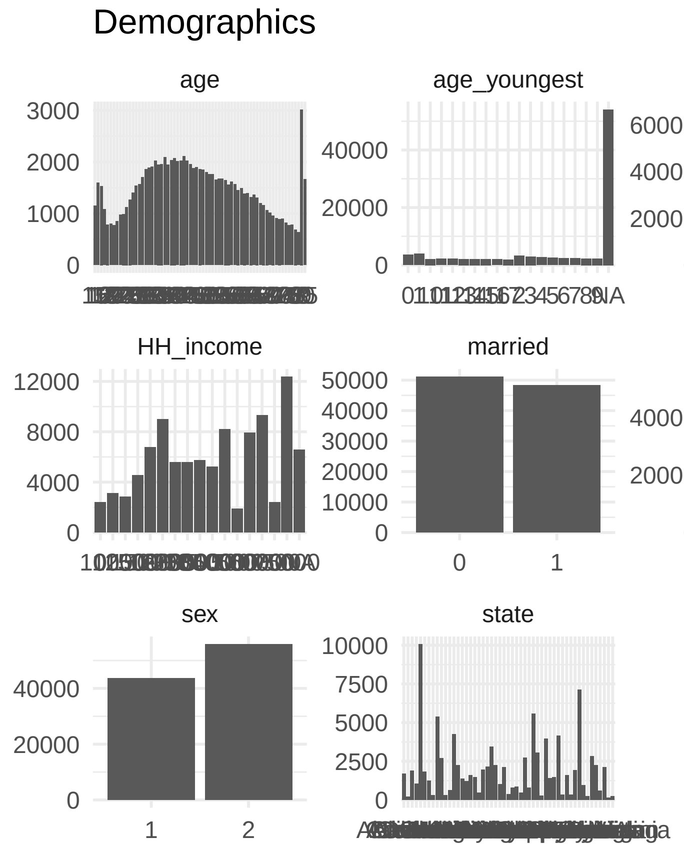
EDA

Only contains weekend observations

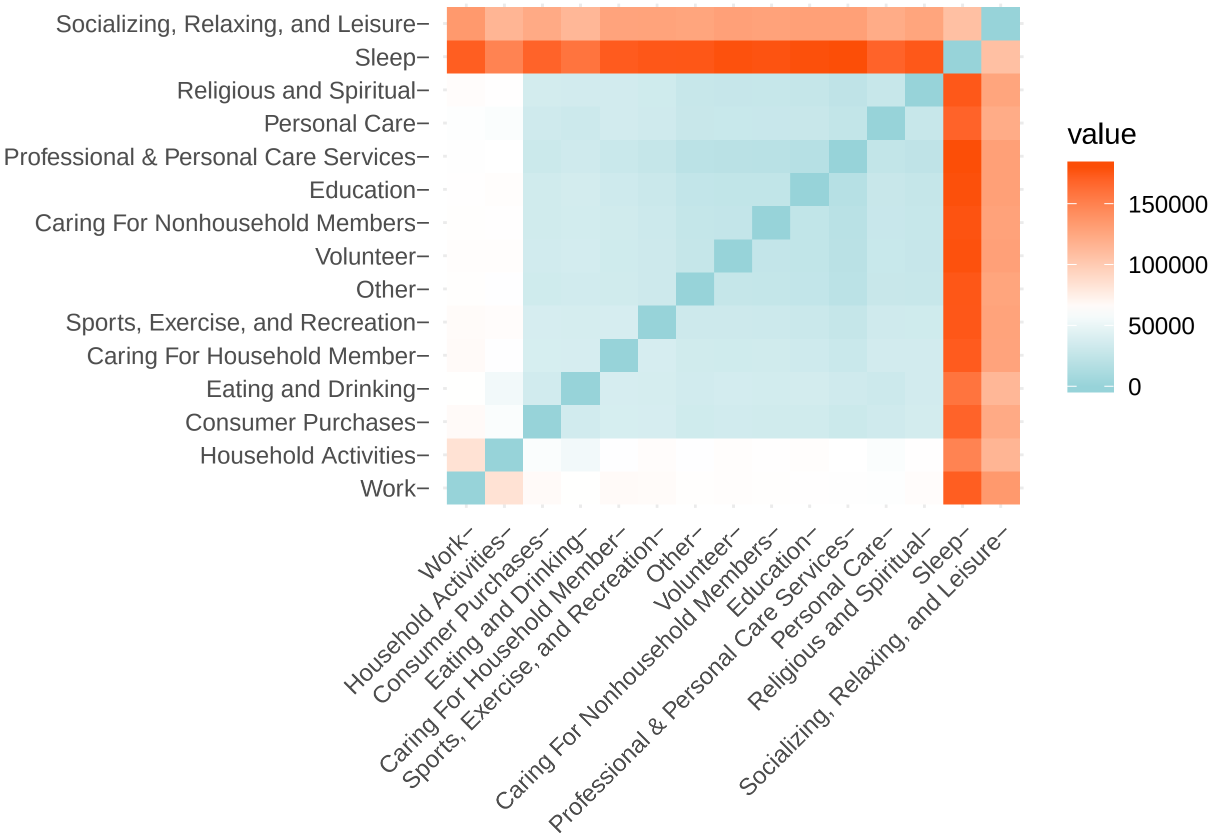
```
# description of the data
atus_long %>%
  group_by(description) %>%
  summarize(Mean = mean(value),
            part_rate = mean(value > 0)) %>%
  arrange(desc(Mean)) %>%
  mutate(Type = 'Continuous') %>%
  janitor::adorn_totals() %>%
  mutate(Mean = round(Mean, 0),
         part_rate = round(part_rate, 2),
         part_rate = ifelse(part_rate == 6.34, '-', part_rate)) %>%
  select(Activity = description, Type, Mean, 'Participation rate' = part_rate) %>%
  pander::pander(justify = c("left", "left", 'right', 'right'))
```

Activity	Type	Mean	Participation rate
Sleep	Continuous	556	1
Socializing, Relaxing, and Leisure	Continuous	351	0.96
Household Activities	Continuous	137	0.8
Eating and Drinking	Continuous	80	0.95
Work	Continuous	76	0.23
Consumer Purchases	Continuous	49	0.45
Personal Care	Continuous	46	0.76
Caring For Household Member	Continuous	34	0.26
Sports, Exercise, and Recreation	Continuous	26	0.18
Religious and Spiritual	Continuous	24	0.17
Other	Continuous	22	0.28
Caring For Nonhousehold Members	Continuous	15	0.14
Volunteer	Continuous	12	0.07
Education	Continuous	8	0.04
Professional & Personal Care Services	Continuous	4	0.04
Total	-	1440	-

```r
# plot of demographic data
demographics %>%
  select(-c('ID', 'survey_weight')) %>%
  mutate_all(as.character) %>%
  pivot_longer(cols = everything()) %>%
  ggplot(aes(x = value)) +
  geom_histogram(stat = 'count') +
  facet_wrap(~ name, scales = 'free') +
  labs(title = 'Demographics',
       x = NULL,
       y = NULL)
```



```
# distance between categories
atus_long %>%
  pivot_wider(values_from = value, names_from = description) %>%
  select(-ID) %>%
  as.matrix() %>%
  t() %>%
  dist() %>%
  factoextra::fviz_dist(gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E07"))
```

