A Two-Step Approach of Using Cluster Correspondence Analysis and Reduced K-means for Low Back Pain Patients Clustering

Fengmei Liu, Suchara Gupta, Cristina Tortora

Department of Mathematics & Statistics, San José State University, California, USA.
E-mail: fengmei.liu@sjsu.edu, sucharu7115@gmail.com, cristina.tortora@sjsu.edu

Abstract

In this report, we state a two-step approach for the IFCS 2017 data challenge regarding low back pain(LBP) patients clustering. Referring the (Nielsen, 2017)[3] result, we did the domain clustering in the first step using cluster correspondence analysis (clusCA). Using the output variables from each domain, we did the second step - Reduced K-means clustering. In the result part, we showed the final clustering result of this two-step approach and a profile plot of these clusters. Every cluster is highly interpretable and evaluated well with the first 10 variables in the data.

Introduction

We will talk about the data preprocessing in Section 1, introduce our clustering methods in Section 2 and show the results and evaluation in Section 3.

1 Data Pretreatment

In this data pretreatment part, we mainly introduce the variable selection and missing data imputation. All the variable indices are the same as the varaible file in the data challenge folder.

1.1 Data Summary

From the provided variable file, there are 122 variables totally. As the first 10 of them will not be considered for the purpose of clustering, we summarized the types of the rest 112 variables and their domains. Here are the tables show the details of this summary.

Variable Type	Count
Continuous	8
Dichotomous	64
Multistate nominal	9
Ordinal	30
Trichotomous	1

Table 1: Variable Type Summary

Variable Domain	Counts
Activity	23
Contextual factors	16
Pain	14
Participation	8
Physical impairment	24
Psychological	27

Table 2: Variable Domain Summary

1.2 Special Missing Data Preprocessing

For Variables from 67 (Fabq60) To 75 (Fabq140) value 0 is imputed for NA (Missing values) as for these variables Students, Unemployed, Pensioners are not eligible to answer the Question(Pain caused by work/accident at work place)
For variable 86 - 89 appropriate values are imputed for patients who doesn't have dominating Back Pain.

1.3 Variable Selection

Criteria for not including variables in our Analysis:

- First 10 variables not included in analysis but used to inform the insights.
- Variables with more than 20% missing data are not included: 91 and 98
- Summary Scores variables are not included : 85,122
  (Nilsen 2016)[4] has pointed out that they got better result by using single items in the questionnaires than using the summary scores, so here we adopted this idea and use single items from the questionnaires
- If variables having 85% or more than 85% one level for each Variable.
  Variables with 85% or more than 85% one dominate level are : 35,57,69,93,94,95,100,109,111,112,113,114,115

1.4 Missing Value Imputation

After all this preprocessing we are left with 95 variables to do analysis.
On top of these 95 Variables we did imputation Using MICE package and RANDOM FOREST method in R language.

2 Cluster analysis

2.1 Introduction of Cluster Correspondence Analysis and Reduced K-means

- Cluster Correspondence Analysis
  Cluster Correspondence Analysis (clusCA) is a method for joining dimension reduction and clustering analysis for categorical data. The related formulas and details are in the Cluster Correspondense Analysis (van de Velden, 2016) [5, 2]. In this article, it's compared to other variations of correspondence analysis including GROUPALS, MCA K-means and i-FCB, and gave a better result in the joint of dimension reduction and clustering. Also, the related R package **clustrd** gives good visualization of the result for clusCA method.
- Reduced K-Means
  Reduced K-means is a popular subspace clustering method, it's designed to maximize the between clusters deviance. Equivalently, it looks for A, M, U to minimize

$$F_{rkm}(A, M, U) = ||X - UMA'||^2 \tag{1}$$

  where X is data, A is the reduced subspace, U is the reduced space membership, and M is the reduced space centroid[1].

2.2 STEP 1 - Cluster Correspondence Analysis

2.2.1 Change Continuous variables to be Categorical

We treated all the variables as categorical. For the continuous variables, we change them to categorical based on quantiles. These variables are Age, Bhoej0(Height), Vasl0(LBP intensity), Okon0(Able to decrease pain), Obeh0(Treatment not essential), Htil0(Self-rated general health) , and bmi.

