Clustering

Sar, North, Henry, Quinn

4/12/2022

```
library(ISLR)
library(dplyr)
```

```
## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'pillar'
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(readr)
```

```
## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'hms'
```

```
library(broom)
```

```
## Warning: package 'broom' was built under R version 4.1.2
```

```
library(ggplot2)
library(splines)
library(tidymodels)
```

```
## Registered S3 method overwritten by 'tune':
##   method                   from
##   required_pkgs.model_spec parsnip
```

```
## -- Attaching packages -------------------------------------- tidymodels 0.1.4 --
```

```
## v dials        0.0.10     v tibble       3.1.6
## v infer        1.0.0      v tidyr        1.1.4
## v modeldata    0.1.1      v tune         0.1.6
## v parsnip      0.1.7      v workflows    0.2.4
## v purrr        0.3.4      v workflowsets 0.1.0
## v recipes      0.1.17     v yardstick    0.0.9
## v rsample      0.1.1

## -- Conflicts ---------------------------------------- tidymodels_conflicts() --
## x purrr::discard()  masks scales::discard()
## x dplyr::filter()   masks stats::filter()
## x dplyr::lag()      masks stats::lag()
## x yardstick::spec() masks readr::spec()
## x recipes::step()   masks stats::step()
## * Search for functions across packages at https://www.tidymodels.org/find/
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(maps)
```

```
##
## Attaching package: 'maps'

## The following object is masked from 'package:purrr':
##
##     map
```

```
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

## The following objects are masked from 'package:yardstick':
##
##     precision, recall, sensitivity, specificity

## The following object is masked from 'package:purrr':
##
##     lift
```

```
tidymodels_prefer()
```

```
COVID_State <- read.csv("COVID - State - Daily.csv", na.strings = ".")

Employment_State <- read.csv("Employment - State - Daily.csv", na.strings = ".")

Mobility_State <- read.csv("Google Mobility - State - Daily.csv", na.strings = ".")

Spending_State <- read.csv("Affinity - State - Daily.csv", na.strings = ".")

regions <- read.csv("regions.csv")
fips <- state.fips
```

```
COVID_State$Date<-as.Date(with(COVID_State,paste(year,month,day,sep="-")),"%Y-%m-%d")