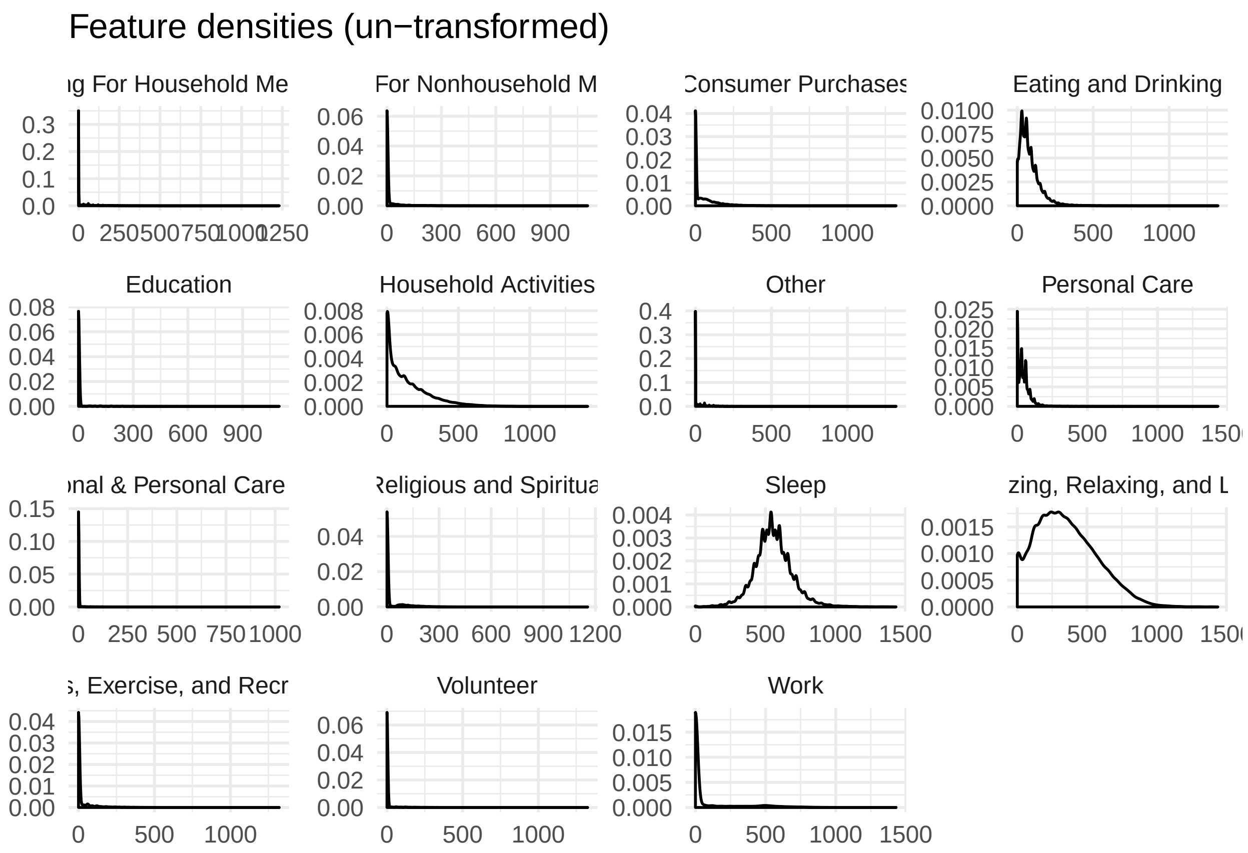


EDA

```
# check for NAs
dim(na.omit(atus_long)) == dim(atus_long)
```

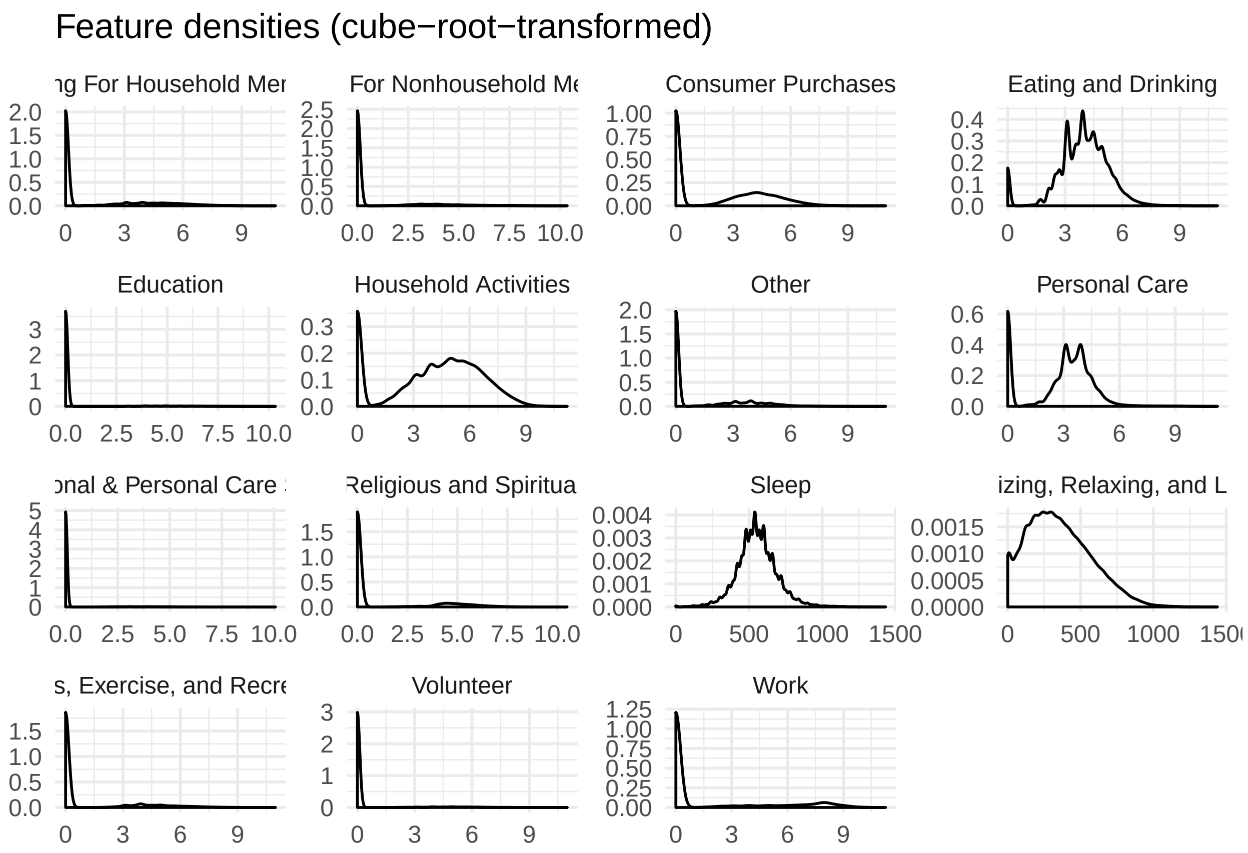
```
## [1] TRUE TRUE
```

```
# look at the data
atus_long %>%
  ggplot(aes(x = value)) +
  geom_density() +
  facet_wrap(~description, scales = "free") +
  labs(title = 'Feature densities (un-transformed)',
       x = NULL,
       y = NULL)
```



Transformations

```r
# log transform the data
cats_to_cube <- unique(atus_long$description)[
  !(unique(atus_long$description) %in% c('Socializing, Relaxing, and Leisure', 'Sleep'))]
atus_cube_root <- atus_long %>%
  filter(description %in% cats_to_cube) %>%
  group_by(description) %>%
  mutate(value = value^(1/3)) %>%
  ungroup() %>%
  bind_rows(
    atus_long %>%
      filter(!(description %in% cats_to_cube))
  )

atus_cube_root %>%
  ggplot(aes(x = value)) +
  geom_density() +
  facet_wrap( ~ description, scales = "free") +
  labs(title = 'Feature densities (cube-root-transformed)',
       x = NULL,
       y = NULL)
```



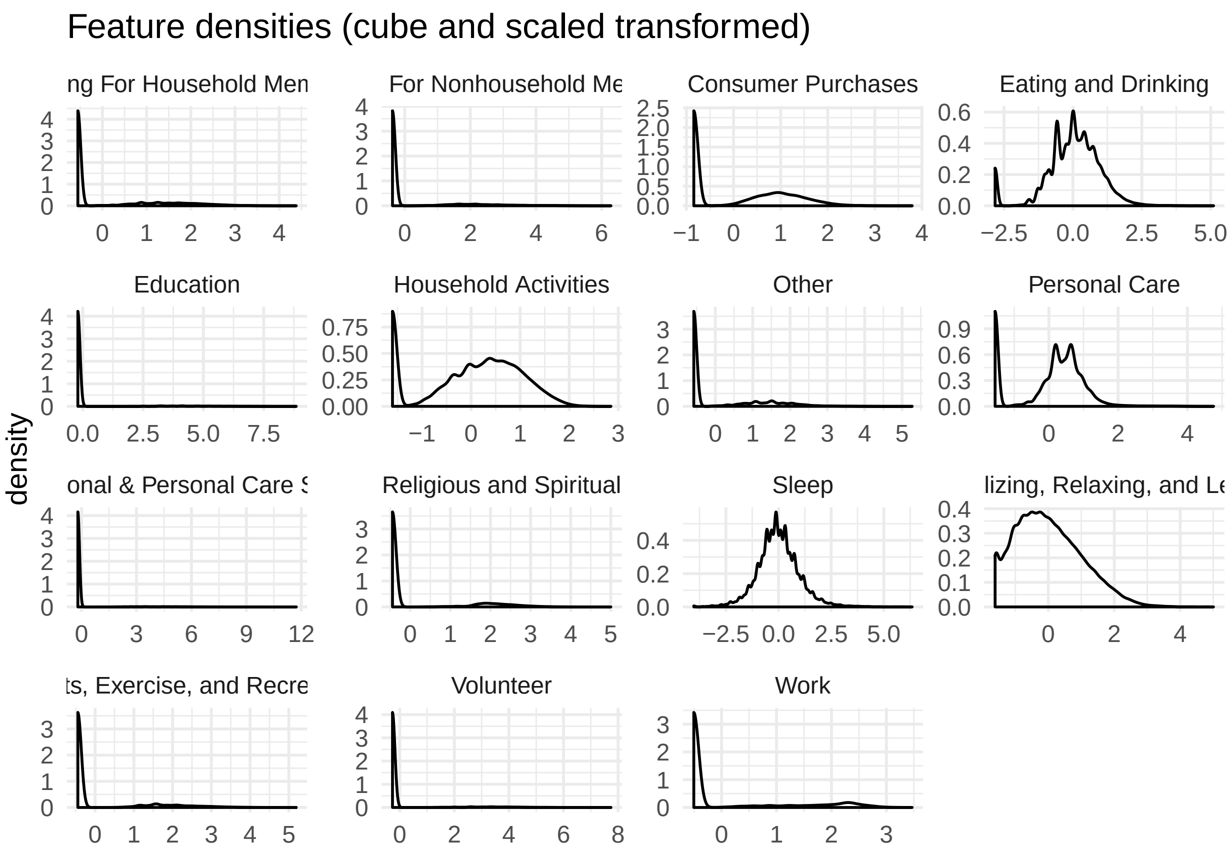
```r
# check spread of the data
atus_cube_root %>%
  group_by(description) %>%
  summarize(variance = var(value)) %>%
  pander::pander(justify = c('left', 'right'), round = 2)
```

description	variance
Caring For Household Member	4.7
Caring For Nonhousehold Members	2.41
Consumer Purchases	5.6
Eating and Drinking	1.92
Education	1.3
Household Activities	6.33
Other	3.52
Personal Care	3.19
Professional & Personal Care Services	0.71
Religious and Spiritual	3.71
Sleep	19103
Socializing, Relaxing, and Leisure	47226
Sports, Exercise, and Recreation	3.79
Volunteer	1.88
Work	8.04

