Homework 5: Self-Organizing Maps

University of Chicago
MACS 30100: Perspectives on Computational Modeling

Wanxi Zhou

1. First of all, load the 2016 ANES data. 14 specified features along with the three party affiliations are selected. For each question, answers that are out of the range of the scale would be rendered invalid (i.e., set to missing values). Meanwhile, respondents whose 'pid' is out of range are labeled as "Other". We could notice that 'birthright_b" and "aa3" contain a massive number of invalid responses. Since there would be only 314 observations left if all the NAs were dropped, I choose to keep the missing values. Fortunately, 'supersom' algorithm in the 'kohonen' package could handle data sets with NA values.

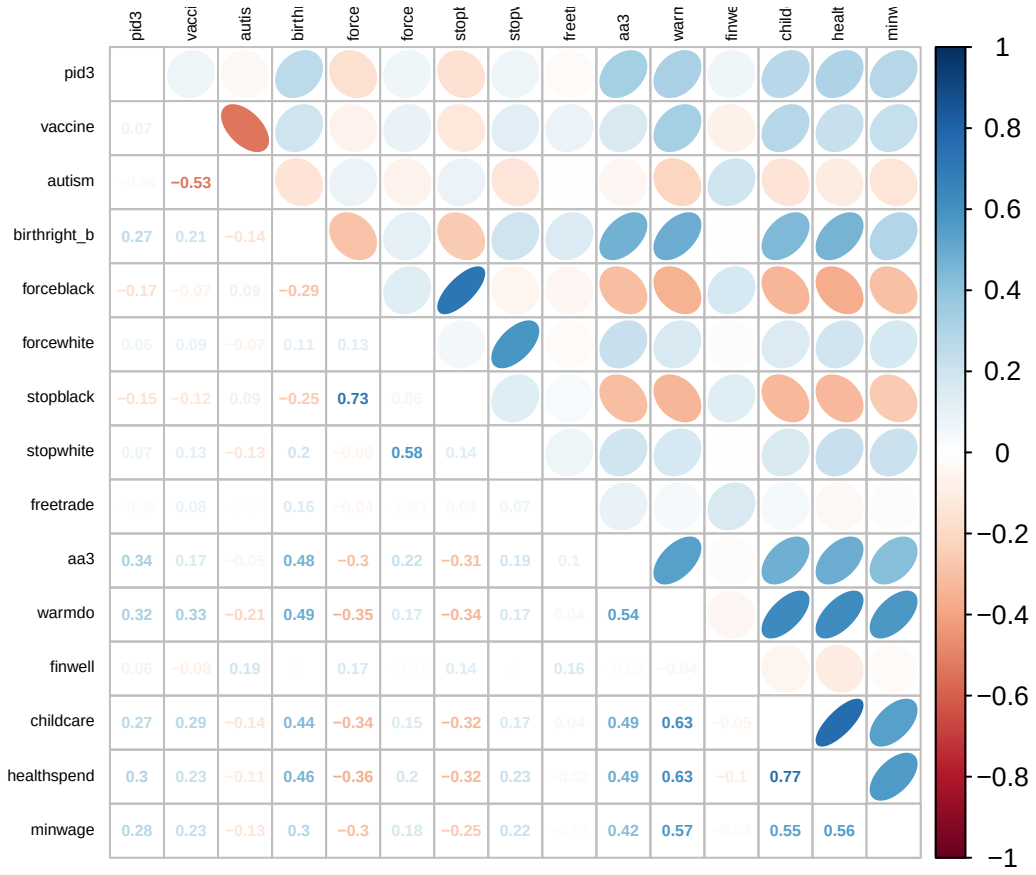
```
library(here)
library(tidyverse)

anes <- read_csv(here("data/anes_2016.csv"))

anes_sample <- anes %>%
  select(pid3, vaccine, autism, birthright_b, forceblack,
         forcewhite,stopblack, stopwhite, freetrade, aa3,
         warmdo, finwell, childcare, healthspend, minwage) %>%
  mutate(pid3 = replace(pid3, pid3 > 3, 2),
         birthright_b = replace(birthright_b, birthright_b > 7, NA),
         stopwhite = replace(stopwhite, stopwhite > 5, NA),
         aa3 = replace(aa3, aa3 > 7, NA),
         warmdo = replace(warmdo, warmdo > 7, NA),
         finwell = replace(finwell, finwell > 7, NA),
         healthspend = replace(healthspend, healthspend > 7, NA),
         minwage = replace(minwage, minwage > 4, NA)) %>%
  mutate(party = pid3) %>%
  mutate(party = case_when(party == 1 ~ "Democrat",
                           party == 2 ~ "Other",
                           party == 3 ~ "Republican"))
```

2. Correlation plots and boxplots are selected to displace the descriptive statistics of the data set. For the purpose of computing the correlation table, missing values are temporarily omitted. From the figure, we could notice strong positive correlations between 'stopblack' and 'forceblack', and between 'healthspend' and 'childcare'. Besides, from the correlations between 'pid3' and the remaining 14 features, we could identify almost uncorrelated relationships between partisan identity and some other features, including 'vaccine', 'autism', 'forcewhite', 'stopwhite', 'freetrade', and 'finwell'.

Boxplots demonstrate the distributions of the responses to each question, grouped by party affiliation. The results support the findings mentioned above that some features share almost identical distributions among partisan identities.

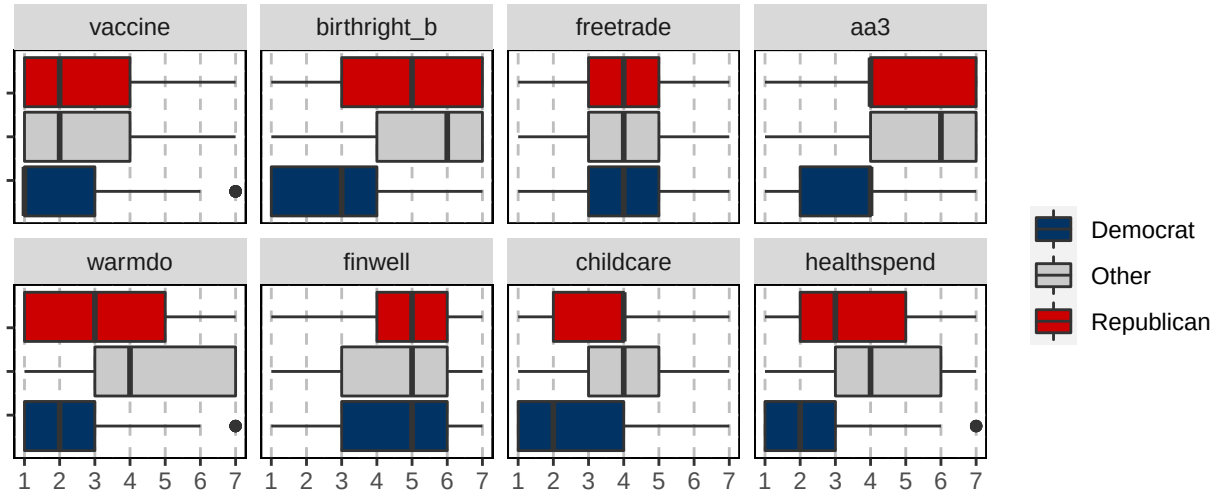
In the following analysis, all the 14 features would be included in the model. However, at the end of the assignment, I would fit an additional model using a subset of the features. The results reveal similar patterns.

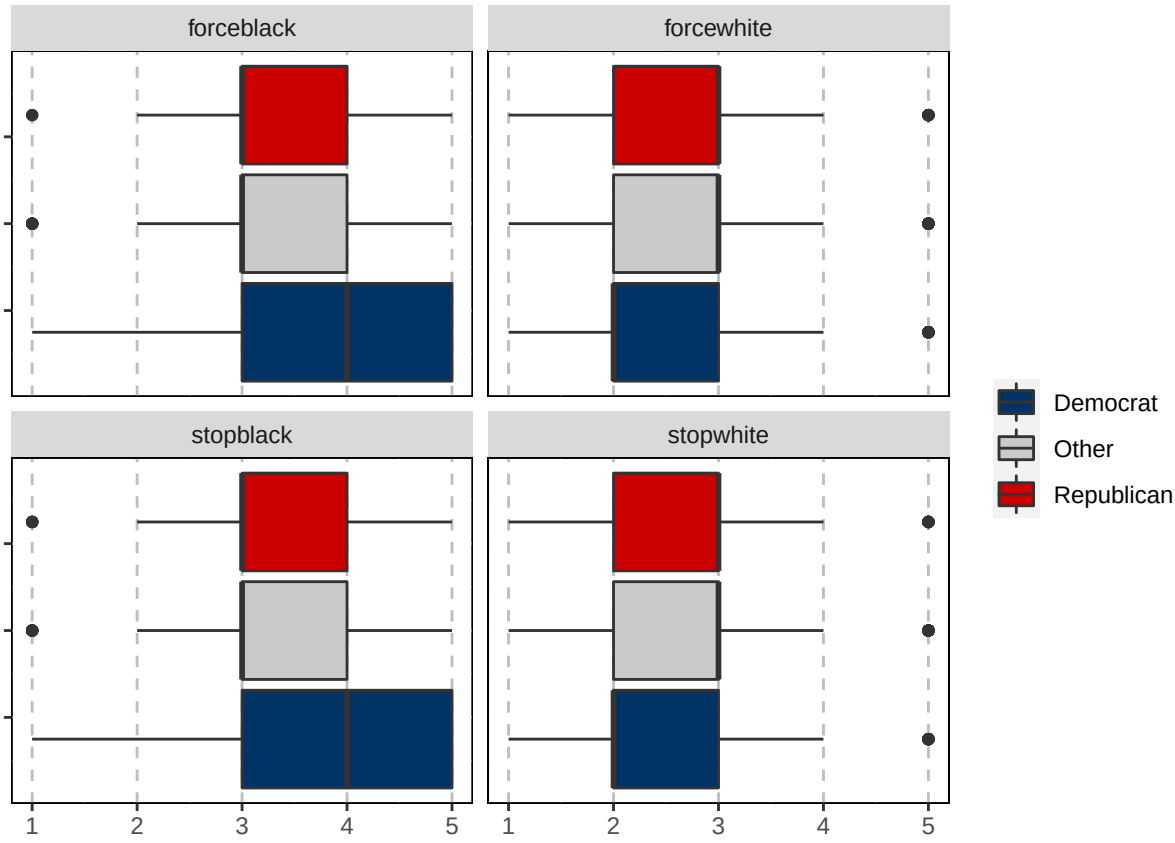
```
library(corrplot)
library(gridExtra)
library(patchwork)
library(ggplot2)
library(reshape2)
library(amerika)

# Customize theme
theme_set(theme(aspect.ratio = 0.75,
                panel.border = element_rect(fill = NA),
                panel.background = element_rect(fill = "white"),
                panel.grid.major.x = element_line(color = "grey", linetype = "dashed"),
                axis.title = element_blank(),
                axis.text.y = element_blank()))

# Plot correlations
C <- cor(anes_sample %>%
           select(-party) %>%
           drop_na())
corrplot.mixed(C, upper = "ellipse", tl.pos = "lt", tl.col = "black",
               tl.cex = 0.5, number.cex = 0.5)
```