2.2.2 clusCA on 6 domains

After transforming all the variables to categorical, we did cluCA based on 6 domains.
The goal is to extra the main components that represent each domain. This step was done using a tuning process, by specifying the groups to be 3:12, and reduced dimensions to be 2:9, and using the criterion of average silhouette width to choose the best reduced dimensions and number of clusters. To interpret and use the result, we mainly utilize the variable component information rather than the clustering information.

- First, we looked into the main biplot, and summarize the main variables that contribute to the positive(H) and negative(L) direction of selected components;
- Secondly, the feature plots of each cluster further showed and helped the variable interpretation from the main biplot. They show the top 20 variables in each cluster, and positive sign means the variable frequency is above average in the cluster, while negative sign means the variable frequency is below average in the cluster.

Below are the output and analysis for the 6 domains. As an illustration of the notation, "HX1" means the positive direction of the first component from the contextual factor domain. "LX4" is the negative direction of the second component from the activity domain.

- **Contextrual Factor domain**
  The output has 4 clusters and 2 components, details of each component are:
  - HX1 : male, age34-43, tall, no chronic disease, full-time work, low education
  - LX1 : female, old, short, has musculoskel/other chronic disease, retired, high education
  - HX2: female, young, tall, no other chronic disease, work, high education.
  - LX2: male, old(52-66), average height, has musculoskel/other chronic disease,(low education)

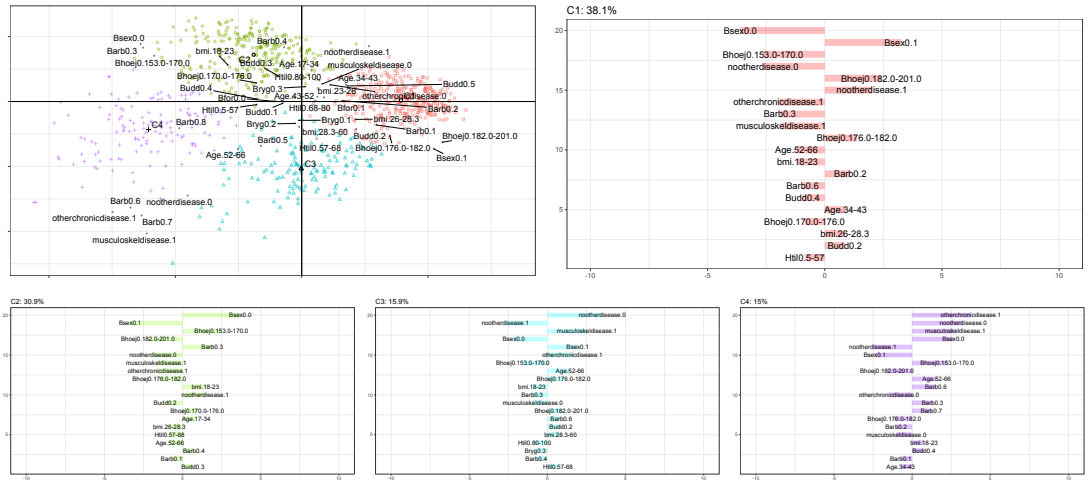

Figure 1: clusCA Output of Contextrual Factor domain

- **Activity domain**
  The output has 3 clusters and 2 components, details of each component are:
  - HX3: home activity slowly, self dress slowly
  - LX3: home activity normal, self dress normal
  - HX4: walk short distance, stand for short time, self-dress normal