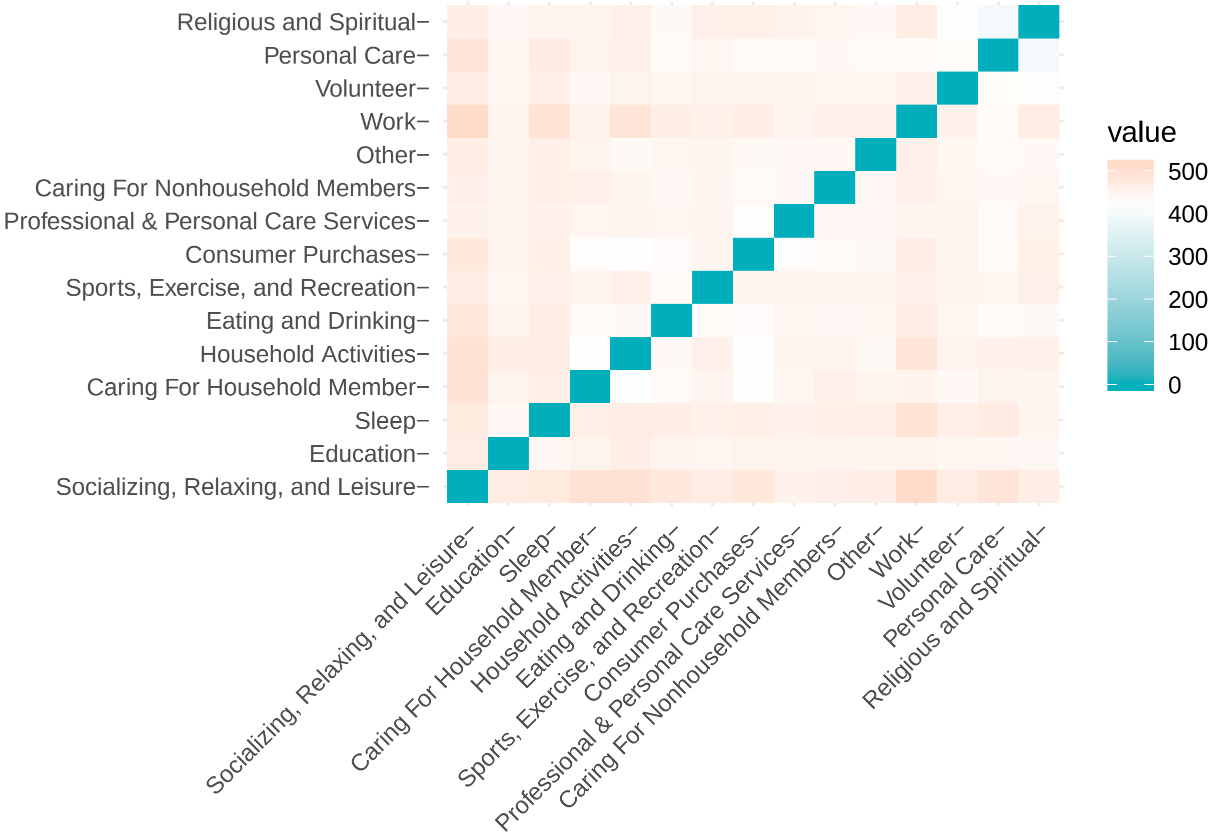
```
# scale the data ??
atus_scaled <- atus_cube_root %>%
  group_by(description) %>%
  mutate(value = scale(value)) %>%
  ungroup()

# check spread of the data
atus_scaled %>%
  group_by(description) %>%
  summarize(variance = var(value)) %>%
  pander::pander(justify = c('left', 'right'), round = 2)
```

description	variance
Caring For Household Member	1
Caring For Nonhousehold Members	1
Consumer Purchases	1
Eating and Drinking	1
Education	1
Household Activities	1
Other	1
Personal Care	1
Professional & Personal Care Services	1
Religious and Spiritual	1
Sleep	1
Socializing, Relaxing, and Leisure	1
Sports, Exercise, and Recreation	1
Volunteer	1
Work	1

```
# look at the data
atus_scaled %>%
  ggplot(aes(x = value)) +
  geom_density() +
  facet_wrap(~description, scales = "free") +
  labs(title = 'Feature densities (cube and scaled transformed)',
       x = NULL)
```



```r
# pivot wider and turn into matrix
atus_wide <- atus_scaled %>%
  pivot_wider(values_from = value, names_from = description) %>%
  select(-ID) %>%
  as.matrix()

# distance between categories
atus_wide %>%
  t() %>%
  dist() %>%
  factoextra::fviz_dist(gradient = list(low = "#00AFBB", mid = "white", high = "#FC4E07"))
```



PCA

```
# run PCA
atus_pca <- prcomp(atus_wide)
summary(atus_pca)$importance %>%
    pander::pander(justify = c('left', rep('right', 15)), round = 2)
```

Table 4: Table continues below

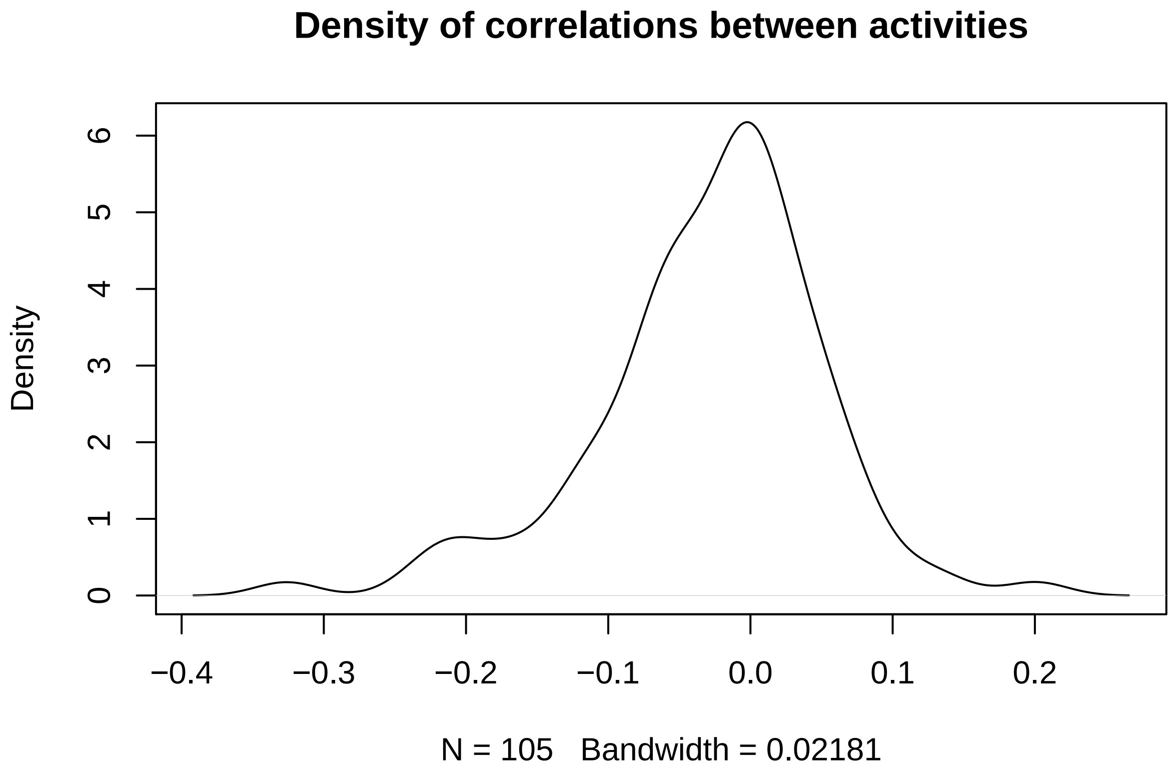
	PC1	PC2	PC3	PC4	PC5	PC6	PC7
Standard deviation	1.28	1.17	1.13	1.09	1.05	1.03	1
Proportion of Variance	0.11	0.09	0.09	0.08	0.07	0.07	0.07
Cumulative Proportion	0.11	0.2	0.29	0.37	0.44	0.51	0.58

	PC8	PC9	PC10	PC11	PC12	PC13	PC14	PC15
Standard deviation	0.99	0.98	0.97	0.94	0.93	0.92	0.84	0.42
Proportion of Variance	0.07	0.06	0.06	0.06	0.06	0.06	0.05	0.01
Cumulative Proportion	0.64	0.71	0.77	0.83	0.89	0.94	0.99	1

```
pca_plot <- atus_pca$x[ , 1:3]
# rgl::plot3d(pca_plot)
```

Resampling the data using survey weights

```
# function to scale [0.1]
scale_01 <- function(x) (x - min(x)) / (max(x) - min(x))

# sample using survey weights
total_rows <- nrow(atus_wide)
sample_size <- 10000
rows_to_keep <- sample(1:total_rows, size = sample_size, prob = scale_01(demographics$survey_weight), r
IDs_kept <- atus_scaled %>%
  pivot_wider(values_from = value, names_from = description) %>%
  select(ID) %>%
  .[rows_to_keep,]
atus_resampled <- atus_wide[rows_to_keep,]

# pairs plot of resampled data
# as_tibble(atus_resampled) %>%
#   GGally::ggpairs(mapping = aes(alpha = 0.2))

# correlations
corr_matrix <- cor(atus_resampled)
corr_matrix[lower.tri(corr_matrix)] %>% density() %>% plot(main = 'Density of correlations between acti
```



```
# run PCA
atus_pca <- prcomp(atus_resampled)
summary(atus_pca)$importance %>%
    pander::pander(justify = c('left', rep('right', 15)), round = 2)
```