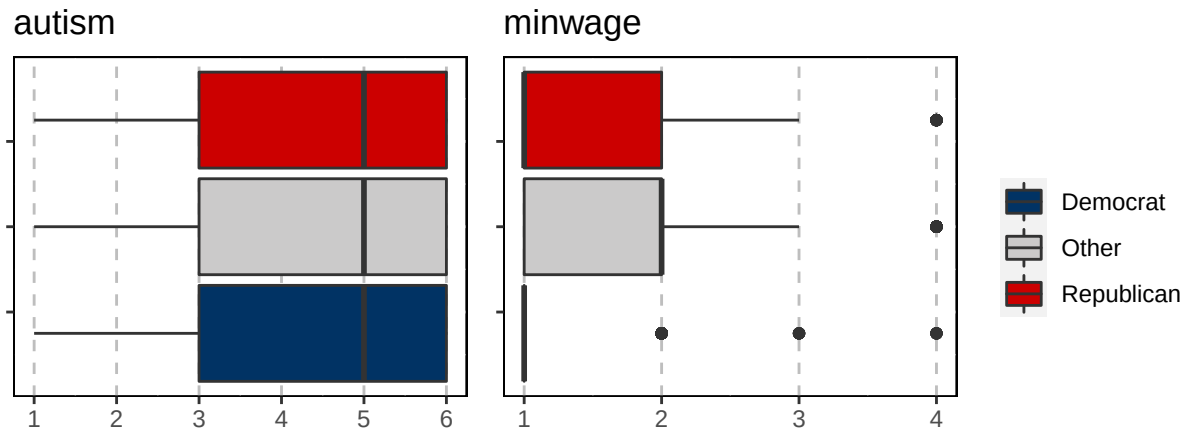



```
# Construct boxplots of the 14 questions
pscale7 <- anes_sample %>%
  select(party, vaccine, birthright_b, freetrade, aa3,
         warmdo, finwell, childcare, healthspend) %>%
```

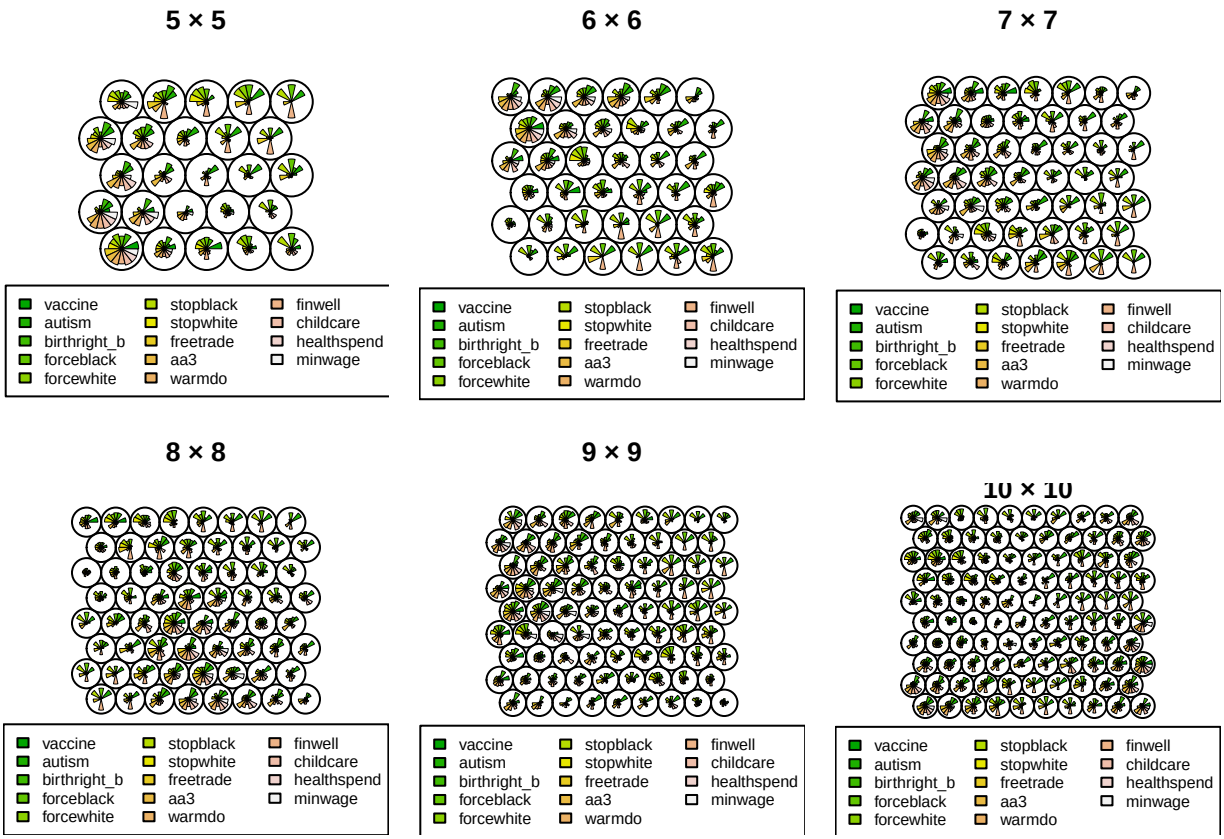
```
  melt(id.vars = "party") %>%
  ggplot(aes(x = value, fill = variable)) +
  geom_boxplot(aes(fill = party)) +
  facet_wrap(~ variable, ncol = 4) +
  scale_x_continuous(n.breaks = 6) +
  labs(fill = "") +
  scale_fill_manual(values = c(amerika_palettes$Dem_Ind_Rep3[1],
                               amerika_palettes$Dem_Ind_Rep3[2],
                               amerika_palettes$Dem_Ind_Rep3[3]))

pscale5 <- anes_sample %>%
  select(party, forceblack, forcewhite, stopblack, stopwhite) %>%
  melt(id.vars = "party") %>%
  ggplot(aes(x = value, fill = variable)) +
  geom_boxplot(aes(fill = party)) +
  facet_wrap(~ variable, ncol = 2) +
  scale_x_continuous(n.breaks = 4) +
  labs(fill = "") +
  scale_fill_manual(values = c(amerika_palettes$Dem_Ind_Rep3[1],
                               amerika_palettes$Dem_Ind_Rep3[2],
                               amerika_palettes$Dem_Ind_Rep3[3]))

pscale4_1 <- anes_sample %>%
  select(party, autism) %>%
  ggplot(aes(x = autism, fill = party)) +
  geom_boxplot() +
  scale_x_continuous(n.breaks = 10) +
  ggtitle("autism") +
  labs(fill = "") +
  scale_fill_manual(values = c(amerika_palettes$Dem_Ind_Rep3[1],
                               amerika_palettes$Dem_Ind_Rep3[2],
                               amerika_palettes$Dem_Ind_Rep3[3])) +
  theme(legend.position = "none")

pscale4_2 <- anes_sample %>%
  select(party, minwage) %>%
  ggplot(aes(x = minwage, fill = party)) +
  geom_boxplot() +
  scale_x_continuous(n.breaks = 3) +
  ggtitle("minwage") +
  labs(fill = "") +
  scale_fill_manual(values = c(amerika_palettes$Dem_Ind_Rep3[1],
                               amerika_palettes$Dem_Ind_Rep3[2],
                               amerika_palettes$Dem_Ind_Rep3[3]))

pscale7; pscale5; pscale4_1 + pscale4_2
```


