k-means with Imputation

`ClustImpute` package

PMS

10 May, 2023

Preliminary

Loading & Cleaning Data

```
set.seed(2023)
library(cluster)
library(ClustImpute)
library(ggplot2)
library(factoextra)
```

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

```
library(clusterCrit)
load('../../../../../local_data/codes/create_master/master_pms_df.Rdata')
```

Assumptions of the Alogrithm

This algorithm "draws the missing values iteratively based on the current cluster assignment so that correlations are considered on this level". Also, "penalizing weights are imposed on imputed values and successively decreased (to zero) as the missing data imputation gets better". The idea is that the missing value is imputed by those other observations that are more similar to it (ie. in the same cluster).

Algorithm steps:

1. It replaces all NAs by random imputation, i.e., for each variable with missings, it draws from the marginal distribution of this variable not taking into account any correlations with other variables
2. Weights < 1 are used to adjust the scale of an observation that was generated in step 1. The weights are calculated by a (linear) weight function that starts near zero and converges to 1 at n_end.
3. A k-means clustering is performed with a number of c_steps steps starting with a random initialization.
4. The values from step 2 are replaced by new draws conditionally on the assigned cluster from step 3.
5. Steps 2-4 are repeated nr_iter times in total. The k-means clustering in step 3 uses the previous cluster centroids for initialization.
6. After the last draws a final k-means clustering is performed.

All Metrics Together

Implementation

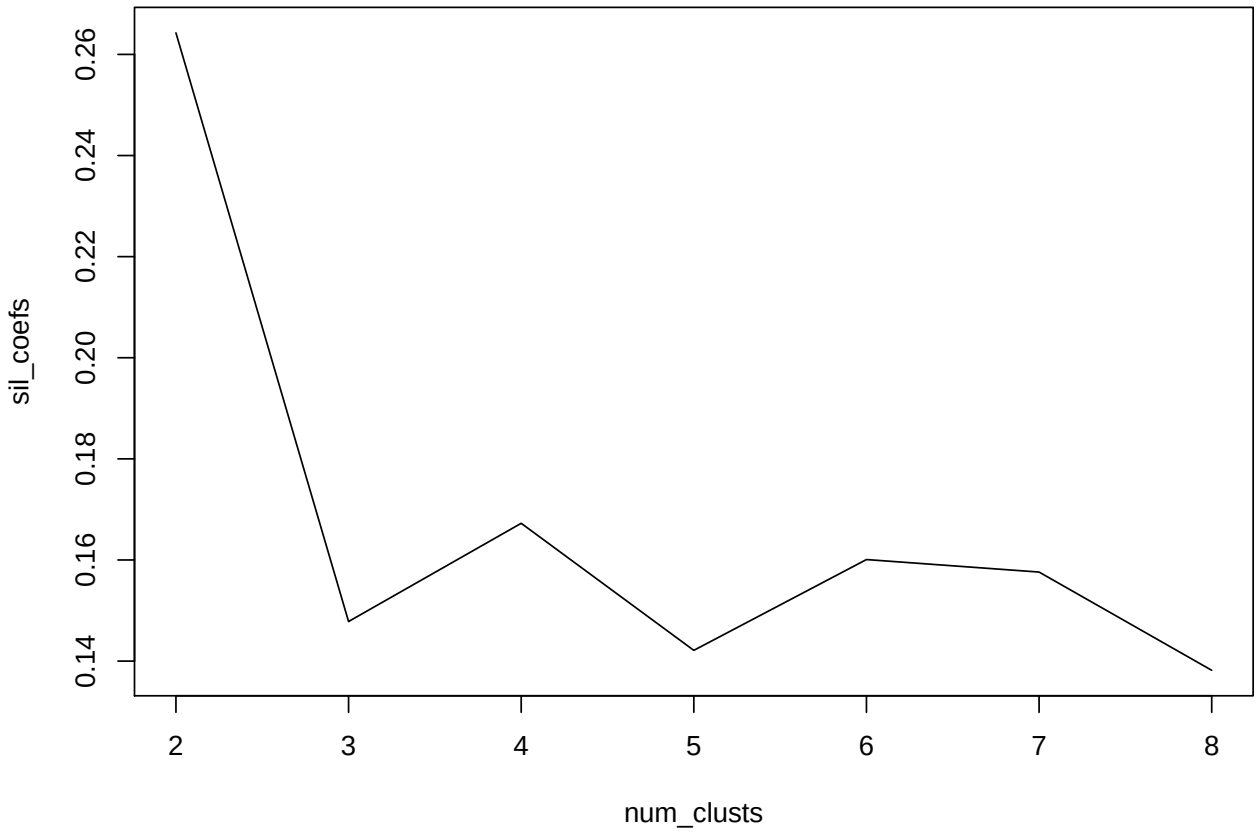
(with 2% subsampling)

```
#cluster data
cols = c("prox_idx_emp", "prox_idx_pharma", "prox_idx_childcare", "prox_idx_health", "prox_idx_grocery"
subsample = nrow(master)/50 # 2% subsampling
subsam = master[sample(nrow(master), subsample), cols]
sum(is.na(subsam))
```

```
## [1] 60763
```

