CAL Project - Knowledge Transfer Documentation - Data Engineering 3

4/28/2020

For questions, please email musch.sam@gmail.com.

Introduction

```
library(tidyverse)
library(flexclust)
library(lubridate)

# csv from previous file
cluster_df <- read_csv('../cluster_prep.csv')
```

This chunk is separating our dataset into members and non-members.

Our initial clusters will be applied to current members only. After that, the same algorithm will applied to non-members.

Note that `members` and `non_members` are now separate dataframes.

```
members <-
    cluster_df %>%
    filter(MEMBERSHIP_STATUS_CODE == "C")

non_members <-
    cluster_df %>%
    filter(MEMBERSHIP_STATUS_CODE != "C")
```

Algorithm

Remember that we had 5 initial columns that we want to add to the algorithm.

- Sports
- Social
- Networking
- Learning
- Legislation

```
set.seed(1)   # Reproduce the same results
cl1 <- kcca(members[, 2:6], k=5, kccaFamily('kmeans'))
cluster <- predict(cl1, save.data=T)

# 2:5      the category score columns
# k = 4    we will find 4 clusters
# kcca     the function that runs our clustering
```

Note that the k-means algorithm finds the **centers** for each of the 4 clusters. The **center** for Cluster 1 is: $-.6, \: -.6, \: -.2, \: -0.2, \: .28$.

	Learn scores	Legis scores	social scores	sports scores	networking scores
Cluster1	-.6	-.6	-.2	-.2	.28
Cluster2	0	0	0	0	0
Cluster3	4.3	-1	-.6	-1	-1.6
Cluster4	-.7	-1	3.9	-.6	-1.5
Cluster5	-.3	-.3	-.1	1.2	-.5

A large positive number indicates what the cluster likes. As an example, we can see that **Cluster1** is the **Networking** cluster.

This chunk of code gets the algorithm from the previous chunk to output the results into our **original** dataframe.

```
members['cluster'] = cluster
```

Our dataframe for `members` now has a column that indicates which cluster each person fell into.

Applying to Non-Members

The previous chunk "learned" from our existing members. This chunk applies the same algorithm, but to new people (our non-members).

The `rbind` just joins the members and non-members back together into 1 dataframe.

```
cluster <- predict(cl1,
                   newdata = non_members[, 2:6],
                   save.data=T)

non_members['cluster'] = cluster

cluster_df <- rbind(members, non_members)
```

Results

We want to add a column that indicates the description of the cluster so we can use them later on. We also want to make sure that this label will be dynamic

This chunk of code is how I calculated the "centers" table from above.

```
cluster_summary <-
  cluster_df %>%
  group_by(cluster) %>%
  summarize_all(list(mean)) %>%
  select(cluster, learning, legis,
       social, sports, networking)
```

Now we are going to assign .25 as a threshold. That means that if a cluster is not above .25 for any one specific category, they fall into the "baseline" group.

```
cluster_summary <-
  cluster_summary %>%
  mutate(Description =
       ifelse(learning > .25, "Learning",
       ifelse(legis > .25, "Legis",
       ifelse(social > .25, "Social",
       ifelse(sports > .25, "Sports", "Baseline"))))) %>%
  select(cluster, Description)


cluster_df <-
  cluster_df %>%
  inner_join(cluster_summary, by = 'cluster')
```

```
# AS OF RIGHT NOW
# 1 - Networking
# 2 - Baseline
# 3 - Learning
# 4 - Social
# 5 - Sports


write.csv(cluster_df, 'D:/Group Folder/Sam/per_person_clusters.csv', row.names =
F)
```