– LX4: walk normal, stand normal, self-dress slowly

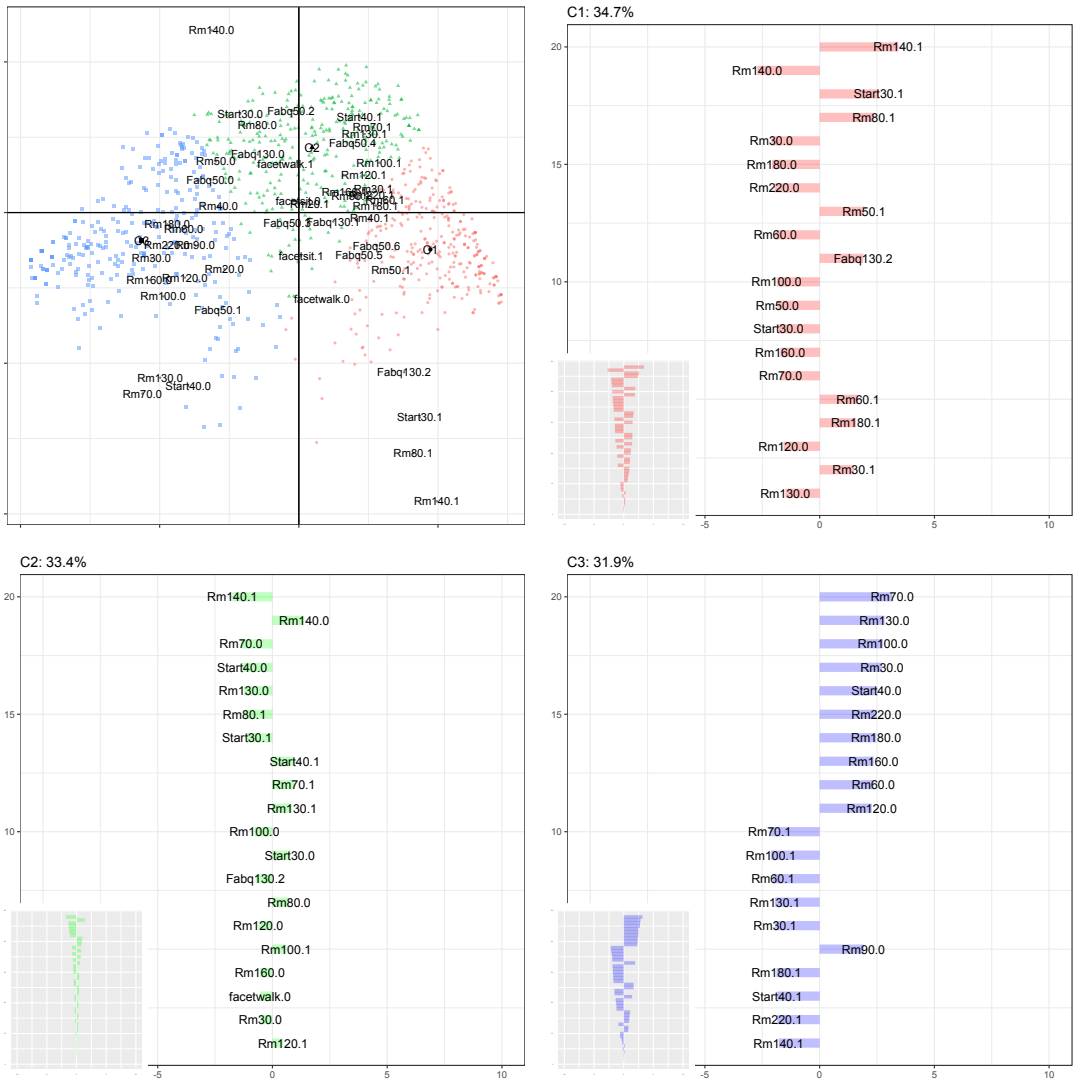

Figure 2: clusCA Output of Activity domain

- **Pain domain**
  The output has 3 clusters and 2 components, details of each component are:
  - HX5: pain has spred dwon to legs, leg pain is intense
  - LX5: pain has not spred down to legs, no leg pain
  - HX6: LBP pain intensity is high, this pain episode last short
  - LX6: LBP pain intensity is low, this pain episode last long