vaccine
birthright_b
freetrade
aa3
warmdo
finwell
childcare
healthspend
1 2 3 4 5 6 7
Democrat
Other
Republican


forceblack
forcewhite
stopblack
stopwhite
1
2
3
4
5
Democrat
Other
Republican

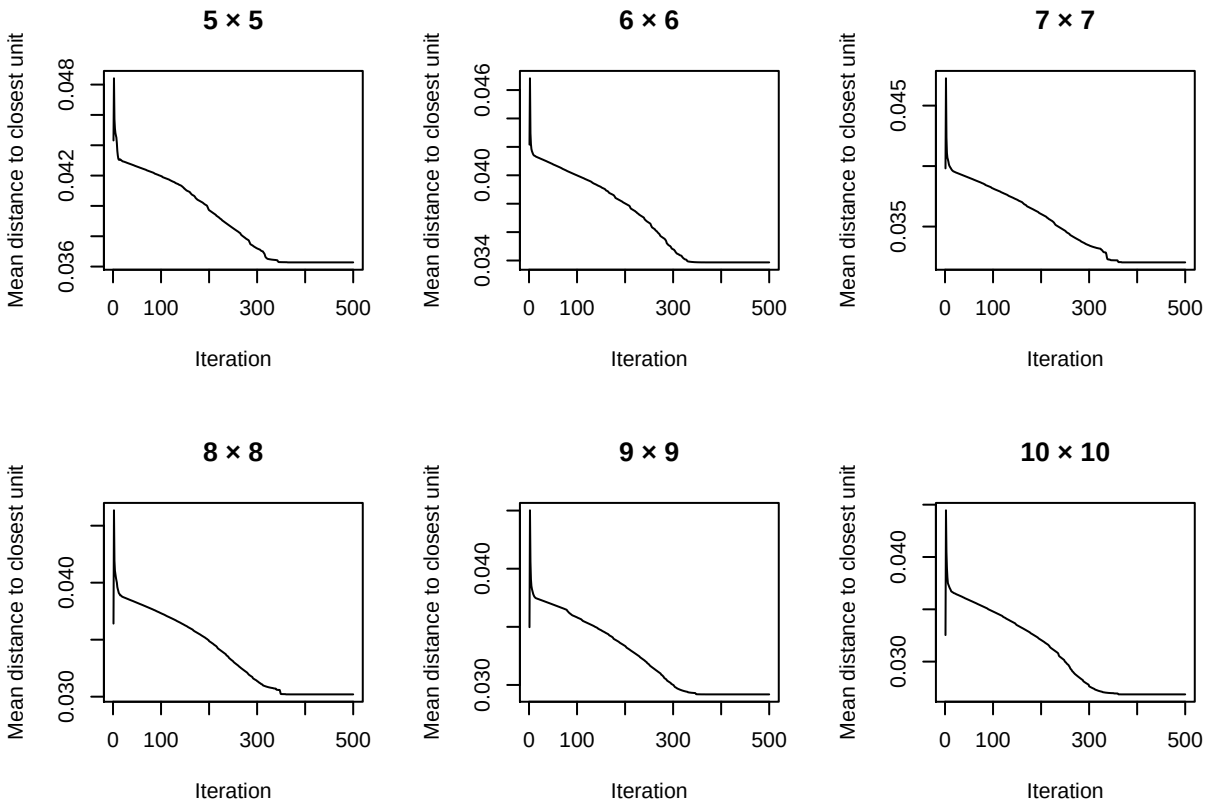


3. Since the questions have various scales, the SOM models are trained on the standardized data set. The codes plots reveal the feature characteristics in each neuron neighborhood such that we could detect the existence of certain classification patterns. The results seem to be promising that they suggest possible partitions within the feature space. For instance, regardless of the tuning parameter - the dimensions of the model grid, nodes with similar and extreme characteristics gather around the corner of the graphs while nodes with relatively neutral feature outcomes group together as well. The space tends to be partitioned into 3 subspaces and this pattern becomes more clear as the dimension of the grid grows. So far, the results seem to coincide with our expectations that observations from the 3 parties might deliver systematically different answers towards the questions and respondents with the same party affiliation might produce similar results.



Additionally, take a look at the performance of the SOM algorithm. Average distance to closest unit drops at iteration proceeds. SOM models with higher-dimension grids achieve more condensed grouping, but might suffer from overfitting problems since the number of observations is limited.

```
# Output change plot of models with grids of different dimensions
par(mfrow = c(2, 3))

for (i in 5:10) {
  som_fit <- som_model(i, i)
  plot(som_fit, type = "change", main = NA) +
  title(str_c(i, " × ", i)) +
  theme(aspect.ratio = 0.75)
}
```

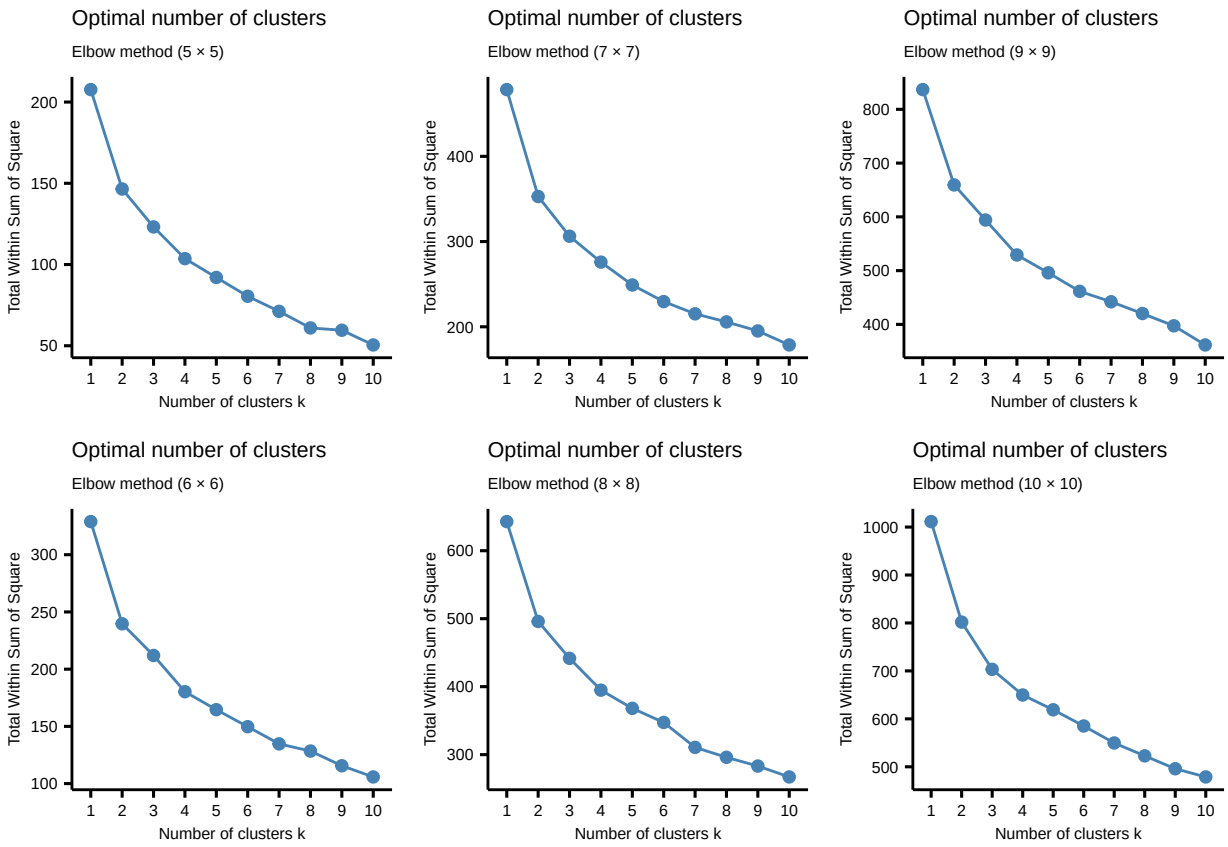


4. Next, K-Means algorithm is utilized to validate the SOM results above. Three methods from the 'factoextra' packages are implemented for determining the optimal k value. The results are unexpected - the suggested clusters are either 1 or 2. Meanwhile, the total within sum of square remains high when the number of clusters is limited.