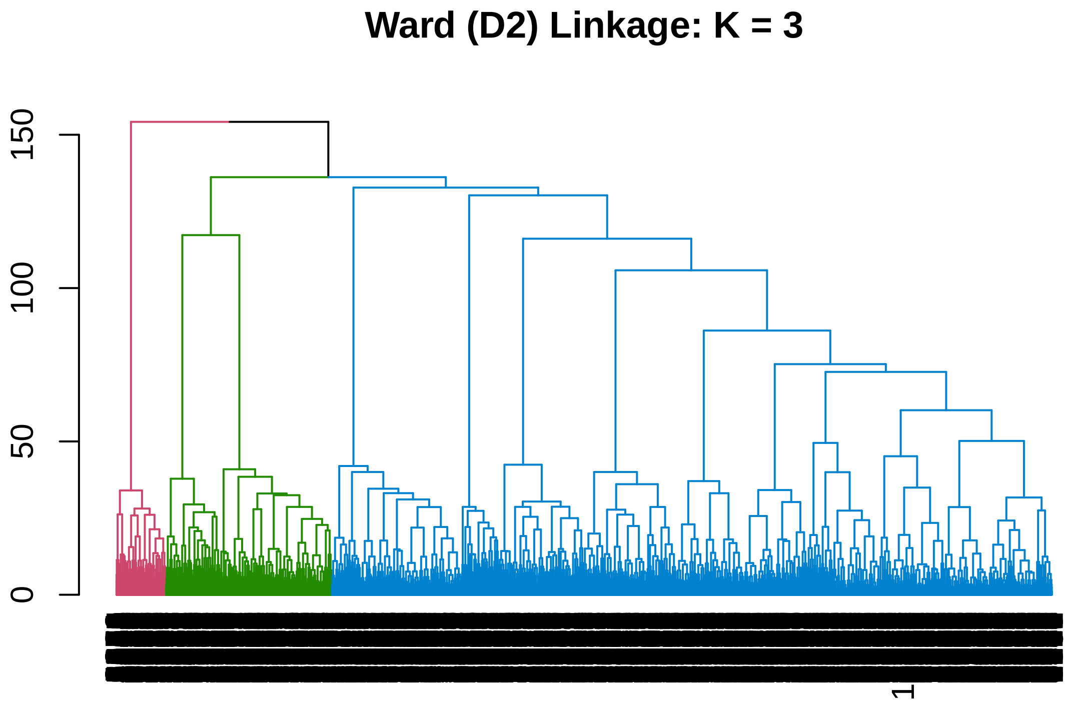
Table 6: Table continues below

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10	PC11	PC12	PC13	PC14	PC15
Standard deviation	1.28	1.21	1.15	1.1	1.07	1.04	1.02	0.99	0.98	0.96	0.93	0.92	0.87	0.82	0.41
Proportion of Variance	0.11	0.1	0.09	0.08	0.08	0.07	0.07	0.07	0.06	0.06	0.06	0.06	0.05	0.04	0.01
Cumulative Proportion	0.11	0.2	0.29	0.37	0.45	0.52	0.59	0.66	0.72	0.78	0.84	0.89	0.94	0.99	1

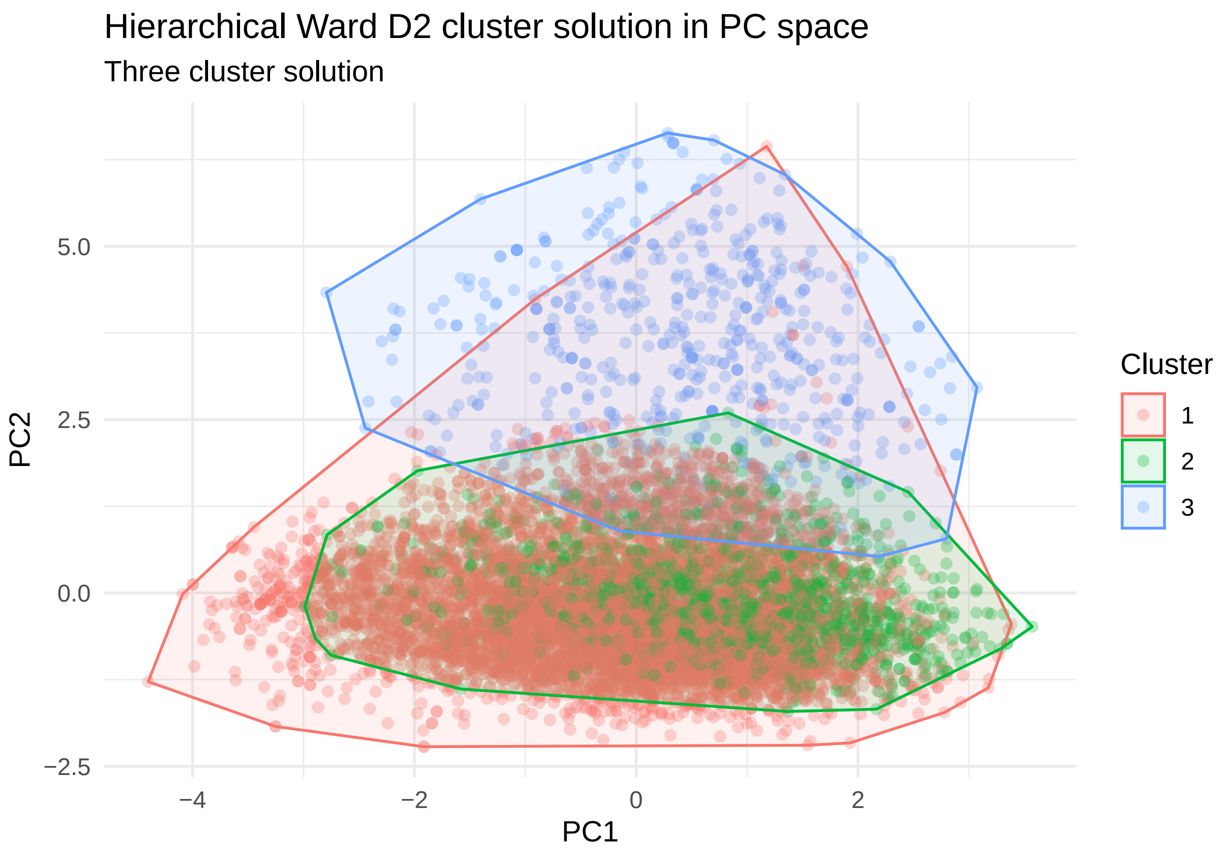
Clustering

Hierarchical cluster

```
# distance matrix for features
dist_sc <- dist(atus_resampled, method = 'euclidean')

# try single, centroid, and ward (D2) linkage hier clustering
# hcl_single <- hclust(d = dist_sc, method = 'single')
# hcl_centroid <- hclust(d = dist_sc, method = 'centroid')
hcl_ward <- hclust(d = dist_sc, method = 'ward.D2')


library(dendextend)

# dev.off()
# par(mfrow = c(3, 1))
# # nearest neighbors method
# plot(hcl_single, hang = -1, main = 'Single Linkage',
#      labels = FALSE, xlab = '', sub = '')
# # groups centroid
# plot(hcl_centroid, hang = -1, main = 'Centroid Linkage',
#      labels = FALSE, xlab = '',  sub = '')
# Ward's minimum variance method,
# with dissimilarities are squared before clustering
dend <- as.dendrogram(hcl_ward)
hcl_k <- 3
dend_col <- color_branches(dend, k = hcl_k)
plot(dend_col, main = paste0('Ward (D2) Linkage: K = ', hcl_k))
```



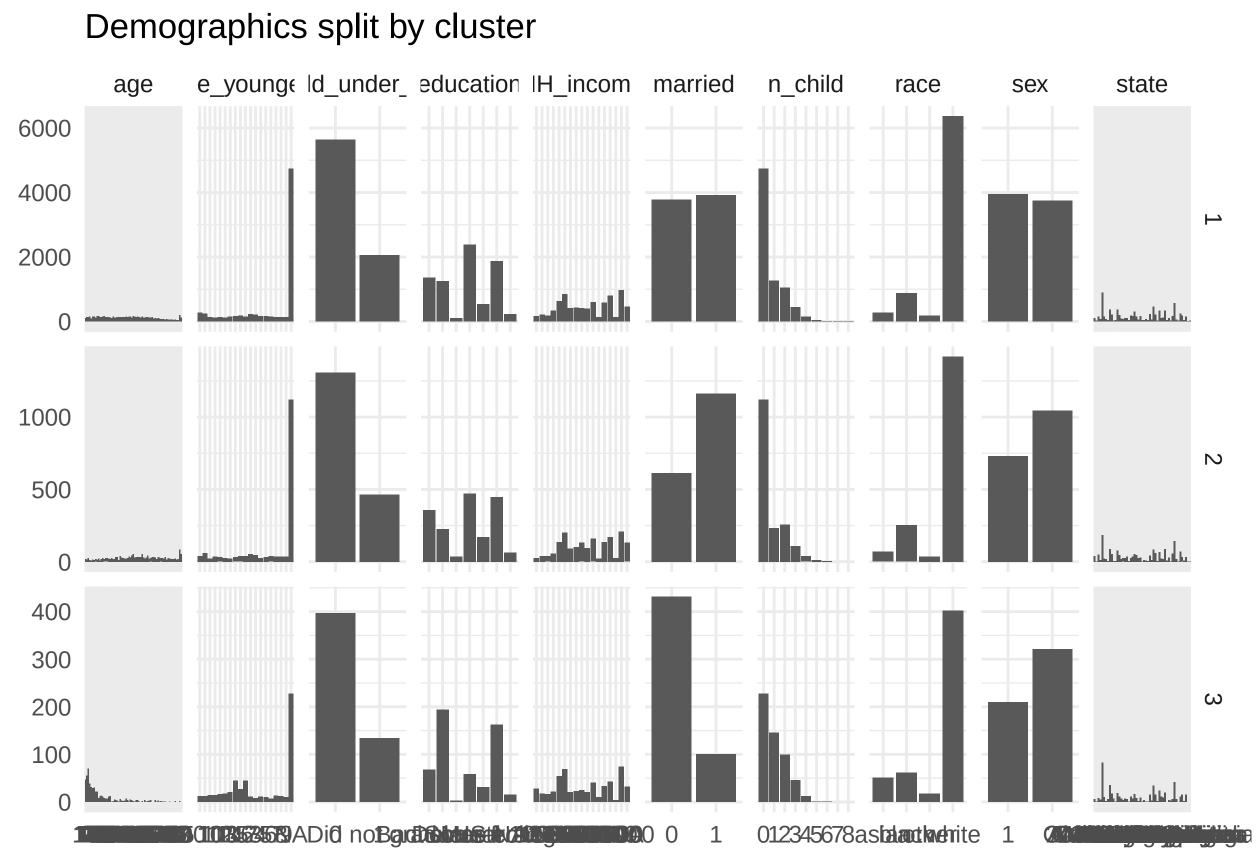
```
groupings <- cutree(hcl_ward, 3)
# plot the clusters in PC space
tmp_plot_data <- atus_pca$x[, 1:2] %>%
  as_tibble() %>%
  mutate(Cluster = as.factor(groupings))
tmp_plot_data %>%
  ggplot(aes(x = PC1, y = PC2, color = Cluster, fill = Cluster)) +
  geom_point(alpha = 0.3) +
  geom_polygon(data = get_cluster_polys(tmp_plot_data, x = PC1, y = PC2, cluster = 'Cluster'),
             alpha = 0.1) +
  labs(title = 'Hierarchical Ward D2 cluster solution in PC space',
       subtitle = 'Three cluster solution')
```



```r
rm(tmp_plot_data)
```