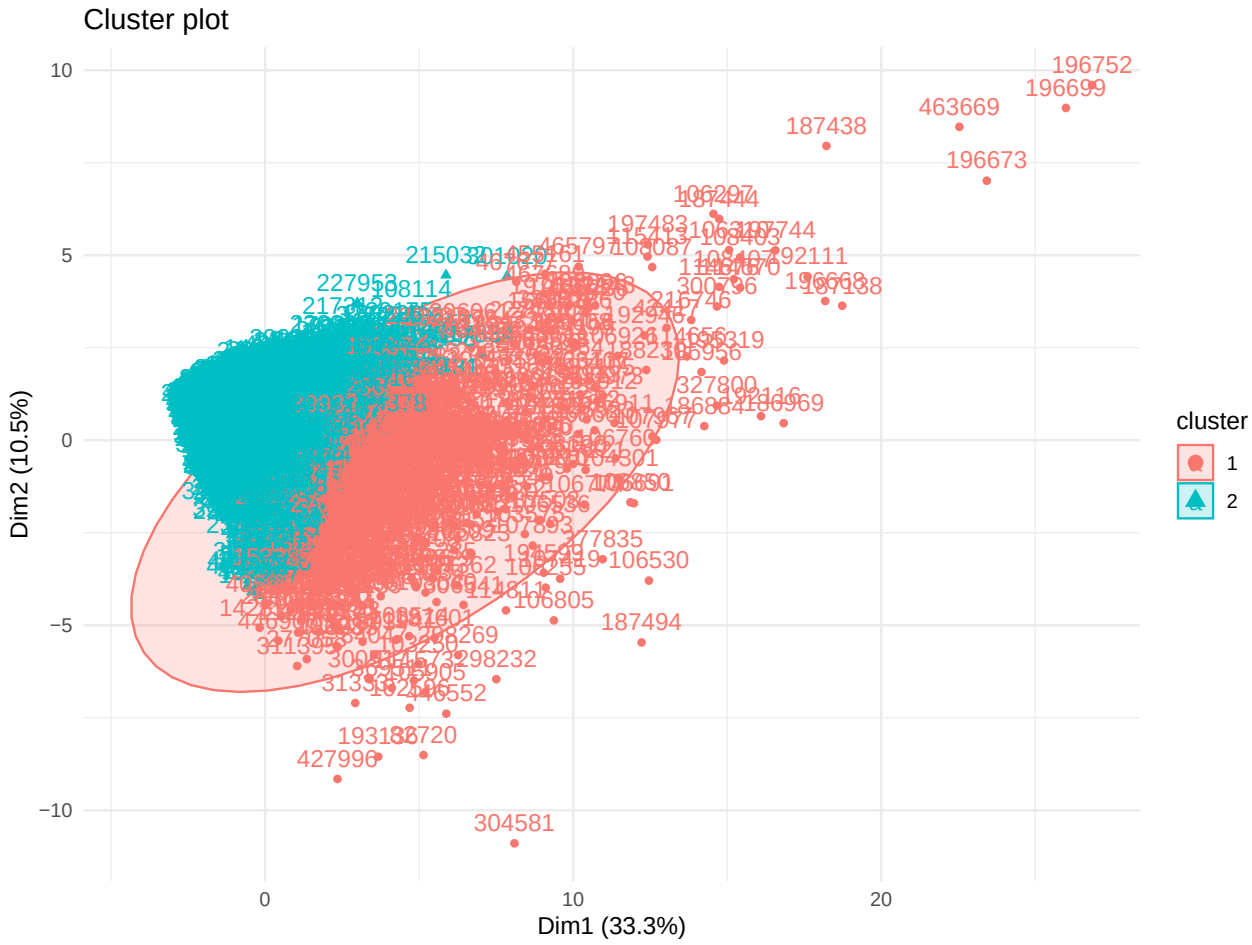
```
#algorithm
sil_coefs = c()
counter = 1
num_clusts = 2:8
for (i in num_clusts){
  nr_iter = 10 # iterations of procedure
  n_end = 10 # step until convergence of weight function to 1
  #nr_cluster = 3 # number of clusters
  c_steps = 50 # number of cluster steps per iteration
  res = ClustImpute(subsam,nr_cluster=i, nr_iter=nr_iter, c_steps=c_steps, n_end=n_end)
  sil_coefs[counter] = intCriteria(as.matrix(res$complete_data),res$clusters, 'Silhouette')$silhouette
  counter = counter + 1
}

#plot silhouette coefficients
plot(sil_coefs~num_clusts, type = 'l')
```




```
#re-run algorithm with highest sil
res = ClustImpute(subsam,nr_cluster=num_clusts[which(sil_coefs == max(sil_coefs))], nr_iter=nr_iter, c_s

#plot
# ggplot(res$complete_data,aes(prox_idx_emp,prox_idx_pharma,color=factor(res$clusters))) + geom_point()
pass = list(data = res$complete_data, cluster = res$clusters)
fviz_cluster(pass, ellipse.type = "norm") + theme_minimal()
```