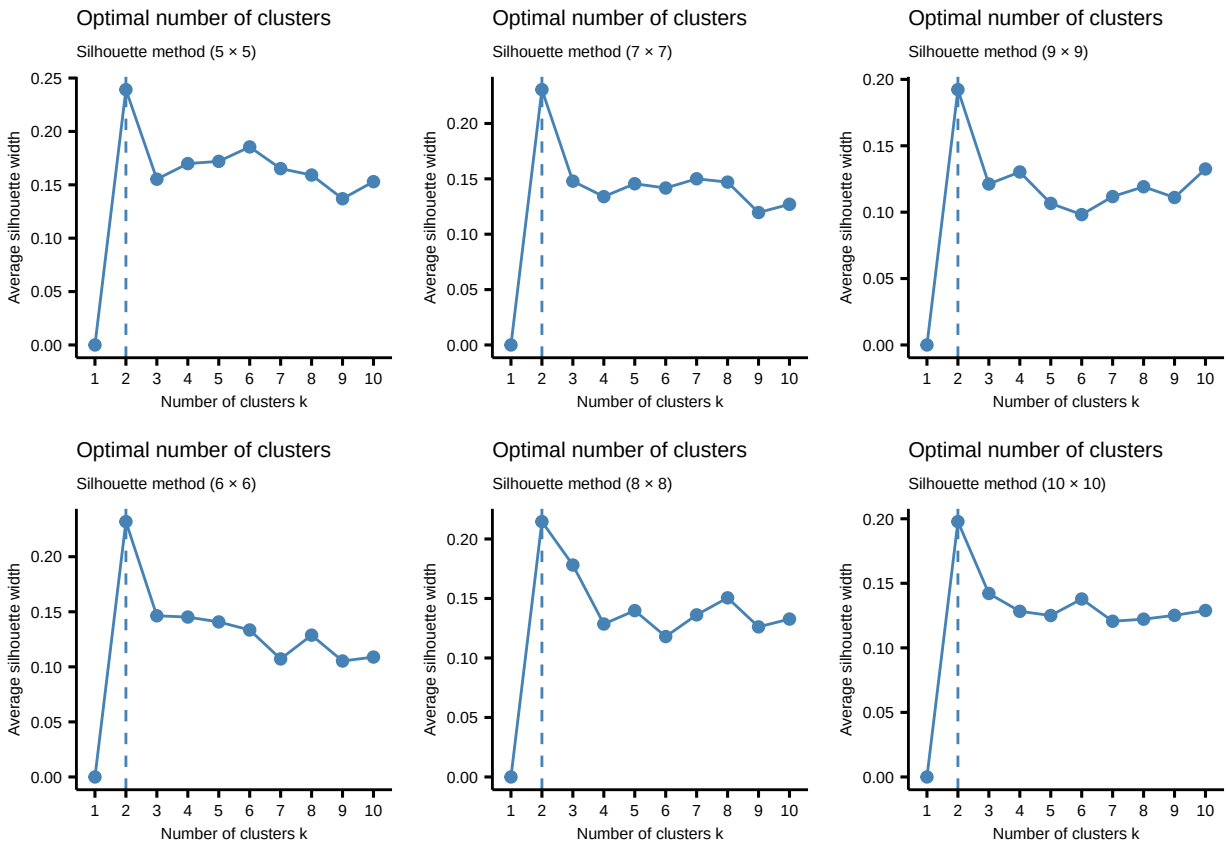
```
source("multiplot.r")
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

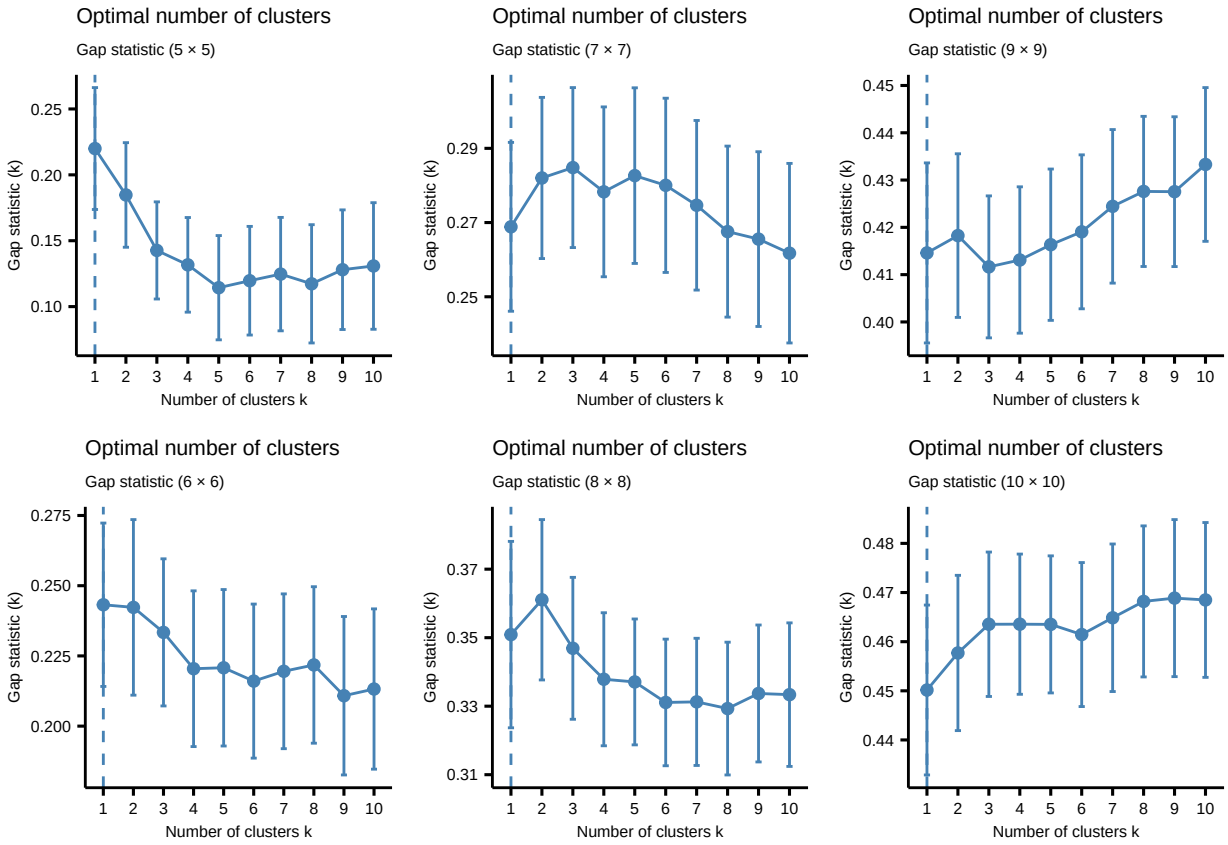
```
# Elbow method
plots1 <- list()

for (i in 5:10) {
  som <- som_model(i, i)
  p <- fviz_nbclust(som$codes[[1]], kmeans, method = "wss") +
       labs(subtitle = str_c("Elbow method (", i, " × ", i, ")")) +
       theme(plot.title = element_text(size = 8),
             plot.subtitle = element_text(size = 6),
             axis.title = element_text(size = 6),
             axis.text = element_text(size = 6))
  plots1[[i - 4]] <- p
}

multiplot(plotlist = plots1, cols = 3)
```




```
# Silhouette method
plots2 <- list()

for (i in 5:10) {
  som <- som_model(i, i)
  p <- fviz_nbclust(som$codes[[1]], kmeans, method = "silhouette") +
       labs(title = "Optimal number of clusters",
            subtitle = str_c("Silhouette method (", i, " × ", i, ")")) +
       theme(plot.title = element_text(size = 8),
             plot.subtitle = element_text(size = 6),
             axis.title = element_text(size = 6),
             axis.text = element_text(size = 6))
  plots2[[i - 4]] <- p
}

multiplot(plotlist = plots2, cols = 3)
```




```
# Gap statistic
plots3 <- list()

for (i in 5:10) {
  som <- som_model(i, i)
  p <- fviz_nbclust(som$codes[[1]], kmeans, nstart = 25,
                    method = "gap_stat", nboot = 50,
                    verbose = FALSE) +
       labs(subtitle = str_c("Gap statistic (", i, " × ", i, ")")) +
       theme(plot.title = element_text(size = 8),
             plot.subtitle = element_text(size = 6),
             axis.title = element_text(size = 6),
             axis.text = element_text(size = 6))
  plots3[[i - 4]] <- p
}