Employment_State$Date<-as.Date(with(Employment_State,paste(year,month,day,sep="-")),"%Y-%m-%d")

Mobility_State$Date<-as.Date(with(Mobility_State,paste(year,month,day,sep="-")),"%Y-%m-%d")

Spending_State$Date<-as.Date(with(Spending_State,paste(year,month,day,sep="-")),"%Y-%m-%d")

full_data <- merge(merge(merge(COVID_State, Employment_State, by=c("Date","statefips")), Mobility_State
```

```
## Warning in merge.data.frame(merge(merge(COVID_State, Employment_State, by =
## c("Date", : column names 'year.x', 'month.x', 'day.x', 'year.y', 'month.y',
## 'day.y' are duplicated in the result
```

```
head(full_data)
```

```
##         Date statefips year.x month.x day.x new_case_count new_death_count
## 1 2020-02-24         1   2020       2    24             NA              NA
## 2 2020-02-24        10   2020       2    24             NA              NA
## 3 2020-02-24        11   2020       2    24             NA              NA
## 4 2020-02-24        12   2020       2    24             NA              NA
## 5 2020-02-24        13   2020       2    24             NA              NA
## 6 2020-02-24        15   2020       2    24             NA              NA
##   case_count death_count vaccine_count fullvaccine_count booster_first_count
## 1         NA           NA            NA                NA                  NA
## 2         NA           NA            NA                NA                  NA
## 3         NA           NA            NA                NA                  NA
## 4         NA           NA            NA                NA                  NA
## 5         NA           NA            NA                NA                  NA
## 6         NA           NA            NA                NA                  NA
##   new_vaccine_count new_fullvaccine_count new_booster_first_count
## 1                NA                   NA                      NA
## 2                NA                   NA                      NA
## 3                NA                   NA                      NA
## 4                NA                   NA                      NA
## 5                NA                   NA                      NA
## 6                NA                   NA                      NA
##   new_test_count test_count hospitalized_count new_case_rate case_rate
## 1             NA         NA                 NA            NA        NA
```