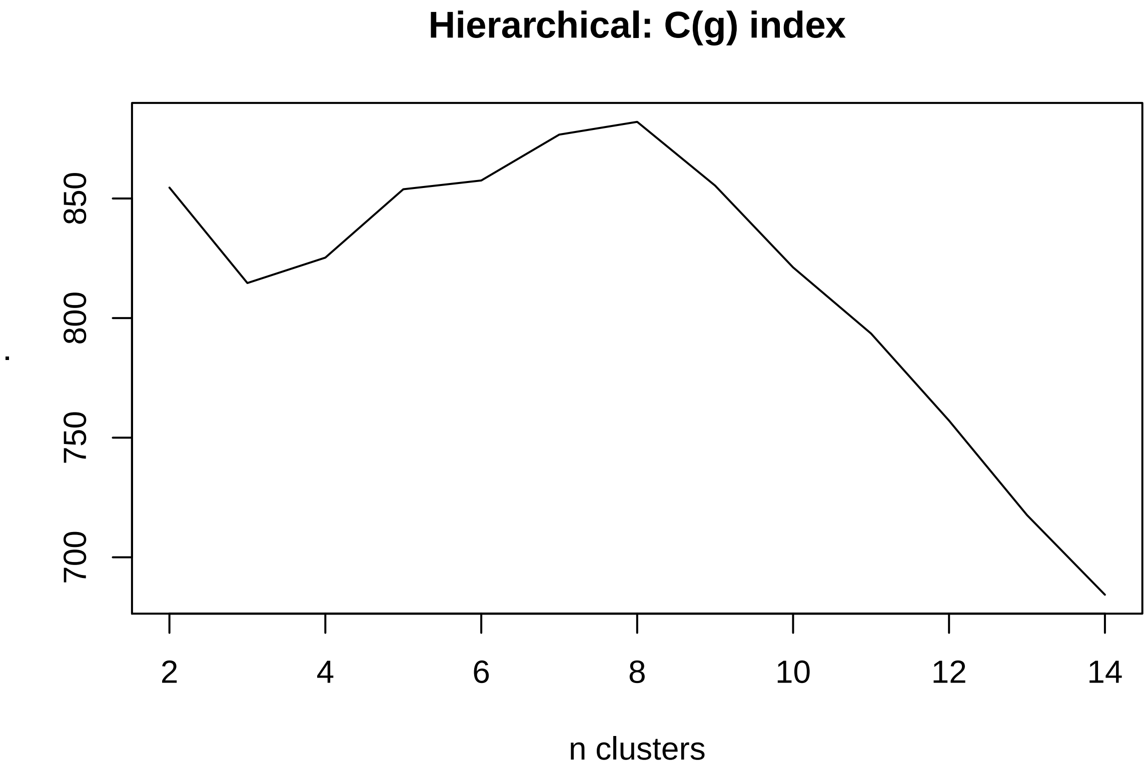
Examine the demographics through the clusters

```r
demographics %>%
  right_join(bind_cols(IDs_kept, group = groupings)) %>%
  select(-c('ID', 'survey_weight')) %>%
  mutate_all(as.character) %>%
  pivot_longer(cols = -group) %>%
  ggplot(aes(x = value)) +
  geom_histogram(stat = 'count') +
  facet_grid(group ~ name, scales = 'free') +
  labs(title = 'Demographics split by cluster',
       x = NULL,
       y = NULL)
```

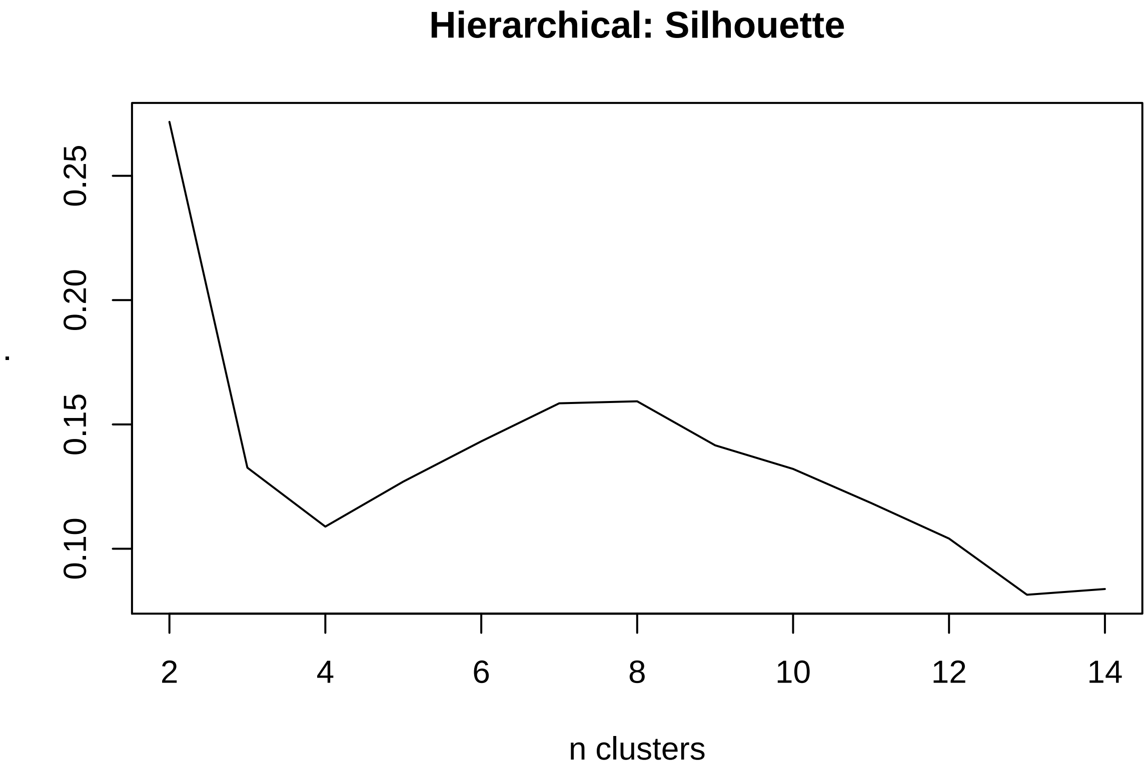


Optimizing hierarchical cluster sizes

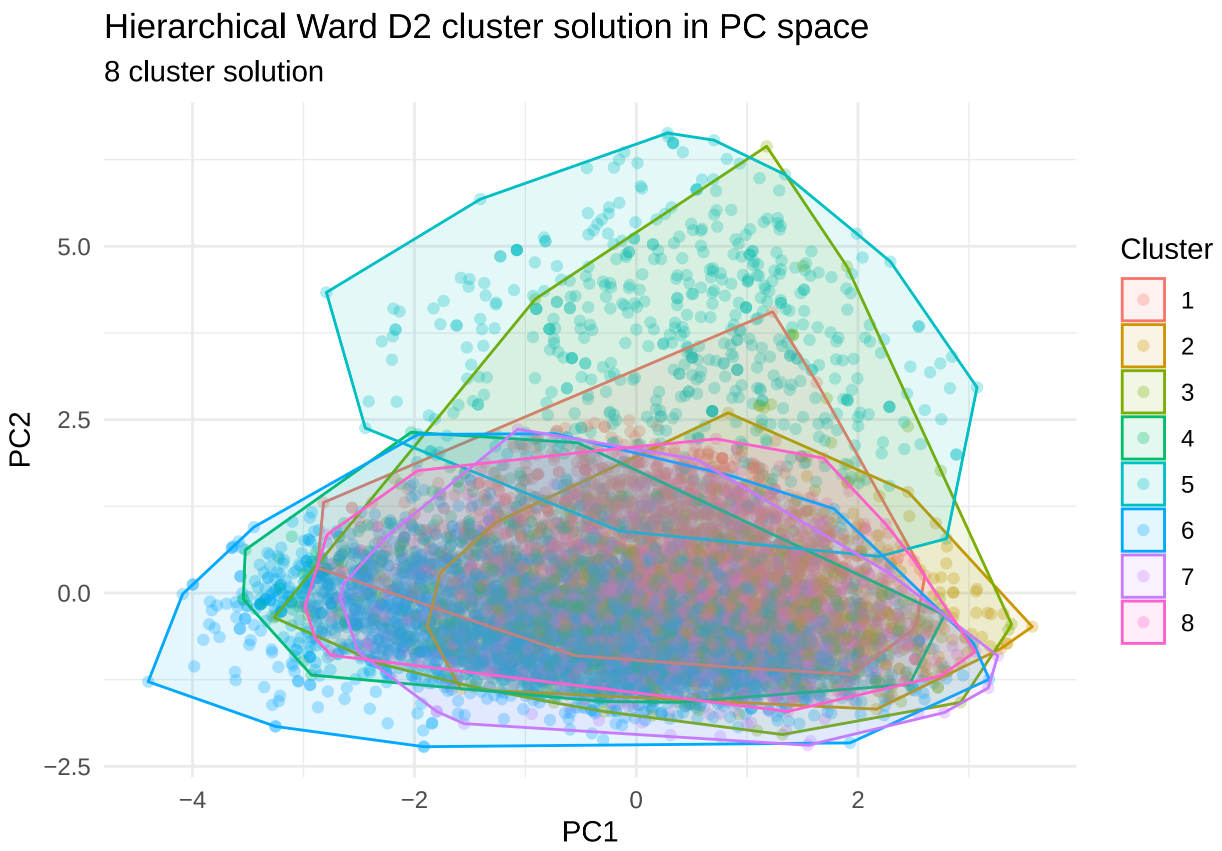
```r
# get optimal cluster sizes: c(g)
hcl_ch <- NbClust(
  data = atus_resampled,
  max.nc = 14,
  method = 'ward.D2',
  index = 'ch'
)
hcl_ch$All.index %>% plot(type = 'l', x = 2:14, main = 'Hierarchical: C(g) index', xlab = 'n clusters')
```



```
# get optimal cluster sizes: silhouette width
hcl_silhouette <- NbClust(
  data = atus_resampled,
  max.nc = 14,
  method = 'ward.D2',
  index = 'silhouette'
)
hcl_silhouette$All.index %>% plot(type = 'l', x = 2:14, main = 'Hierarchical: Silhouette', xlab = 'n clu
```



```
# plot the clusters in PC space
tmp_plot_data <- atus_pca$x[, 1:2] %>%
  as_tibble() %>%
  mutate(Cluster = as.factor(hcl_ch$Best.partition))
tmp_plot_data %>%
  ggplot(aes(x = PC1, y = PC2, color = Cluster, fill = Cluster)) +
  geom_point(alpha = 0.3) +
  geom_polygon(data = get_cluster_polys(tmp_plot_data, x = PC1, y = PC2, cluster = 'Cluster'),
               alpha = 0.1) +
  labs(title = 'Hierarchical Ward D2 cluster solution in PC space',
       subtitle = paste0(hcl_ch$Best.nc[[1]], ' cluster solution'))
```