multiplot(plotlist = plots3, cols = 3)
```



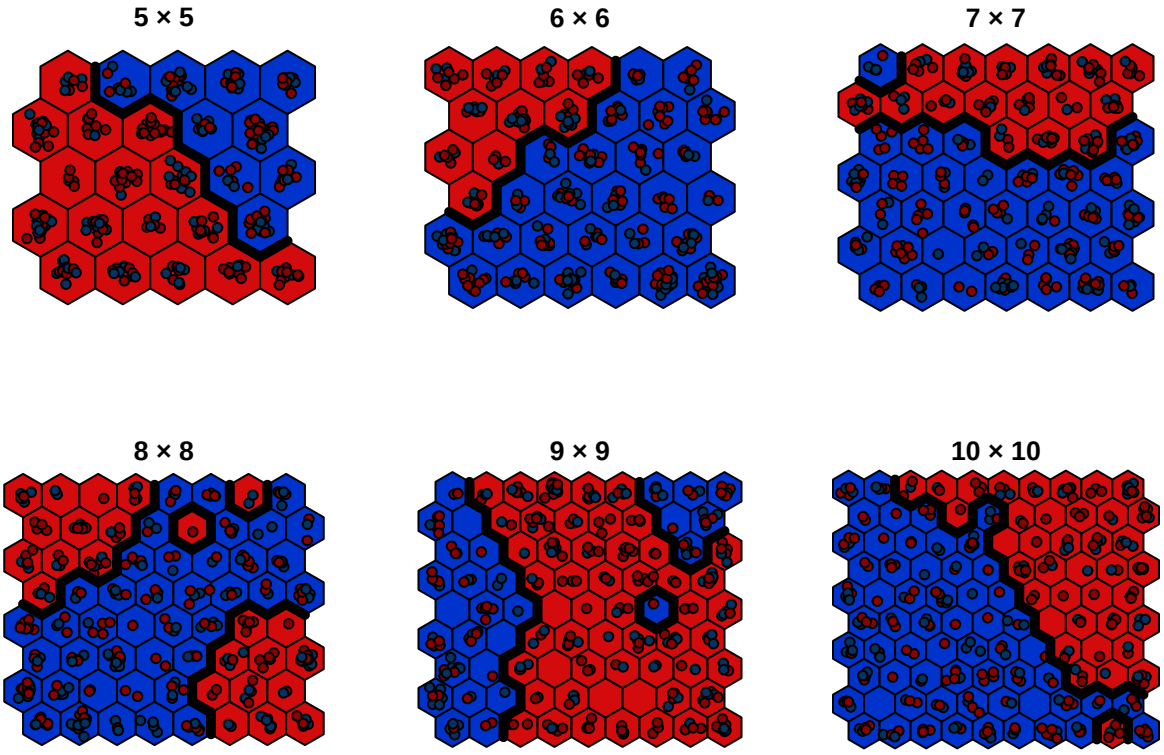
5. As suggested from the optimal cluster figures, I choose to separate the feature space into 2 clusters. However, k-means algorithm with 3 clusters is also applied later on for space exploration. Both algorithms produce unsteady results.

As demonstrated by the 2-cluster result, regardless of the grid dimension, the classification is inaccurate. On the one hand, the borders are broken in grids with higher dimensions, indicating overfitting of the models. On the other hand, in most neighborhoods, there would exist neurons whose true party affiliations are not aligned with the assigned cluster. Furthermore, in an evident portion of the neighborhoods, the misclassified neurons outnumber the correct predictions.

The second graph lays out the k-means clusters of the SOM model with the same tuning parameters in multiple attempts. 5 * 5 grid is selected since it generates the least WSS as suggested above. The results reveal unstable partitions, which deteriorates the reliability of the classification.

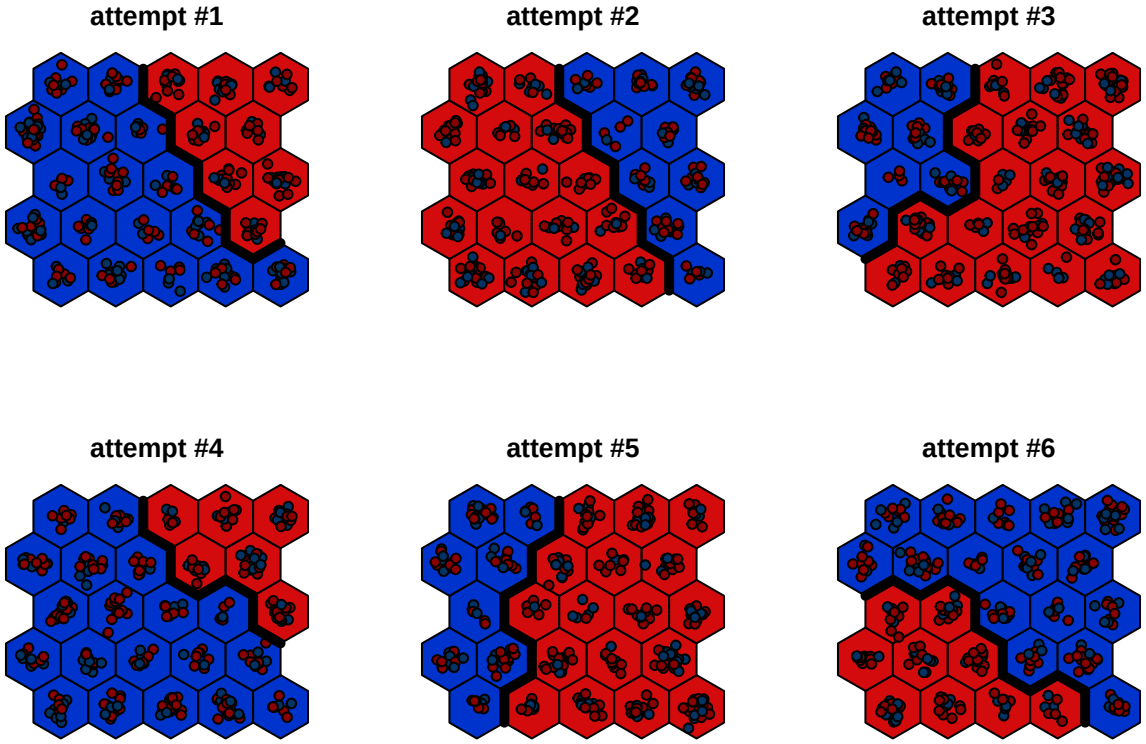
```
# Set point and neuron colors
point_colors2 <- c(amerika_palettes$Dem_Ind_Rep5[1],
                   amerika_palettes$Dem_Ind_Rep5[5])

neuron_colors2 <- c(amerika_palettes$Dem_Ind_Rep5[2],
                    amerika_palettes$Dem_Ind_Rep5[4])

# Convert 3 party affiliations into 2
democrat <- ifelse(anes_sample$pid3 == 1, 1, 2)

# Fit 2-cluster k-means model with various somgrid dimensions
par(mfrow = c(2, 3))

for (i in 5:10) {
```

```
  som_fit <- som_model(i, i)

  kmeans_clusters <- kmeans(som_fit$codes[[1]], 2)

  plot(som_fit,
       type = "mapping",
       pch = 21,
       bg = point_colors2[democrat],
       shape = "straight",
       bgcol = neuron_colors2[kmeans_clusters$cluster],
       main = str_c(i, " × ", i))

  add.cluster.boundaries(x = som_fit, clustering = kmeans_clusters$cluster,
                         lwd = 5, lty = 5)
}
```




```
# Fit SOM model and apply k-means algorithm for multiple times
par(mfrow = c(2, 3))

for (i in 1:6) {
  som_fit <- som_model(5, 5)

  kmeans_clusters <- kmeans(som_fit$codes[[1]], 2)

  plot(som_fit,
       type = "mapping",
```

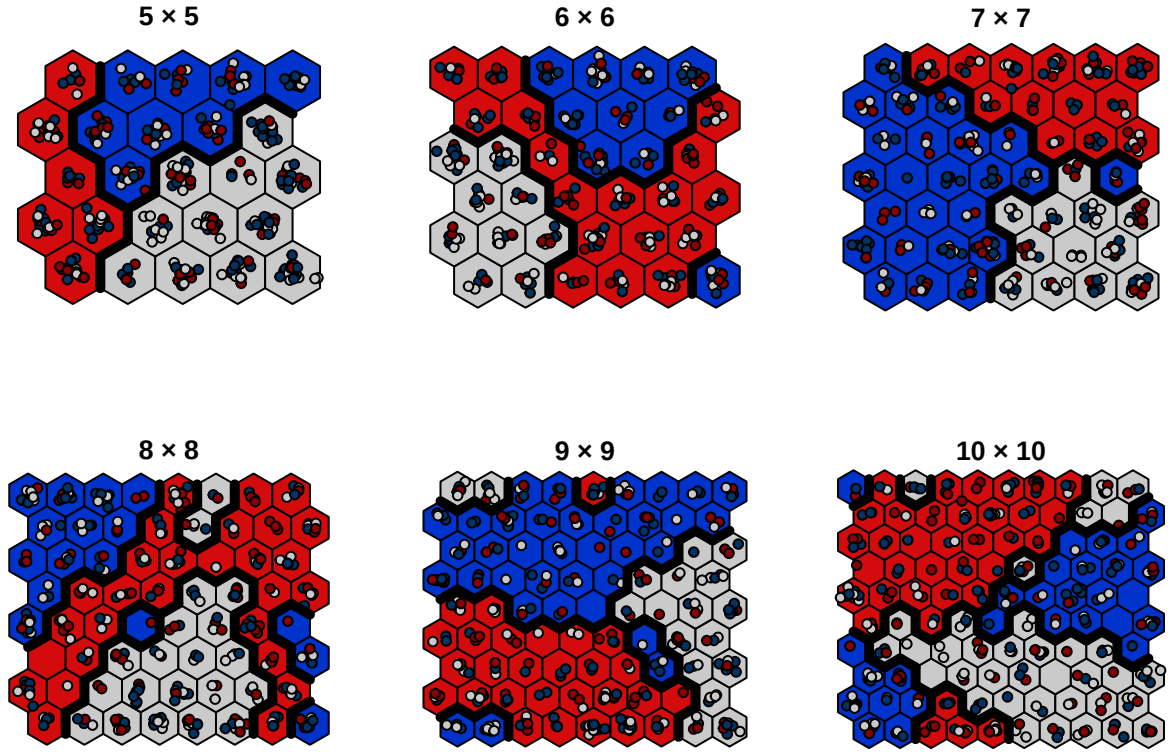
```
        pch = 21,
        bg = point_colors2[democrat],
        shape = "straight",
        bgcol = neuron_colors2[kmeans_clusters$cluster],
        main = str_c("attempt #", i))