```
## 2             NA         NA                 NA            NA        NA
## 3             NA         NA                 NA            NA        NA
## 4             NA         NA                 NA            NA        NA
## 5             NA         NA                 NA            NA        NA
## 6             NA         NA                  0            NA        NA
##   new_death_rate death_rate new_test_rate test_rate new_vaccine_rate
## 1             NA         NA            NA        NA               NA
## 2             NA         NA            NA        NA               NA
## 3             NA         NA            NA        NA               NA
## 4             NA         NA            NA        NA               NA
## 5             NA         NA            NA        NA               NA
## 6             NA         NA            NA        NA               NA
##   vaccine_rate new_fullvaccine_rate fullvaccine_rate new_booster_first_rate
## 1           NA                   NA               NA                    NA
## 2           NA                   NA               NA                    NA
## 3           NA                   NA               NA                    NA
## 4           NA                   NA               NA                    NA
## 5           NA                   NA               NA                    NA
## 6           NA                   NA               NA                    NA
##   booster_first_rate hospitalized_rate year.y month.y day.y      emp emp_incq1
## 1                 NA               NA   2020       2    24  0.01580   0.00751
## 2                 NA               NA   2020       2    24  0.00537  -0.02670
## 3                 NA               NA   2020       2    24       NA        NA
## 4                 NA               NA   2020       2    24  0.00448  -0.00263
## 5                 NA               NA   2020       2    24  0.00532  -0.00537
## 6                 NA                0   2020       2    24 -0.03530  -0.07190
##   emp_incq2 emp_incq3 emp_incq4 emp_incmiddle emp_incbelowmed emp_incabovemed
## 1   0.02320   0.01680        NA       0.01960        0.013600          0.0183
## 2   0.00570   0.01680    0.0242       0.01170       -0.011400          0.0206
## 3        NA        NA        NA            NA              NA              