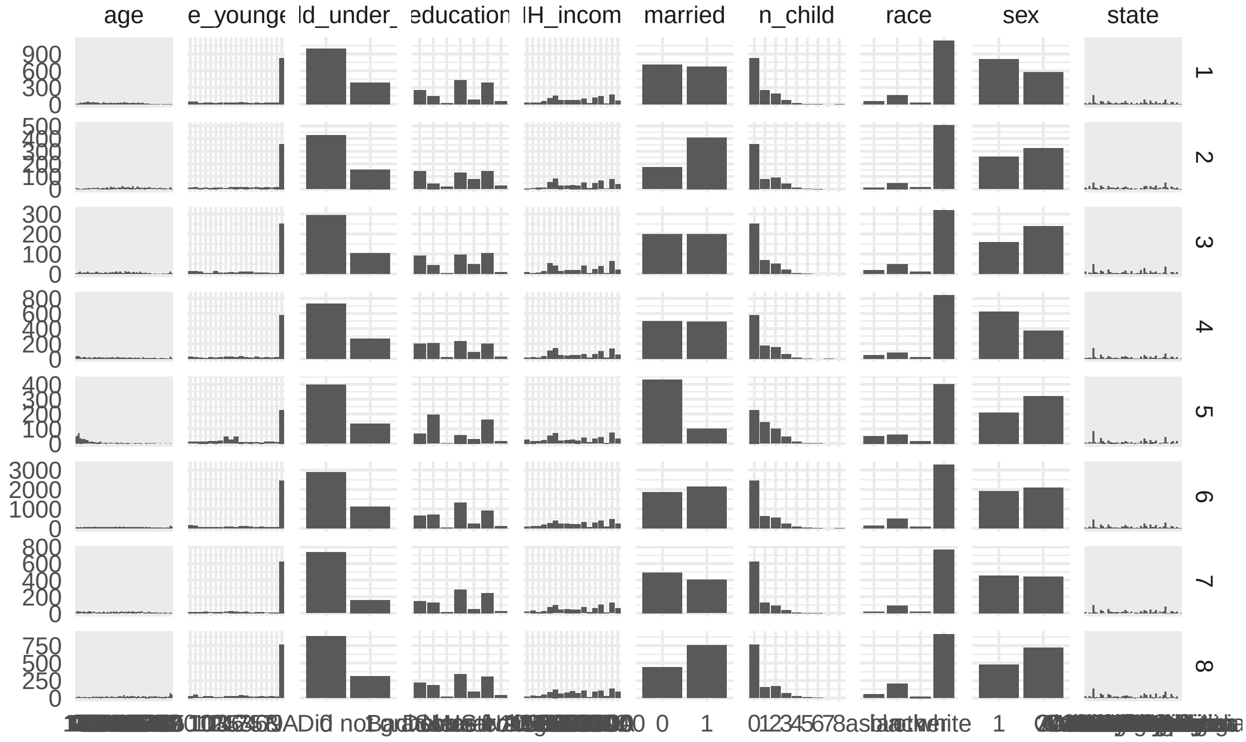
```r
rm(tmp_plot_data)
```

Examine the demographics through the clusters

```r
demographics %>%
  right_join(bind_cols(IDs_kept, cluster =as.factor(hcl_ch$Best.partition))) %>%
  select(-c('ID', 'survey_weight')) %>%
  mutate_all(as.character) %>%
  pivot_longer(cols = -cluster) %>%
  ggplot(aes(x = value)) +
  geom_histogram(stat = 'count') +
  facet_grid(cluster ~ name, scales = 'free') +
  labs(title = 'Demographics split by cluster',
       subtitle = 'Hierarchical Ward D2 cluster solution',
       x = NULL,
       y = NULL)
```