Cut-off Values

```
for (k in cols){
  clus_medians = c()
  counter = 1
  for (i in unique(res$clusters)){
    clus_medians[counter] = median(res$complete_data[res$clusters == i,k])
    counter = counter + 1
  }
  cutoff = c()
  for (j in 1:(length(clus_medians)-1)){
    cutoff[j] = (clus_medians[j] + clus_medians[j+1])/2
  }
  print(k)
  print(round(cutoff, 5))
}
```

```
## [1] "prox_idx_emp"
## [1] 0.04935
## [1] "prox_idx_pharma"
## [1] 0.04485
## [1] "prox_idx_childcare"
## [1] 0.1023
```

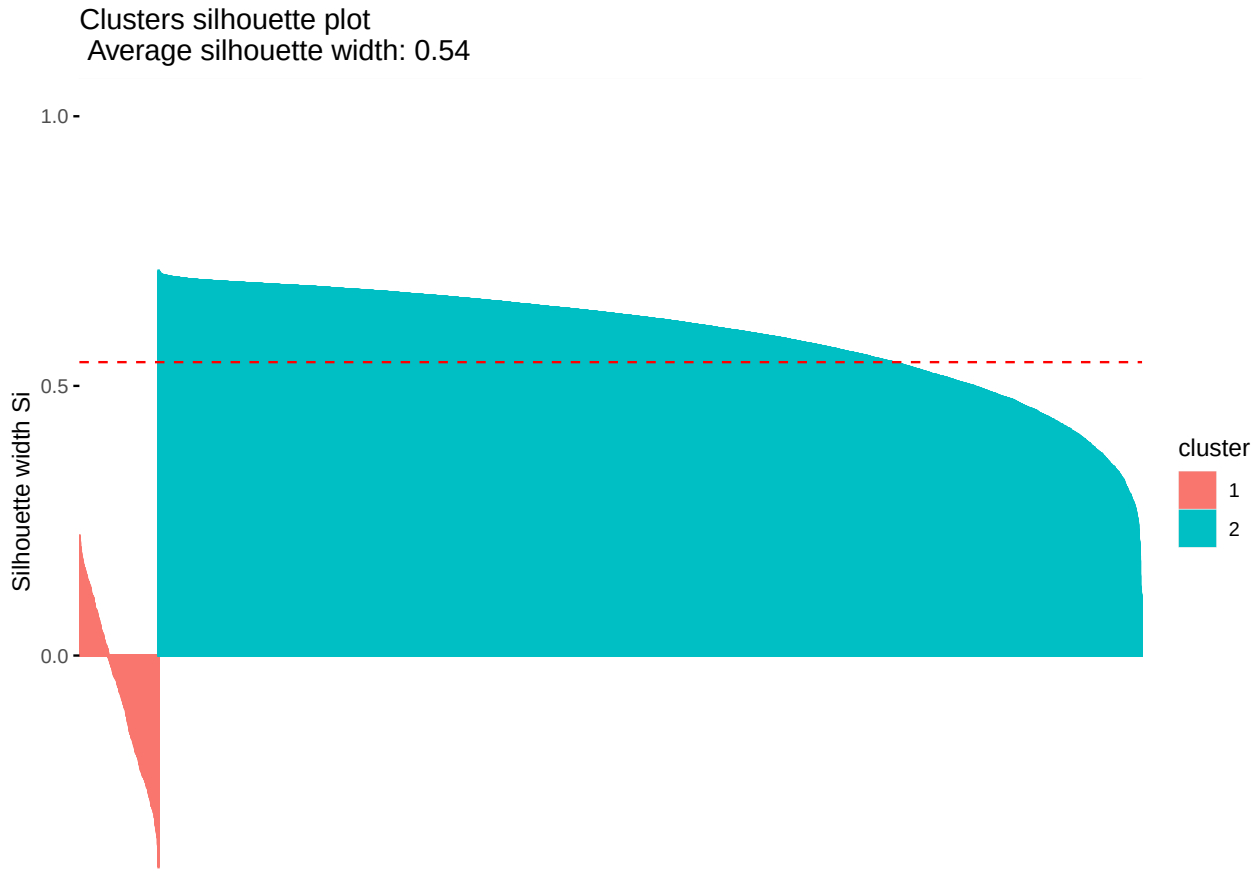
```
## [1] "prox_idx_health"
## [1] 0.014
## [1] "prox_idx_grocery"
## [1] 0.0714
## [1] "prox_idx_educpri"
## [1] 0.15545
## [1] "prox_idx_educsec"
## [1] 0.1126
## [1] "prox_idx_lib"
## [1] 0.0991
## [1] "prox_idx_parks"
## [1] 0.06895
## [1] "prox_idx_transit"
## [1] 0.0222
```

Silhouette Plot

```
# plt = cluster::silhouette(res$clusters, dist(res$complete_data))
# plot(plt, col = 1:4)
# abline(v=mean(plt[,3]), col="red", lty=2)

sil = silhouette(res$clusters, dist(res$complete_data))
fviz_silhouette(sil)
```

```
##   cluster size ave.sil.width
## 1       1  743         -0.06
## 2       2 9227          0.59
```



Cluster Profiles

```
#
```

Conclusion

text

Linked with Index of Remoteness

Implementation

```
#
```

Cut-off Values

```
#
```

Silhouette Plot

```
#
```

Cluster Profiles

```
#
```

Conclusion

text