NA
## 4  -0.00458   0.01070    0.0164       0.00324       -0.003550          0.0133
## 5   0.00520   0.00873    0.0140       0.00710       -0.000838          0.0114
## 6  -0.04920  -0.00520        NA      -0.02980       -0.058300         -0.0112
##    emp_ss40 emp_ss60 emp_ss65 emp_ss70 year.x month.x day.x
## 1  0.001540 -0.00399  0.05300 -0.01620   2020       2    24
## 2  0.015400  0.01340  0.01030 -0.05550   2020       2    24
## 3        NA       NA       NA       NA   2020       2    24
## 4 -0.002320  0.00134  0.00576  0.01620   2020       2    24
## 5 -0.000237  0.00168  0.00889  0.00964   2020       2    24
## 6  0.054800       NA       NA -0.01530   2020       2    24
##   gps_retail_and_recreation gps_grocery_and_pharmacy gps_parks
## 1                   0.00286                -0.00714    0.0557
## 2                   0.03710                 0.01290    0.2340
## 3                  -0.01140                -0.03290    0.1400
## 4                   0.02710                 0.00714    0.0943
## 5                  -0.00571                -0.02290    0.0186
## 6                   0.01140                -0.00571    0.0814
##   gps_transit_stations gps_workplaces gps_residential gps_away_from_home year.y
## 1              0.06000        0.01290         0.00857           -0.00798   2020
## 2              0.07000        0.02860        -0.00571            0.00850   2020
## 3              0.00571       -0.01430         0.00714           -0.00492   2020
## 4              0.03430        0.01000         0.00143           -0.00138   2020
## 5              0.01710       -0.01140         0.01000           -0.00781   2020
## 6              0.02570        0.00714         0.00143           -0.00049   2020
```

```
##   month.y day.y freq spend_all spend_aap spend_acf spend_aer spend_apg
## 1       2    24    d   -0.0198   -0.1320   -0.0220   -0.1000   -0.0810
## 2       2    24    d   -0.0461    0.1130   -0.0279   -0.6280    0.4140
## 3       2    24    d    0.0192   -0.1280   -0.0113    0.0740   -0.0855
## 4       2    24    d   -0.0452   -0.0847   -0.0493   -0.1020   -0.0675
## 5       2    24    d   -0.0163   -0.0321   -0.0334    0.0287   -0.0308
## 6       2    24    d   -0.0504   -0.1210   -0.0447   -0.1650   -0.0851
##   spend_durables spend_nondurables spend_grf spend_gen spend_hic spend_hcs