  add.cluster.boundaries(x = som_fit, clustering = kmeans_clusters$cluster,
                         lwd = 5, lty = 5)
}
```



6. As mentioned above, 3-cluster k-means method is also utilized to further explore the data set. The results display similar patterns. The misclassification rates are high and the border is unsettled.

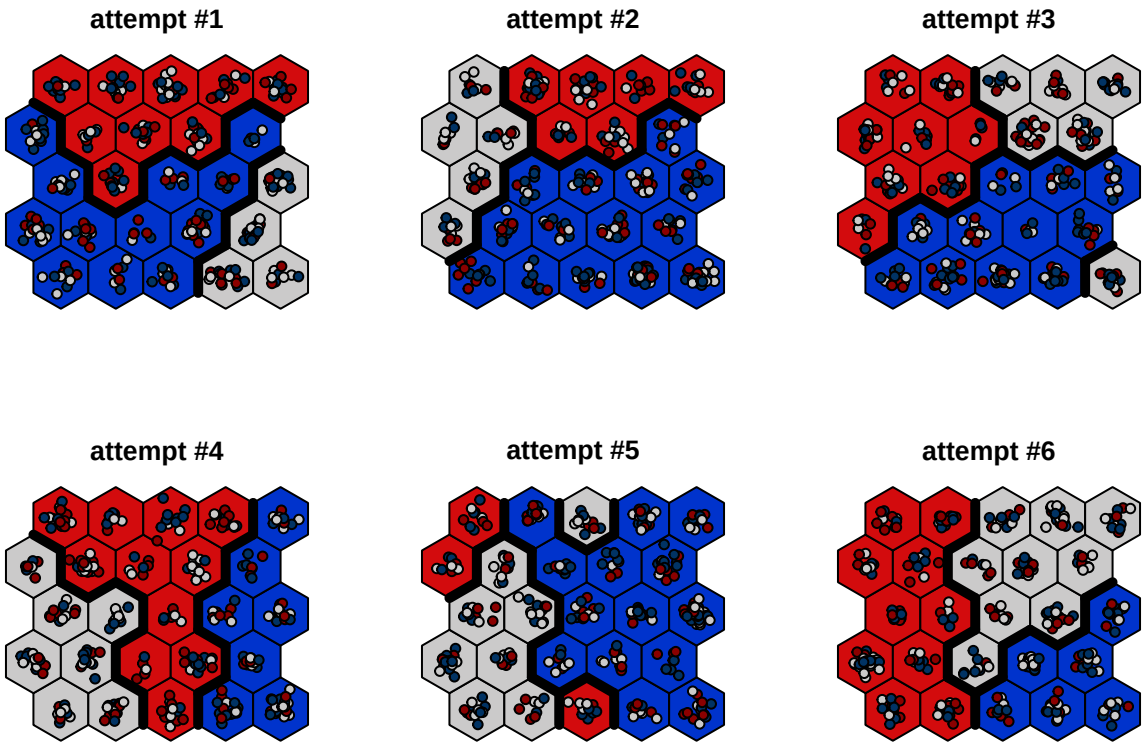
```
# Set point and neuron colors
point_colors3 <- c(amerika_palettes$Dem_Ind_Rep5[1],
                   amerika_palettes$Dem_Ind_Rep5[3],
                   amerika_palettes$Dem_Ind_Rep5[5])

neuron_colors3 <- c(amerika_palettes$Dem_Ind_Rep5[2],
                    amerika_palettes$Dem_Ind_Rep5[3],
                    amerika_palettes$Dem_Ind_Rep5[4])

# Fit 3-cluster k-means model with various somgrid dimensions
par(mfrow = c(2, 3))

for (i in 5:10) {
  som_fit <- som_model(i, i)
```

```
  kmeans_clusters <- kmeans(som_fit$codes[[1]], 3)

  plot(som_fit,
       type = "mapping",
       pch = 21,
       bg = point_colors3[anes_sample$pid3],
       shape = "straight",
       bgcol = neuron_colors3[kmeans_clusters$cluster],
       main = str_c(i, " × ", i))

  add.cluster.boundaries(x = som_fit, clustering = kmeans_clusters$cluster,
                         lwd = 5, lty = 5)
}
```




```
# Fit SOM model and apply k-means algorithm for multiple times
par(mfrow = c(2, 3))

for (i in 1:6) {
  som_fit <- som_model(5, 5)

  kmeans_clusters <- kmeans(som_fit$codes[[1]], 3)

  plot(som_fit,
       type = "mapping",
       pch = 21,
       bg = point_colors3[anes_sample$pid3],
```

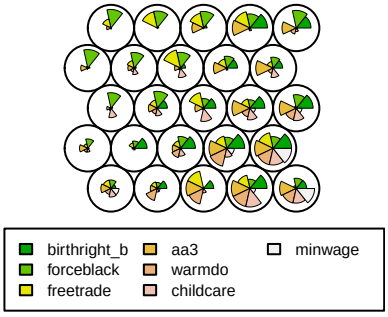
```
      shape = "straight",
      bgcol = neuron_colors3[kmeans_clusters$cluster],
      main = str_c("attempt #", i))