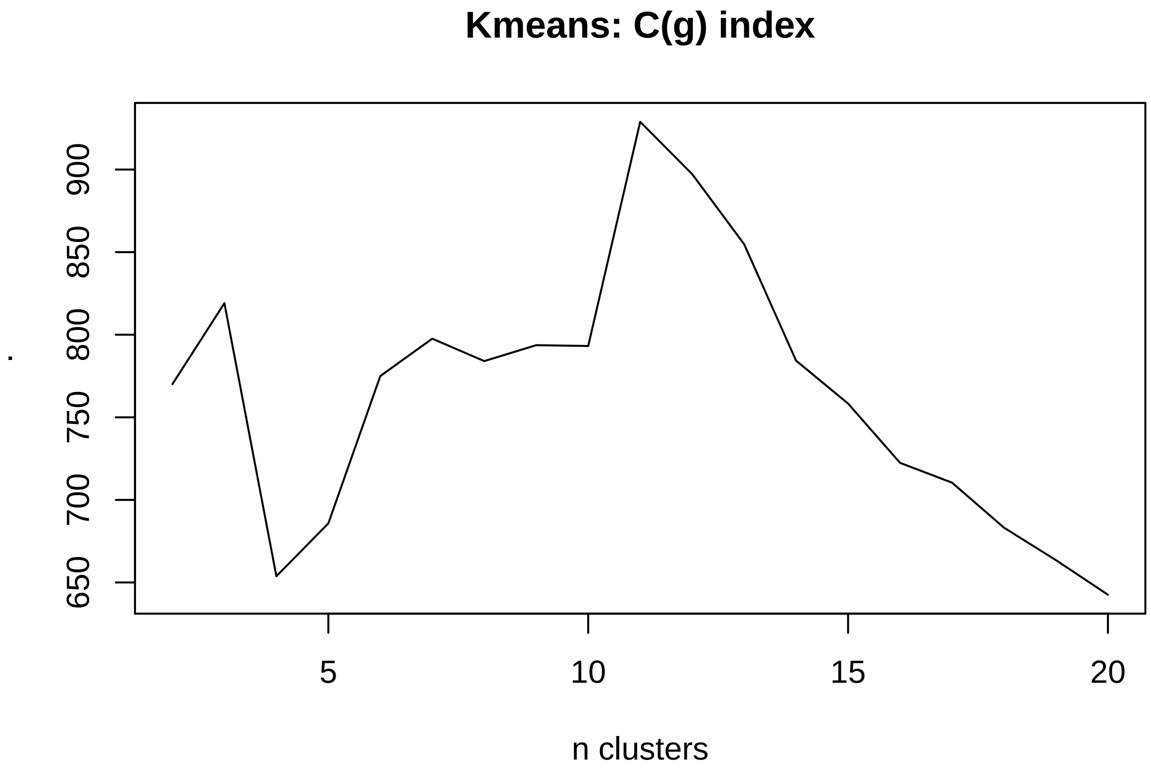
Demographics split by cluster

Hierarchical Ward D2 cluster solution

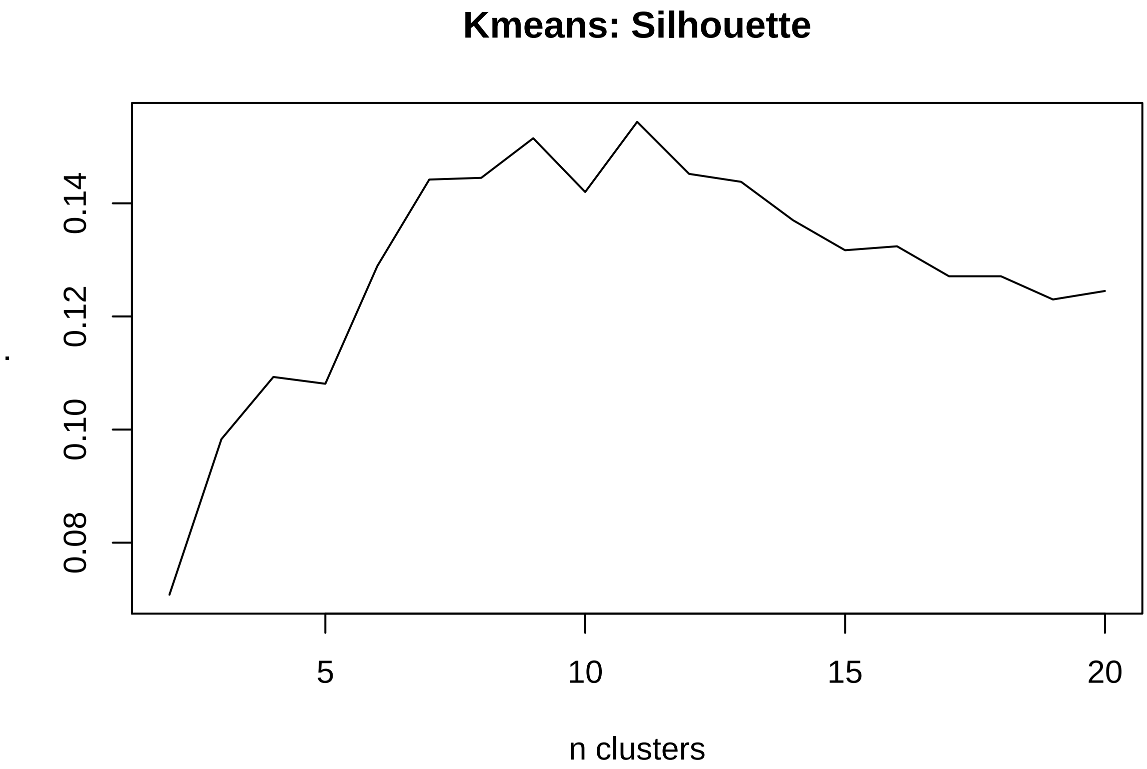


kmeans clustering

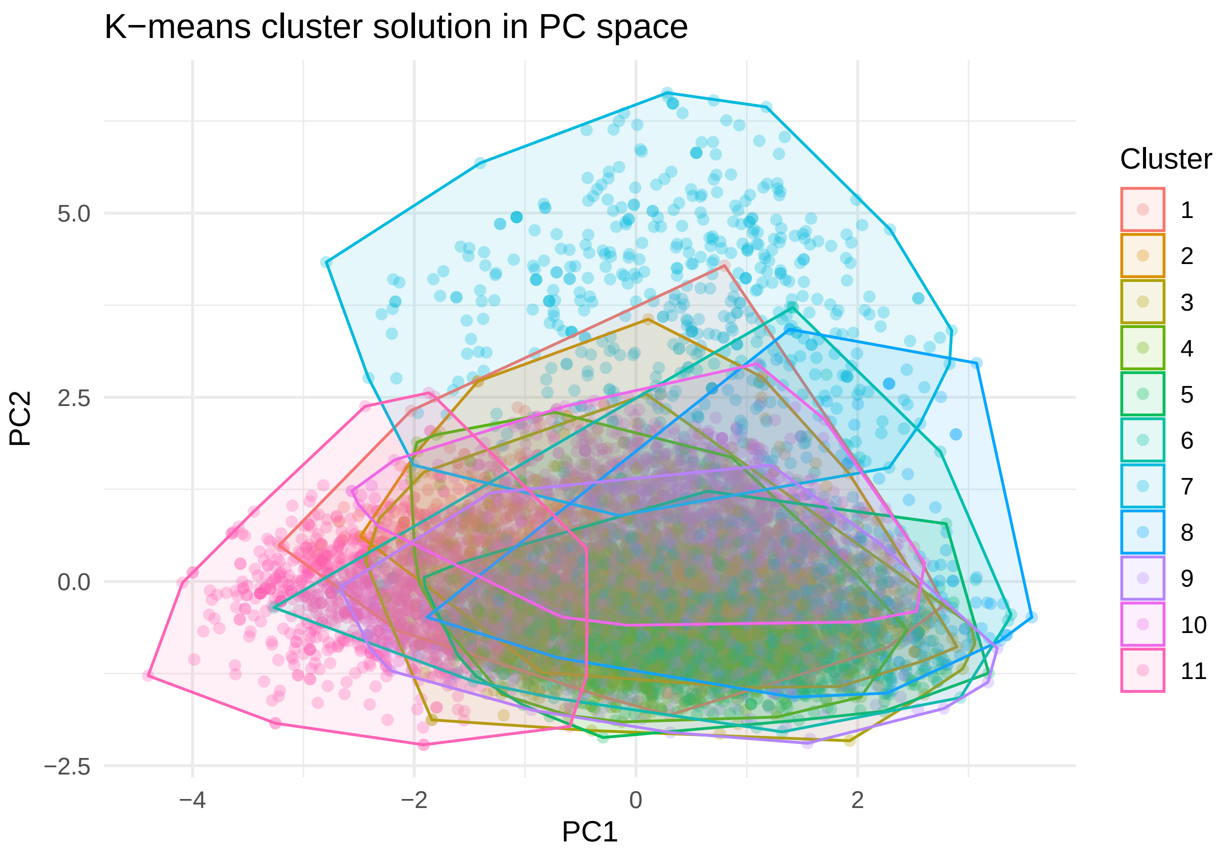
```
# get optimal cluster sizes
km_ch <- NbClust(
  data = atus_resampled,
  max.nc = 20,
  method = 'kmeans',
  index = 'ch'
)
km_ch$All.index %>% plot(type = 'l', x = 2:20, main = 'Kmeans: C(g) index', xlab = 'n clusters')
```



```
# get optimal cluster sizes
km_silhouette <- NbClust(
  data = atus_resampled,
  max.nc = 20,
  method = 'kmeans',
  index = 'silhouette'
)
km_silhouette$All.index %>% plot(type = 'l', x = 2:20, main = 'Kmeans: Silhouette', xlab = 'n clusters'
```



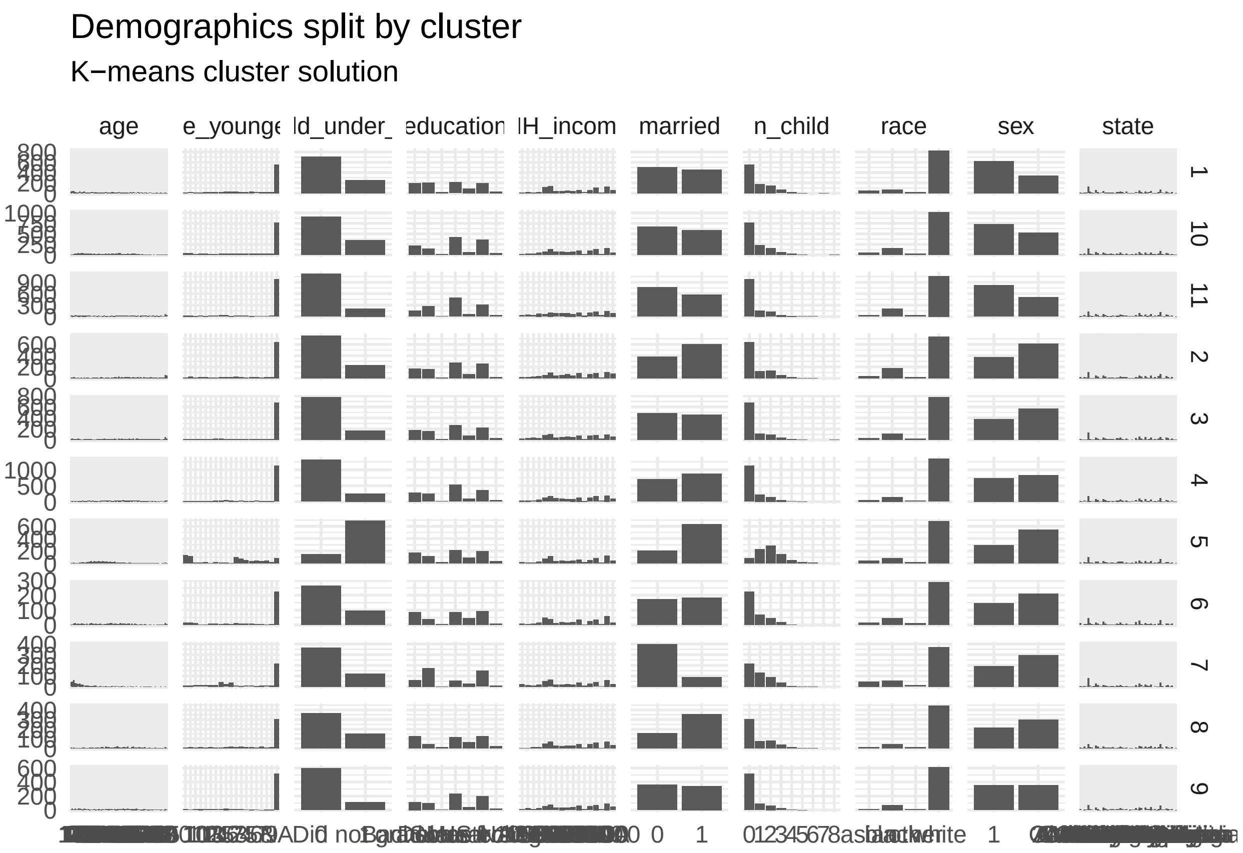
```
# run the final kmeans algo with optimal number of clusters
km_eleven <- kmeans(x = atus_resampled,
                    centers = km_ch$Best.nc[[1]],
                    nstart = 100,
                    iter.max = 30,
                    algorithm = 'Hartigan-Wong')

# plot the clusters in PC space
tmp_plot_data <- atus_pca$x[, 1:2] %>%
  as_tibble() %>%
  mutate(Cluster = as.factor(km_eleven$cluster))
tmp_plot_data %>%
  ggplot(aes(x = PC1, y = PC2, color = Cluster, fill = Cluster)) +
  geom_point(alpha = 0.3) +
  geom_polygon(data = get_cluster_polys(tmp_plot_data, x = PC1, y = PC2, cluster = 'Cluster'),
               alpha = 0.1) +
  labs(title = 'K-means cluster solution in PC space')
```



```
rm(tmp_plot_data)
```

Examine the demographics through the clusters

```
demographics %>%
  right_join(bind_cols(IDs_kept, cluster = as.factor(km_eleven$cluster))) %>%
  select(-c('ID', 'survey_weight')) %>%
  mutate_all(as.character) %>%
  pivot_longer(cols = -cluster) %>%
  ggplot(aes(x = value)) +
  geom_histogram(stat = 'count') +
  facet_grid(cluster ~ name, scales = 'free') +
  labs(title = 'Demographics split by cluster',
       subtitle = 'K-means cluster solution',
       x = NULL,
       y = NULL)
```