## 1        -0.0317          -0.04750   -0.0223  -0.01050  -0.06180  -0.07310
## 2         0.0208           0.13400   -0.0284   0.63600   0.13400  -0.01060
## 3         0.0311          -0.00364    0.0294   0.00856   0.59500   0.02630
## 4        -0.0492          -0.04720   -0.0468  -0.03810  -0.08320   0.00175
## 5        -0.0164          -0.02450   -0.0110  -0.03000  -0.00361  -0.02010
## 6        -0.0118          -0.04380   -0.0173  -0.04770   0.16600  -0.08730
##   spend_inpersonmisc spend_remoteservices spend_sgh spend_tws
## 1             0.0062              0.02110   -0.0453   -0.1020
## 2            -0.1380             -0.15500   -0.1540   -0.0929
## 3             0.2100             -0.03610   -0.1230   -0.1360
## 4            -0.0815             -0.04600   -0.0426   -0.1030
## 5            -0.0658             -0.00774    0.0940   -0.1060
## 6            -0.0645             -0.04000   -0.2270   -0.0909
##   spend_retail_w_grocery spend_retail_no_grocery spend_all_incmiddle
## 1               -0.03910                -0.0459           -0.02970
## 2                0.10200                 0.1560           -0.06480
## 3               -0.00169                -0.0124           -0.06430
## 4               -0.04390                -0.0421           -0.03880
## 5               -0.01640                -0.0176           -0.01870
## 6               -0.03610                -0.0498            0.00268
##   spend_all_q1 spend_all_q2 spend_all_q3 spend_all_q4 provisional
## 1      -0.0158      -0.0717     0.036100     0.009840           0
## 2       0.2240      -0.0565    -0.068700    -0.016000           0
## 3      -0.0265      -0.5850    -0.047300     0.039400           0
## 4      -0.0677      -0.0420    -0.035100    -0.035700           0
## 5      -0.0386      -0.0234    -0.015600    -0.000937           0
## 6           NA       0.0134     0.000257    -0.076700           0
```

```
full_data1 <- full_data %>%
  select(-year.x, -month.x, -day.x, - year.y, -month.y, -day.y, -year.x )

regions <- regions%>%
  inner_join(fips, by=c("State.Code"="abb"))

#created dataset with the fips code
full_cut <- full_data1 %>%
  filter(Date > "2020-04-13")%>%
  select(statefips, Date, gps_away_from_home, case_rate, hospitalized_rate, spend_remoteservices, spend_
  left_join(regions, by=c("statefips"="fips"))