  add.cluster.boundaries(x = som_fit, clustering = kmeans_clusters$cluster,
                         lwd = 5, lty = 5)
}
```



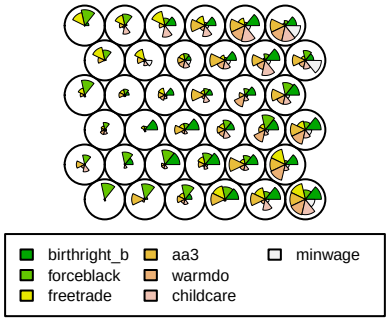
7. Finally, a data set with selected features is inspected using the identical algorithms applied so far. Features that are uncorrelated with the 'pid3' and that are highly correlated with one of the other features are removed from the feature space. The aim is to examine whether more seemingly 'pid'-related features would engender significant differences in public opinions among groups.

According to the figures, the 2-cluster k-means results with 5 * 5 somgrid demonstrate more steady classification outcomes. Nevertheless, the misclassification rate remains high regardless of the grid dimensions and the cluster numbers. However, the SOM results reveal more distinct differences between areas in the graphs.

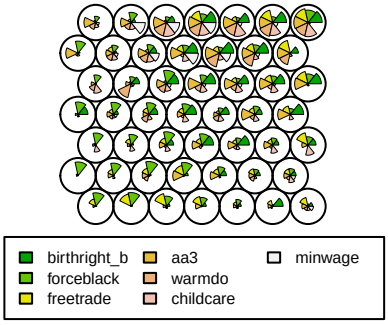
Codes Plot
5 × 5



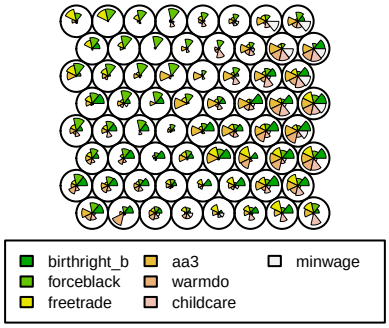
Codes Plot
6 × 6



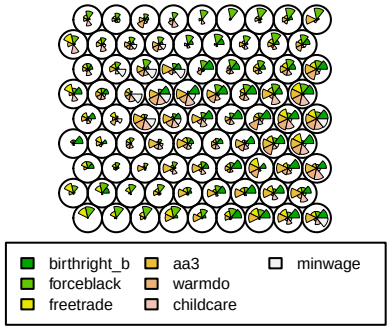
Codes Plot
7 × 7



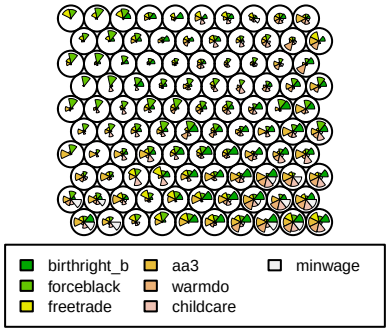
Codes Plot
8 × 8

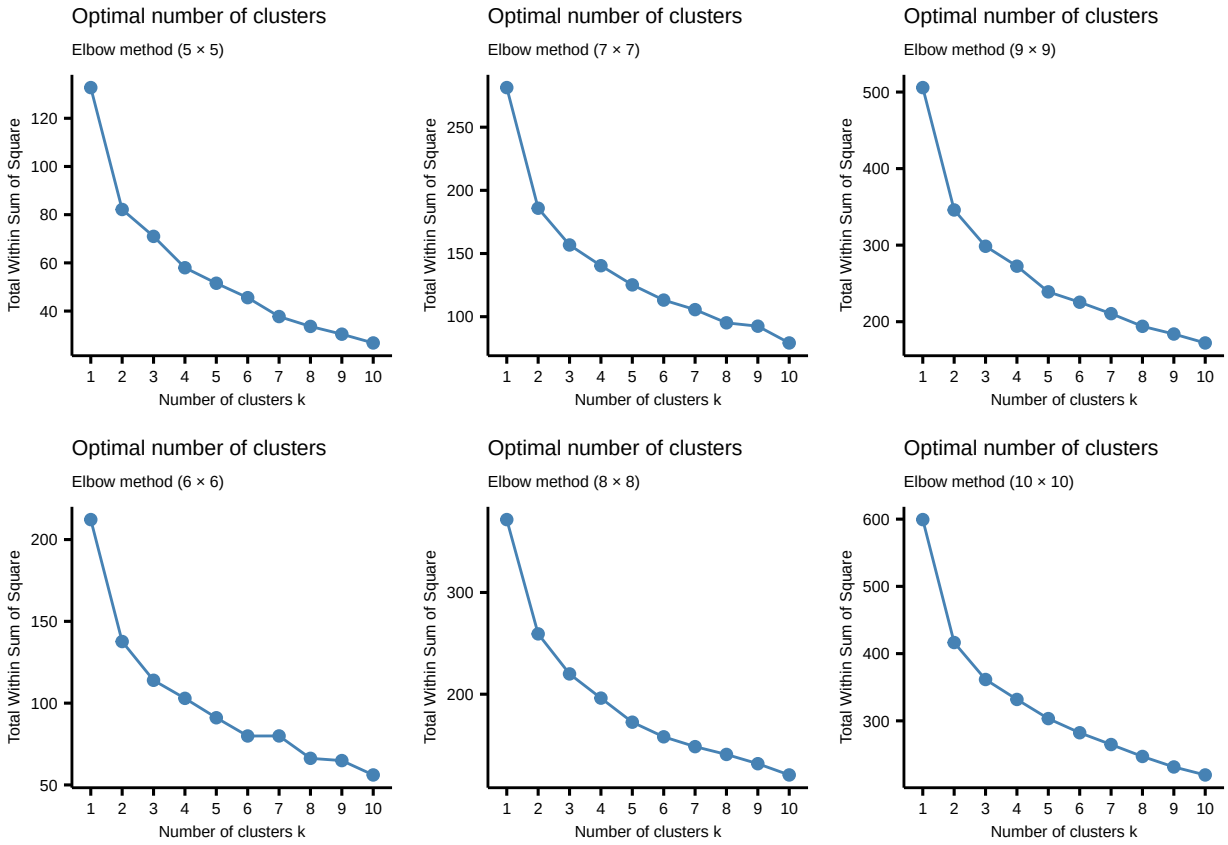


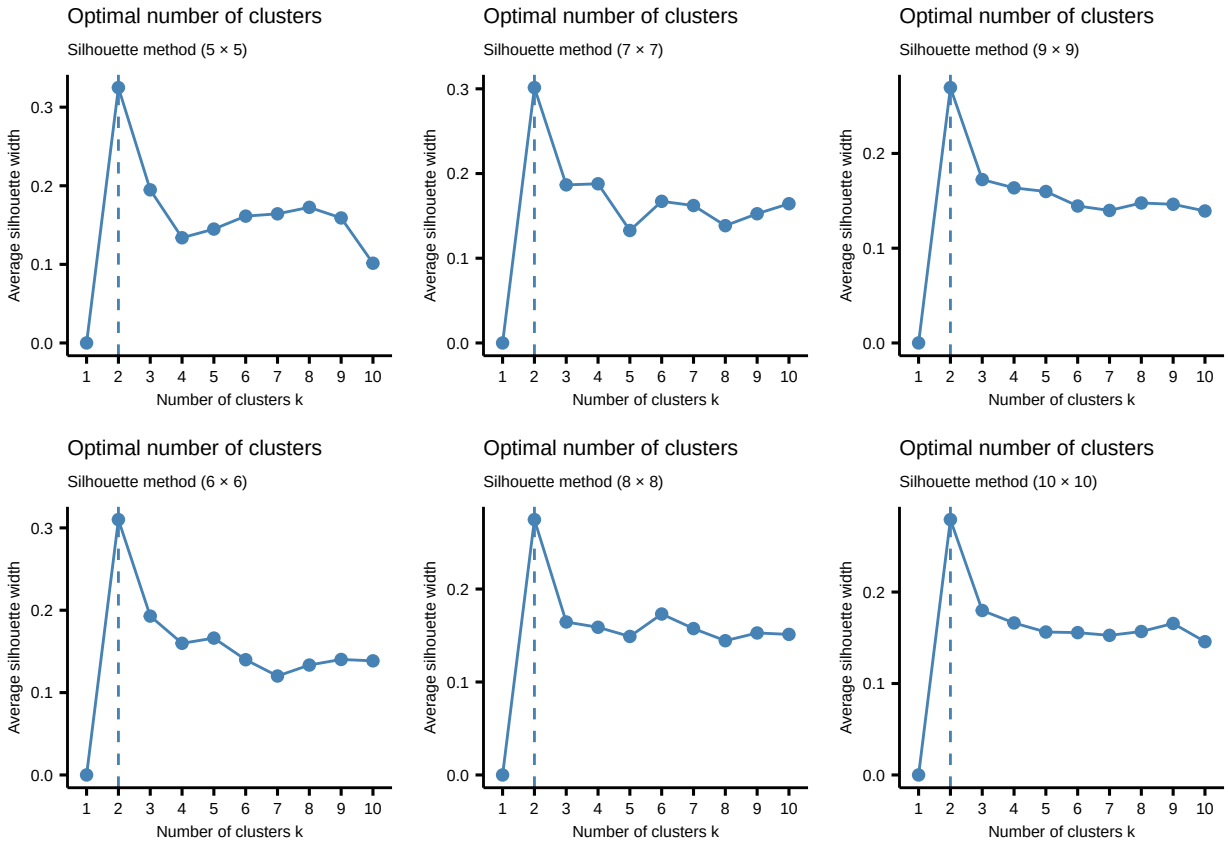
Codes Plot
9 × 9

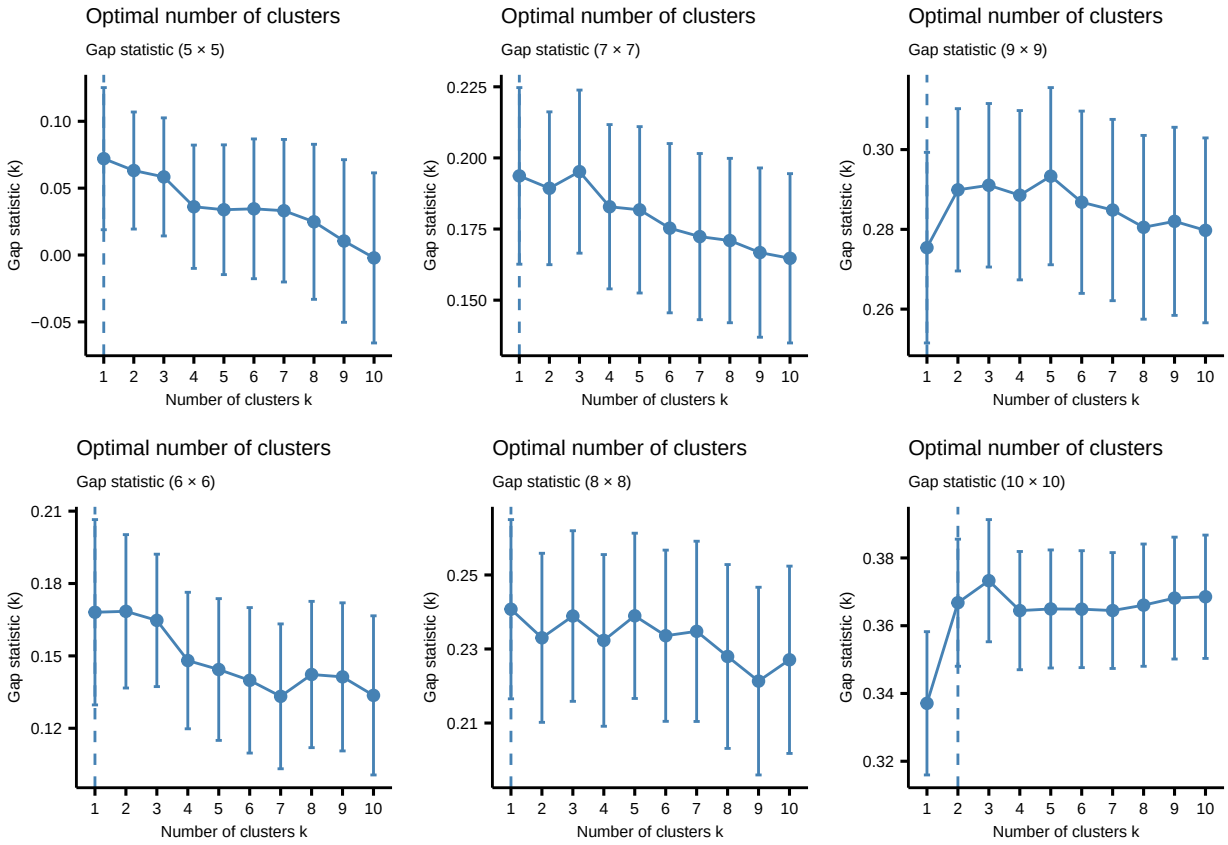


Codes Plot
10 × 10

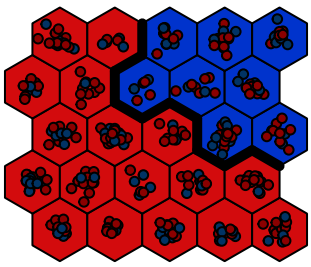



Optimal number of clusters
Elbow method (5 × 5)
Total Within Sum of Square
Number of clusters k
Optimal number of clusters
Elbow method (7 × 7)
Total Within Sum of Square
Number of clusters k
Optimal number of clusters
Elbow method (9 × 9)
Total Within Sum of Square
Number of clusters k
Optimal number of clusters
Elbow method (6 × 6)
Total Within Sum of Square
Number of clusters k
Optimal number of clusters
Elbow method (8 × 8)
Total Within Sum of Square
Number of clusters k
Optimal number of clusters
Elbow method (10 × 10)
Total Within Sum of Square
Number of clusters k

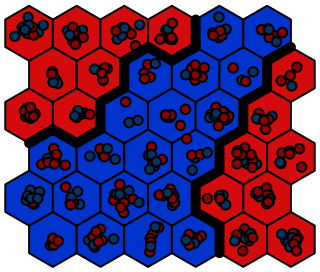

Optimal number of clusters
Silhouette method (5 × 5)
Optimal number of clusters