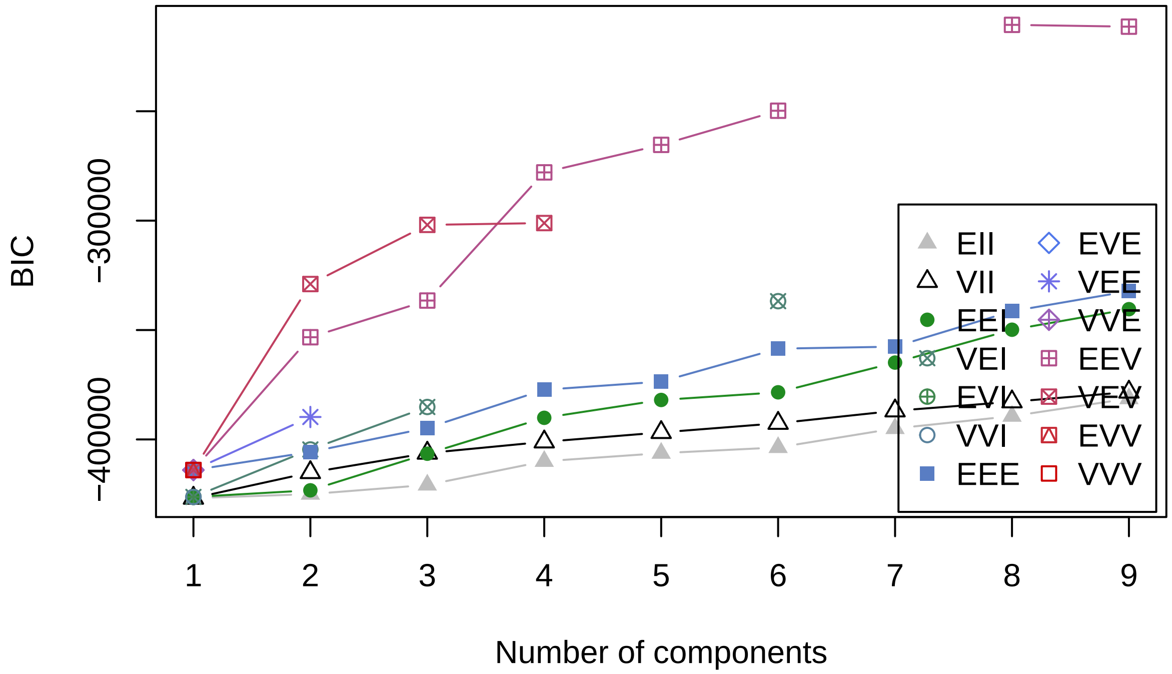
Kmeans gap statistic

Model based clustering

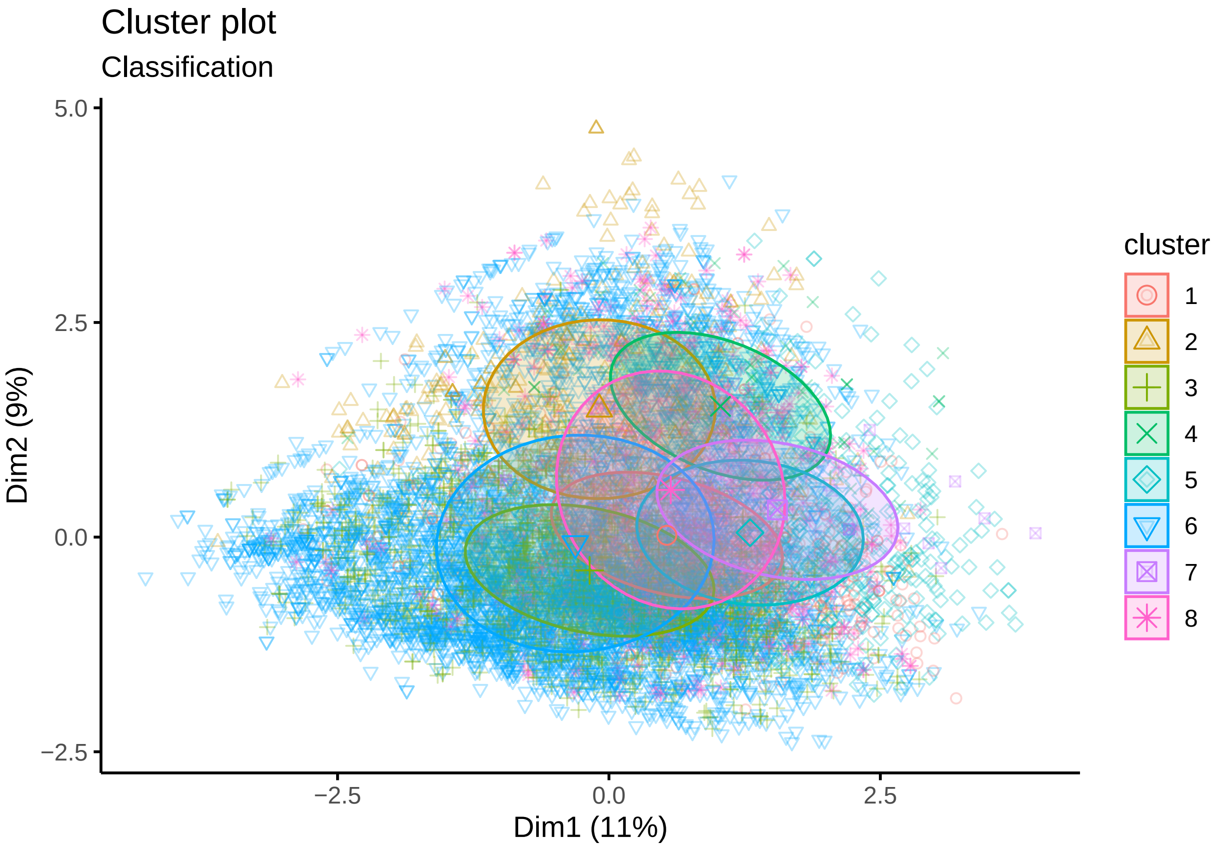
```r
# run model
mcl <- Mclust(atus_resampled)
summary(mcl)
```

```
## ----------------------------------------------------
## Gaussian finite mixture model fitted by EM algorithm
## ----------------------------------------------------
##
## Mclust EEV (ellipsoidal, equal volume and shape) model with 8 components:
##
##  log-likelihood     n  df       BIC     ICL
##       -100741.5 10000 982 -210527.5 -210728
##
## Clustering table:
##    1    2    3    4    5    6    7    8
##  989  406 1935   98  753 5206   42  571
```

```r
# plot
plot(mcl, what = "BIC")
```



```
factoextra::fviz_mclust(mcl, "classification", geom = "point", alpha = 0.3)
```

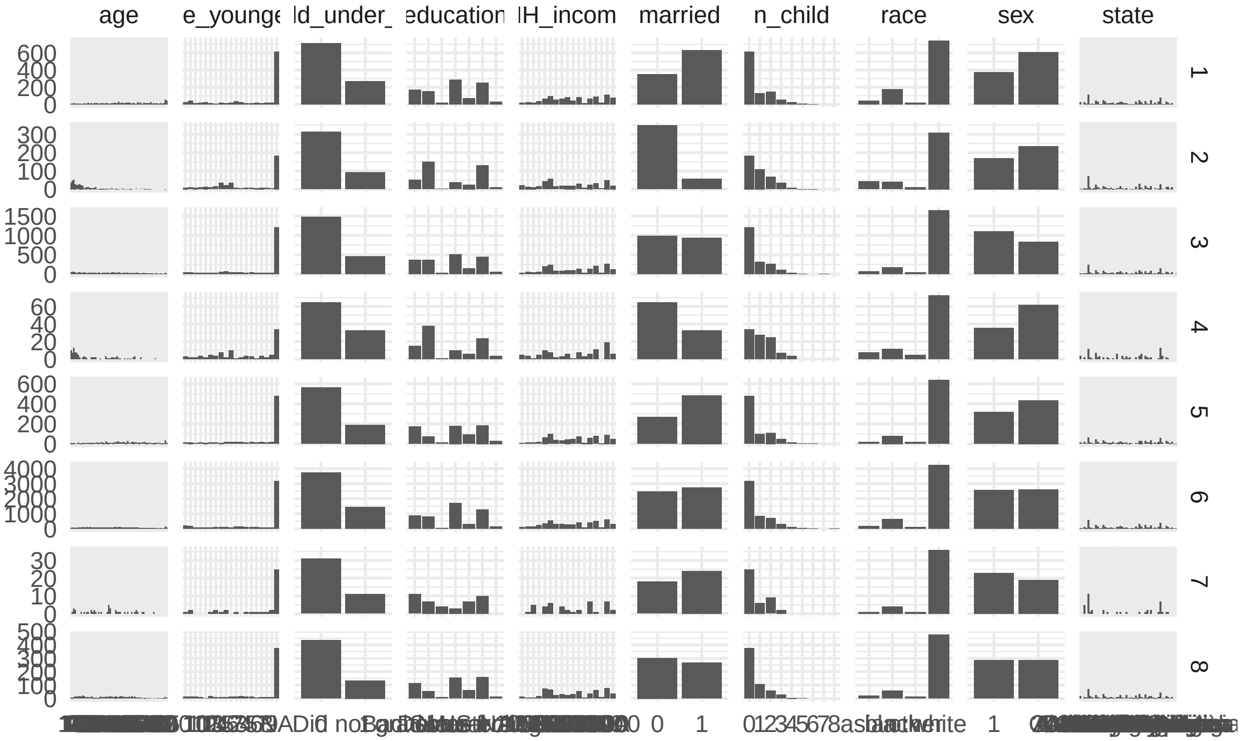


Examine the demographics through the clusters

```
demographics %>%
  right_join(bind_cols(IDs_kept, cluster = as.factor(mcl$classification))) %>%
  select(-c('ID', 'survey_weight')) %>%
  mutate_all(as.character) %>%
  pivot_longer(cols = -cluster) %>%
  ggplot(aes(x = value)) +
  geom_histogram(stat = 'count') +
  facet_grid(cluster ~ name, scales = 'free') +
  labs(title = 'Demographics split by cluster',
       subtitle = 'Model-based cluster solution',
       x = NULL,
       y = NULL)
```

Demographics split by cluster

Model−based cluster solution


age
e_younge
ld_under_
education
IH_incom
married
n_child
race
sex
state
asian
black
other
white