full_cut <- full_cut %>%
  select(statefips, Date, gps_away_from_home, case_rate, hospitalized_rate, spend_remoteservices, spend_

full_cut <- full_cut[,-1]
full_cut <- full_cut %>% na.omit() #there are 6 missing values in two variables
full_cut <- full_cut %>%
```

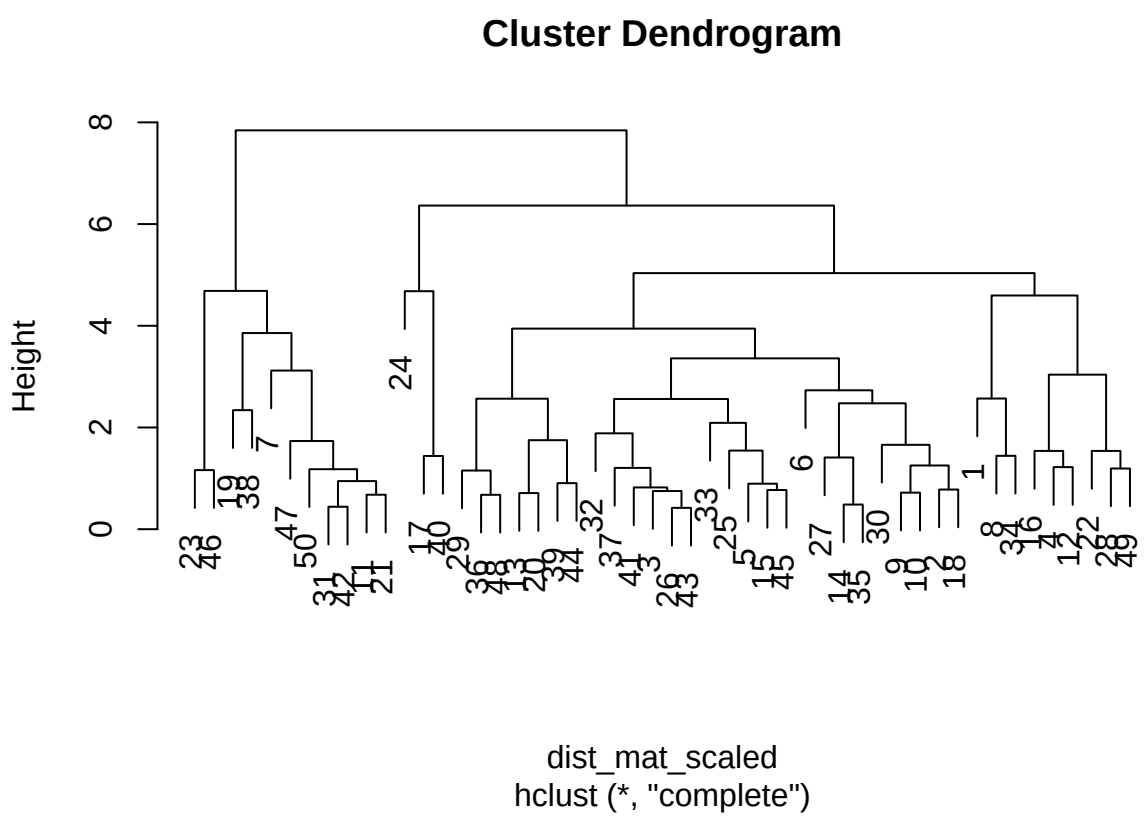
```
  mutate(Region = factor(Region)) %>% #make sure outcome is factor
  mutate(across(where(is.character), as.factor))
```

Hierarchical Clustering

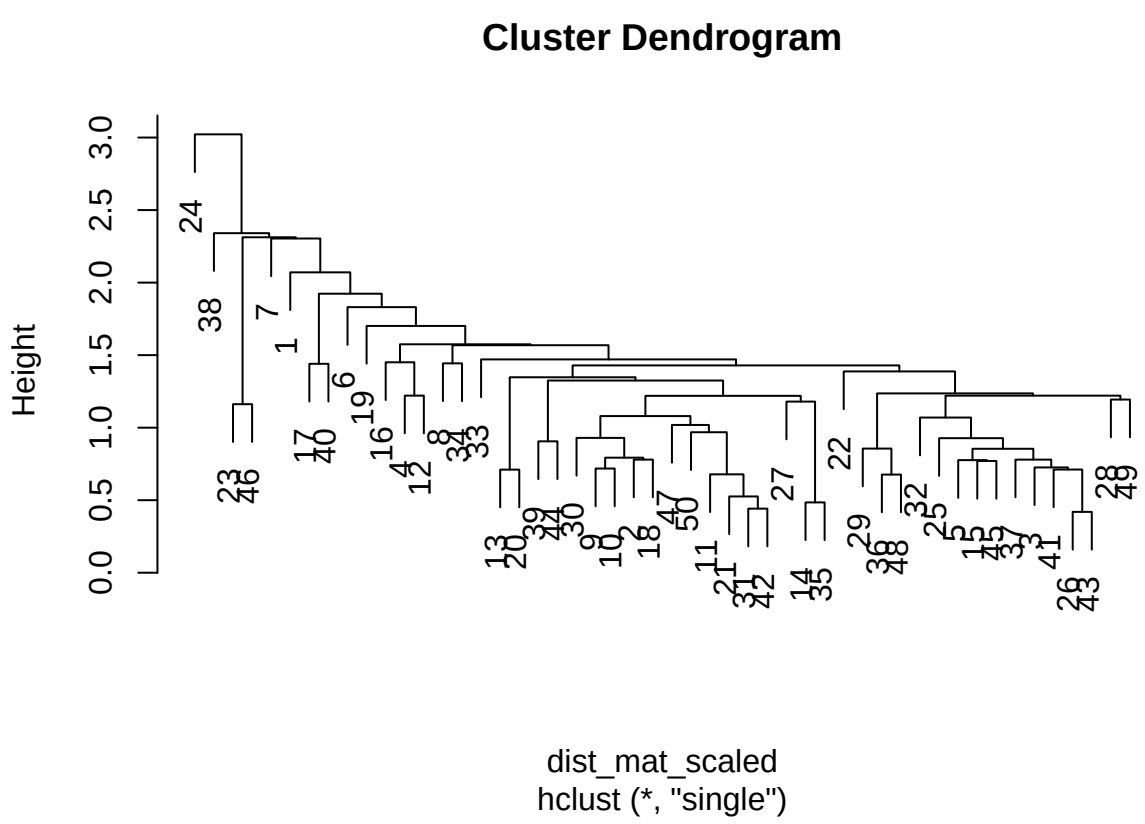
```
set.seed(253)
full_cut <- full_cut %>%
    slice_sample(n = 50)

# Select the variables to be used in clustering
full_cut_sub <- full_cut %>%
    select(gps_away_from_home, case_rate, hospitalized_rate, spend_remoteservices, spend_hcs, emp_incbel

# Summary statistics for the variables
summary(full_cut_sub)
```

```
##  gps_away_from_home   case_rate       hospitalized_rate spend_remoteservices
##  Min.   :-0.17100   Min.   :   89.3   Min.   : 2.980    Min.   :-0.622000
##  1st Qu.:-0.09910   1st Qu.: 1858.0   1st Qu.: 6.728    1st Qu.:-0.006428
##  Median :-0.07345   Median : 6635.0   Median :11.800    Median : 0.086500
##  Mean   :-0.07739   Mean   : 6278.4   Mean   :15.382    Mean   : 0.098263
##  3rd Qu.:-0.04210   3rd Qu.: 9833.2   3rd Qu.:16.600    3rd Qu.: 0.199250
##  Max.   :-0.01050   Max.   :13570.0   Max.   :55.800    Max.   : 0.669000
##    spend_hcs        emp_incbelowmed
##  Min.   :-0.49200   Min.   :-0.2650
##  1st Qu.:-0.08397   1st Qu.:-0.1638
##  Median : 0.01360   Median :-0.1245
##  Mean   : 0.02145   Mean   :-0.1235
##  3rd Qu.: 0.11225   3rd Qu.:-0.0981
##  Max.   : 0.63400   Max.   : 0.1570
```

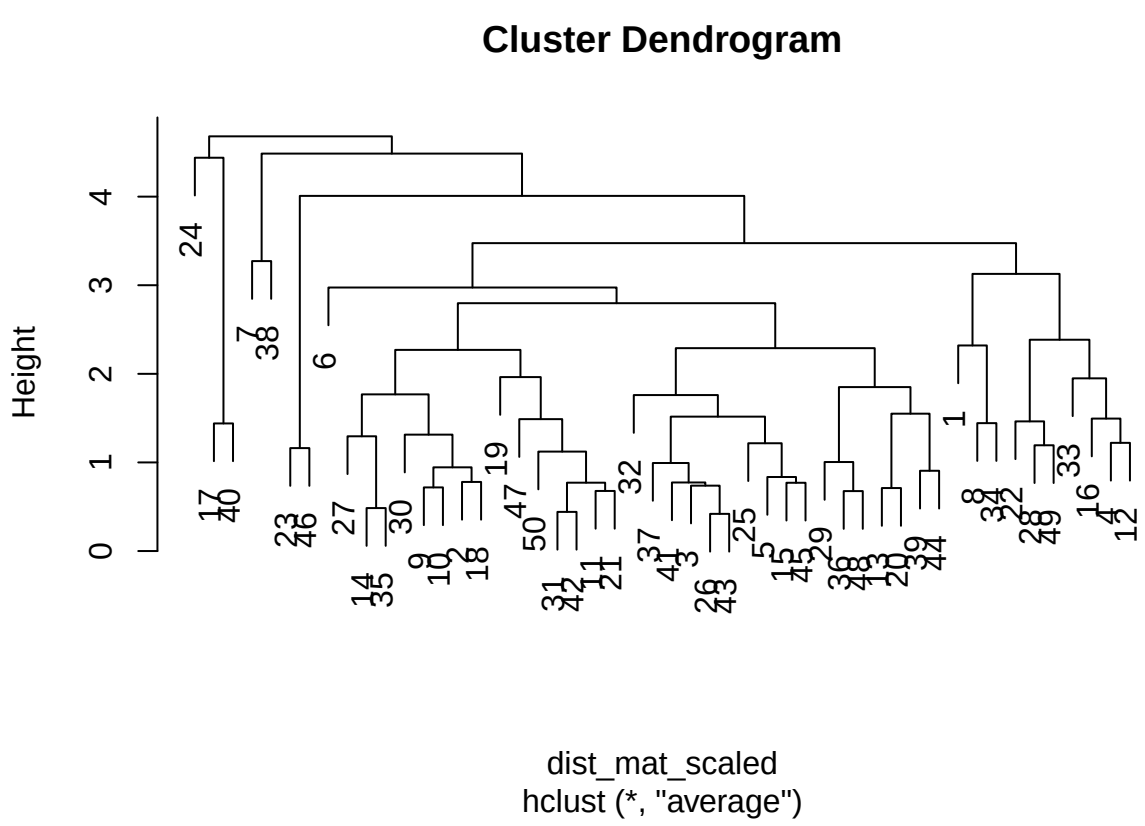
```
# Compute a distance matrix on the scaled data
dist_mat_scaled <- dist(scale(full_cut_sub))

# The (scaled) distance matrix is the input to hclust()
# The method argument indicates the linkage type
hc_complete <- hclust(dist_mat_scaled, method = "complete")
hc_single <- hclust(dist_mat_scaled, method = "single")
hc_average <- hclust(dist_mat_scaled, method = "average")
hc_centroid <- hclust(dist_mat_scaled, method = "centroid")

# Plot dendrograms
plot(hc_complete)
```



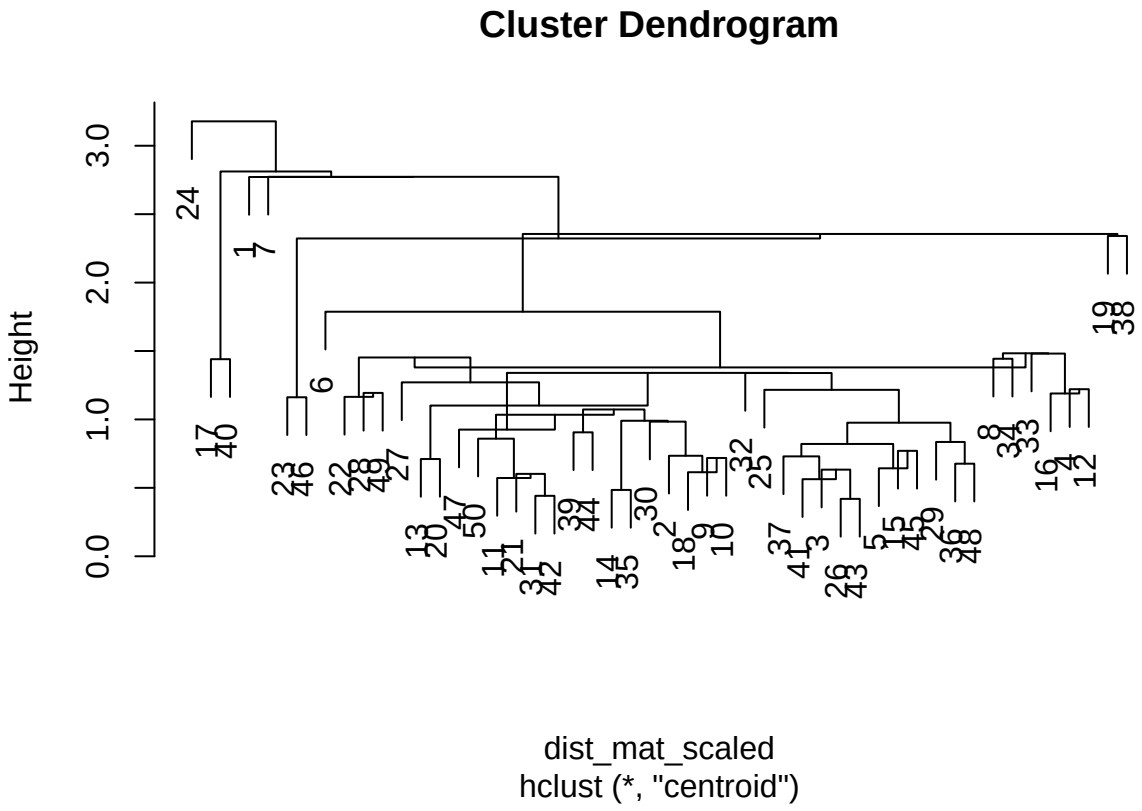

```
plot(hc_single)
```



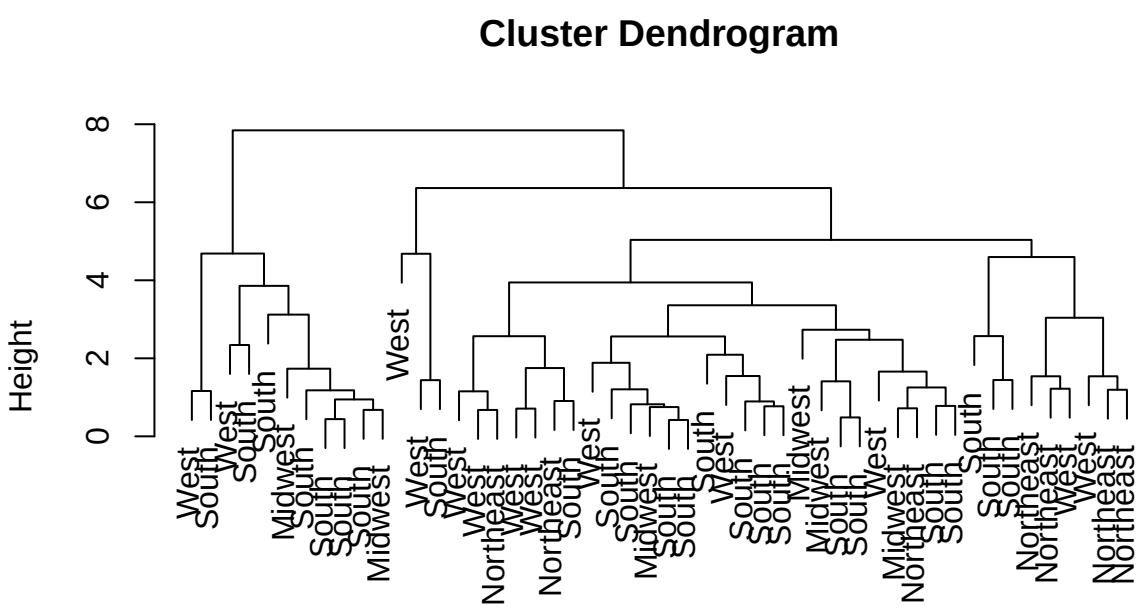

```
plot(hc_average)
```



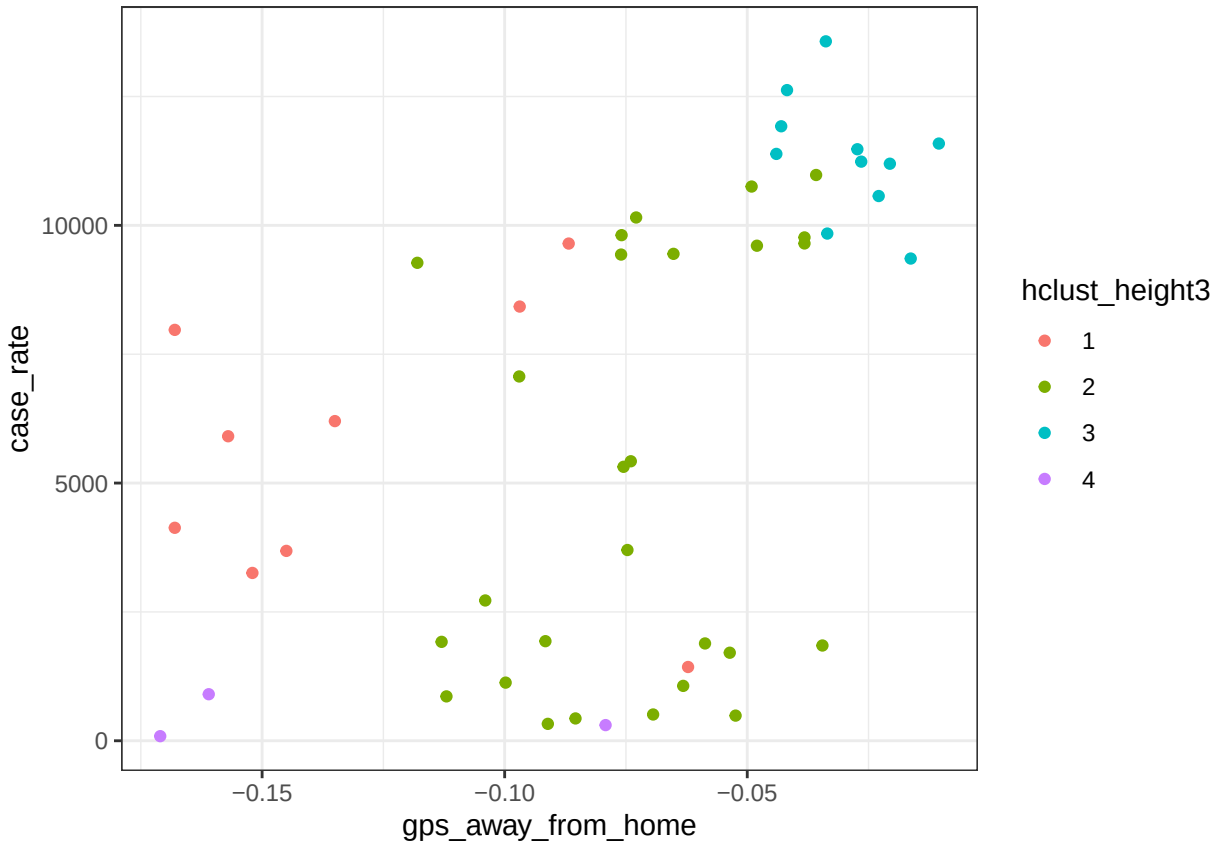