Silhouette method (7 × 7)
Optimal number of clusters
Silhouette method (9 × 9)
Optimal number of clusters
Silhouette method (6 × 6)
Optimal number of clusters
Silhouette method (8 × 8)
Optimal number of clusters
Silhouette method (10 × 10)
Average silhouette width
Number of clusters k


Optimal number of clusters
Gap statistic (5 × 5)
Gap statistic (k)
Number of clusters k
Optimal number of clusters
Gap statistic (7 × 7)
Gap statistic (k)
Number of clusters k
Optimal number of clusters
Gap statistic (9 × 9)
Gap statistic (k)
Number of clusters k
Optimal number of clusters
Gap statistic (6 × 6)
Gap statistic (k)
Number of clusters k
Optimal number of clusters
Gap statistic (8 × 8)
Gap statistic (k)
Number of clusters k
Optimal number of clusters
Gap statistic (10 × 10)
Gap statistic (k)
Number of clusters k

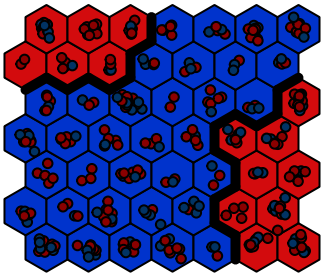
5 × 5



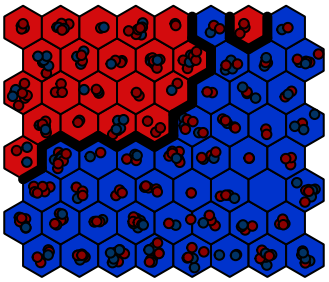
6 × 6



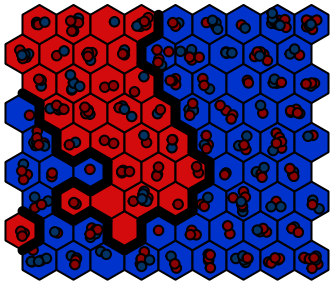
7 × 7



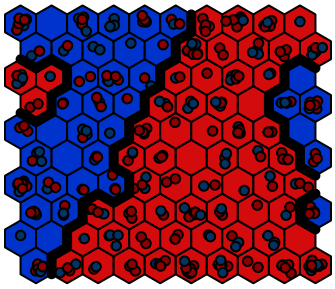
8 × 8



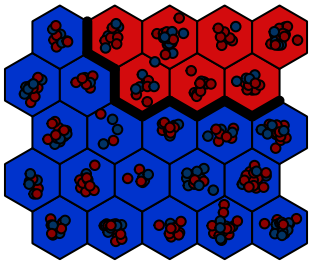
9 × 9



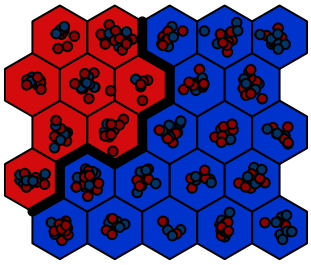
10 × 10



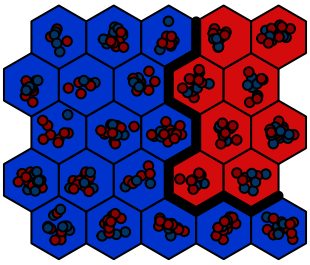
attempt #1



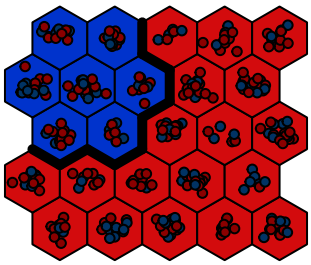
attempt #2



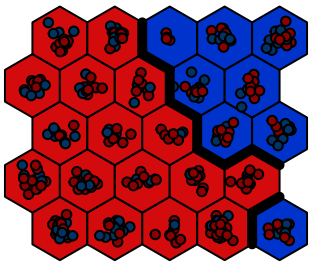
attempt #3



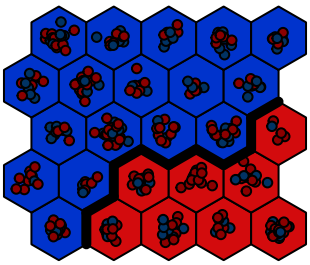
attempt #4



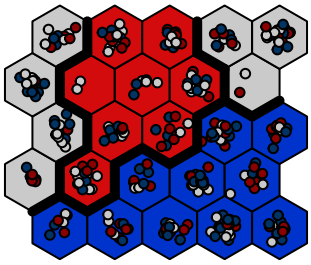
attempt #5



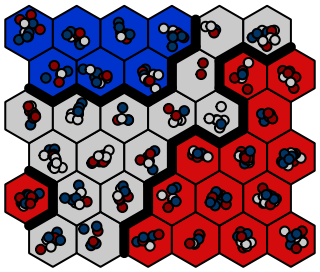
attempt #6



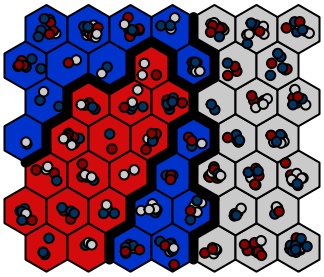
5 × 5



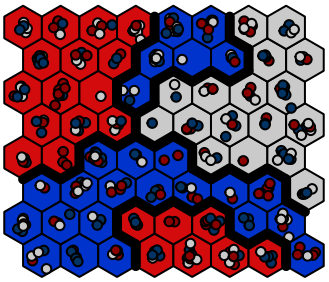
6 × 6



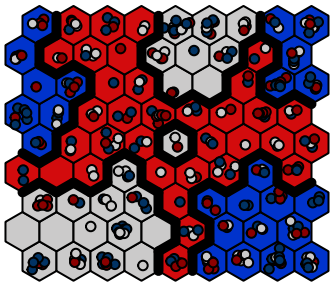
7 × 7



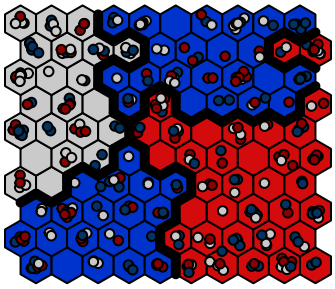
8 × 8

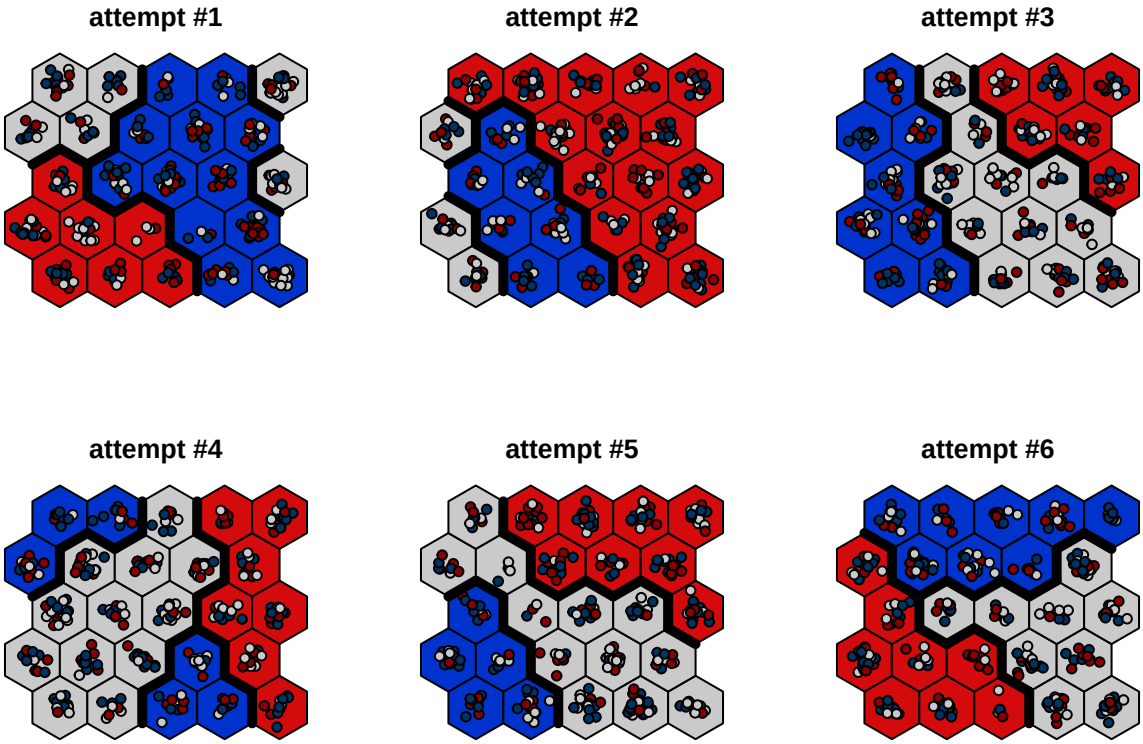


9 × 9



10 × 10





8. To sum up, although the SOM results indicate possible partisan differences within the feature space, the k-means algorithm manifests that the results are too intertwined to separate. Meanwhile, even if the features are selected, the k-means clustering results are still noisy. Nevertheless, the SOM results show more evident grouping patterns.