```
plot(hc_centroid)
```




```
#plot with labels
plot(hc_complete, labels = full_cut$Region)
```




```
#complete linkage gives tighter, denser clusters because it is easier to split on the 4 clusters that I

#scatterplot with colors
full_cut <- full_cut %>%
    mutate(
        hclust_height3 = factor(cutree(hc_complete, h = 5)), # Cut at height (h) 3
        hclust_num6 = factor(cutree(hc_complete, k = 4)) # Cut into 4 clusters (k)
    )


ggplot(full_cut, aes(x=gps_away_from_home, y=case_rate, color=hclust_height3))+
    geom_point()+
    theme_bw()
```



K-Means Clustering

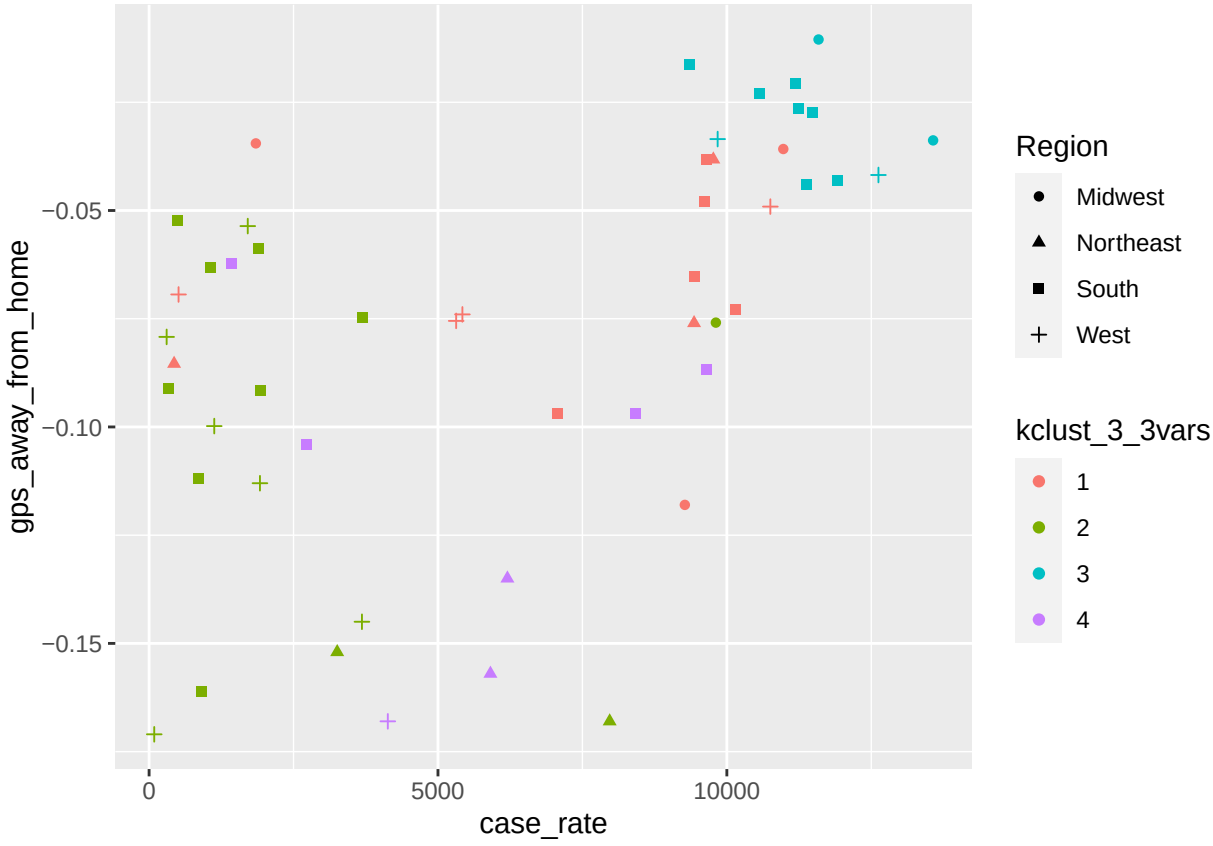
```
# Look at summary statistics of the 3 variables
full_cut_cut <- full_cut %>%
  select(gps_away_from_home, case_rate, hospitalized_rate, spend_remoteservices,spend_hcs, emp_incbelow
summary(full_cut_cut)
```

```
##  gps_away_from_home   case_rate       hospitalized_rate spend_remoteservices
##  Min.   :-0.17100   Min.   :   89.3   Min.   : 2.980    Min.   :-0.622000
##  1st Qu.:-0.09910   1st Qu.: 1858.0   1st Qu.: 6.728    1st Qu.:-0.006428
##  Median :-0.07345   Median : 6635.0   Median :11.800    Median : 0.086500
##  Mean   :-0.07739   Mean   : 6278.4   Mean   :15.382    Mean   : 0.098263
##  3rd Qu.:-0.04210   3rd Qu.: 9833.2   3rd Qu.:16.600    3rd Qu.: 0.199250
##  Max.   :-0.01050   Max.   :13570.0   Max.   :55.800    Max.   : 0.669000
##    spend_hcs         emp_incbelowmed
##  Min.   :-0.49200   Min.   :-0.2650
##  1st Qu.:-0.08397   1st Qu.:-0.1638
##  Median : 0.01360   Median :-0.1245
##  Mean   : 0.02145   Mean   :-0.1235
##  3rd Qu.: 0.11225   3rd Qu.:-0.0981
##  Max.   : 0.63400   Max.   : 0.1570
```

```
# Perform clustering: should you use scale()?
set.seed(253)
kclust_k3_3vars <- kmeans(scale(full_cut_cut), centers = 4)

full_cut_sub2 <- full_cut %>%
    mutate(kclust_3_3vars = factor(kclust_k3_3vars$cluster))

#can type out more variables if you want to see
full_cut_sub2 %>%
    group_by(kclust_3_3vars) %>%
    summarize(across(c(gps_away_from_home, case_rate, spend_remoteservices,spend_hcs, emp_incbelowmed),
```

```
## # A tibble: 4 x 6
##   kclust_3_3vars gps_away_from_home case_rate spend_remoteservices spend_hcs
##   <fct>                       <dbl>     <dbl>                <dbl>     <dbl>
## 1 1                         -0.0651     7310.                0.109    0.0641
## 2 2                         -0.104      2414.               -0.0606  -0.123
## 3 3                         -0.0291    11342.                0.290    0.197
## 4 4                         -0.116      5495.                0.160    0.00469
## # ... with 1 more variable: emp_incbelowmed <dbl>
```

```
#vizualizing two random variables
ggplot(full_cut_sub2, aes(y=gps_away_from_home, x=case_rate, color=kclust_3_3vars, shape=Region)) +
    geom_point()
```



```r
#confusion matrix table
calculate_mode <- function(x) {
  uniqx <- unique(na.omit(x))
  uniqx[which.max(tabulate(match(x, uniqx)))]
}

confMatrix_k <- full_cut_sub2 %>%
  select(Region, kclust_3_3vars)%>%
  group_by(Region)%>%
  summarise(cluster=as.numeric(calculate_mode(kclust_3_3vars)))
confMatrix_k
```

```
## # A tibble: 4 x 2
##   Region    cluster
##   <fct>       <dbl>
## 1 Midwest         1
## 2 Northeast       1
## 3 South           2
## 4 West            2
```

```r
confMatrix_k[confMatrix_k$Region=="West", "cluster"] <- 2

#full_cut_sub2 <- full_cut_sub2 %>%
#  mutate(regionNum = as.numeric(case_when(Region=="Midwest"~3, #Region=="Northeast"~4,Region=="South"~

full_cut_sub2 <- full_cut_sub2 %>%
  mutate(Pred.Region = as.factor(case_when(kclust_3_3vars==3~"Midwest", kclust_3_3vars==4~"Northeast",k

print(confMatrix_k)
```

```
## # A tibble: 4 x 2
##   Region    cluster
##   <fct>       <dbl>
## 1 Midwest         1
## 2 Northeast       1
## 3 South           2
## 4 West            2
```

```r
full_cut_sub2 %>%
  conf_mat(truth = Region, estimate = Pred.Region)
```

```
##            Truth
## Prediction  Midwest Northeast South West
##   Midwest         2         0     7    2
##   Northeast       0         2     4    1
##   South           3         3     5    4
##   West            1         2     8    6
```

```r
log_metrics <- metric_set(sens, yardstick::spec, accuracy)

full_cut_sub2 %>%
  log_metrics(estimate = Pred.Region, truth = Region)
```

```
## # A tibble: 3 x 3
##   .metric  .estimator .estimate
##   <chr>    <chr>          <dbl>
## 1 sens     macro          0.322
## 2 spec     macro          0.749
## 3 accuracy multiclass     0.3
```

```
#rand_index <- adj.rand.index(full_cut_sub2$Region, full_cut_sub2$Pred.Region)

#Accuracy is 34%, this is kind of awful lol.
```