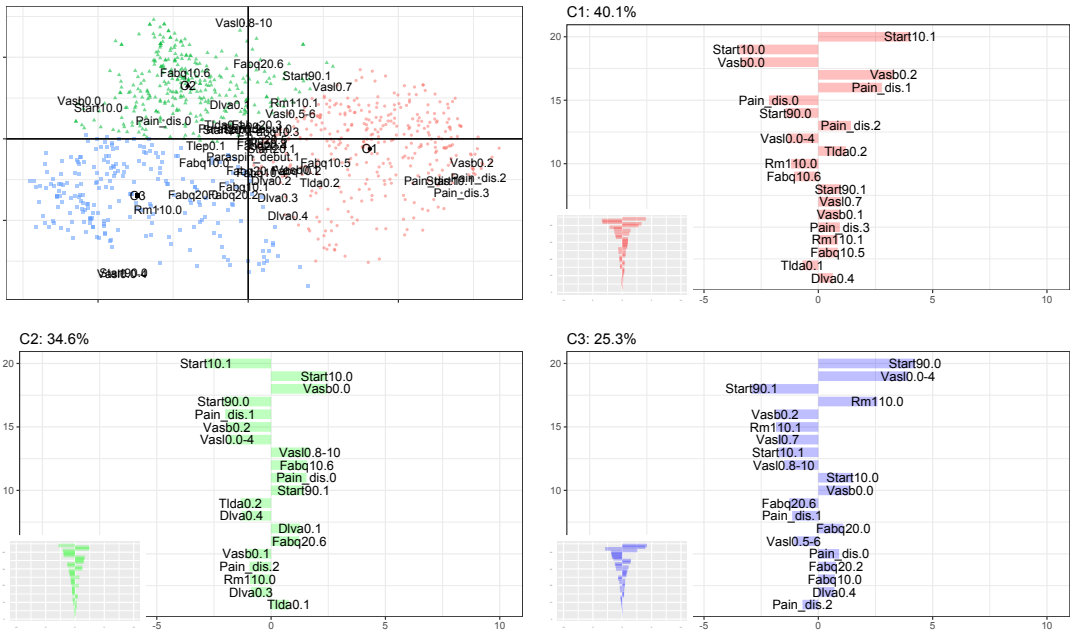

Figure 3: clusCA Output of Pain domain

- **Particification domain**
  The output has 4 clusters and 2 components, details of each component are:
  - HX7: work not make pain worse, phisical work load sitting/walking
  - LX7: work make pain worse, heavey phisical work
  - HX8: + work is heavey, more sick leave and stay home time.
  - LX8 - unsure if work is heavy, less sick leave and stay home time

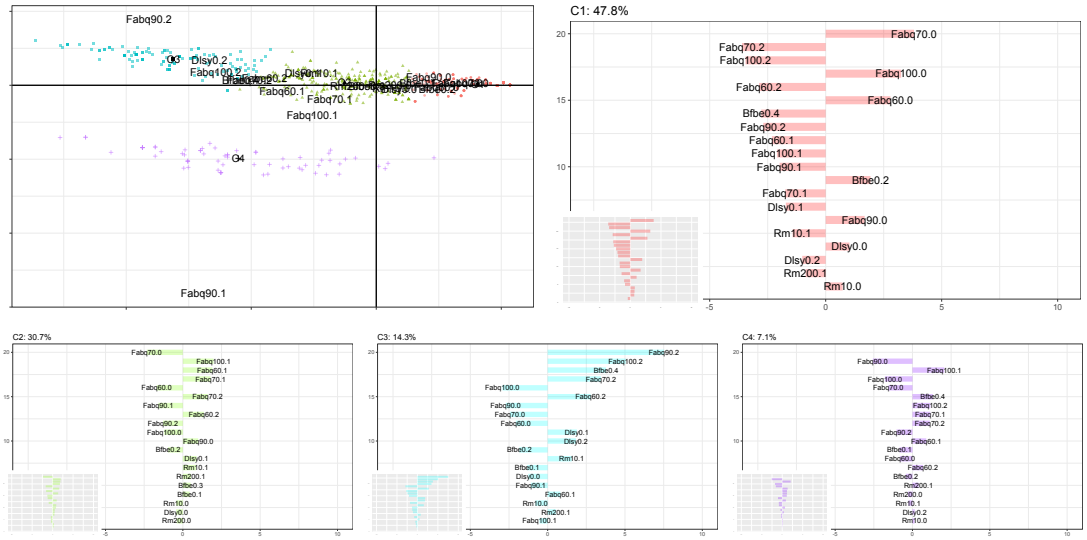

Figure 4: clusCA Output of Contextrual Participation domain

- **Physical Impairment domain**
  The output has 3 clusters and 2 components, details of each component are:
  - HX9: + no pain on AROM
  - LX9: leg pain on AROM test
  - HX10: negative on SI-joint tests, no pain on palpation
  - LX10: positive on SI-joint test, back pain on AROM test

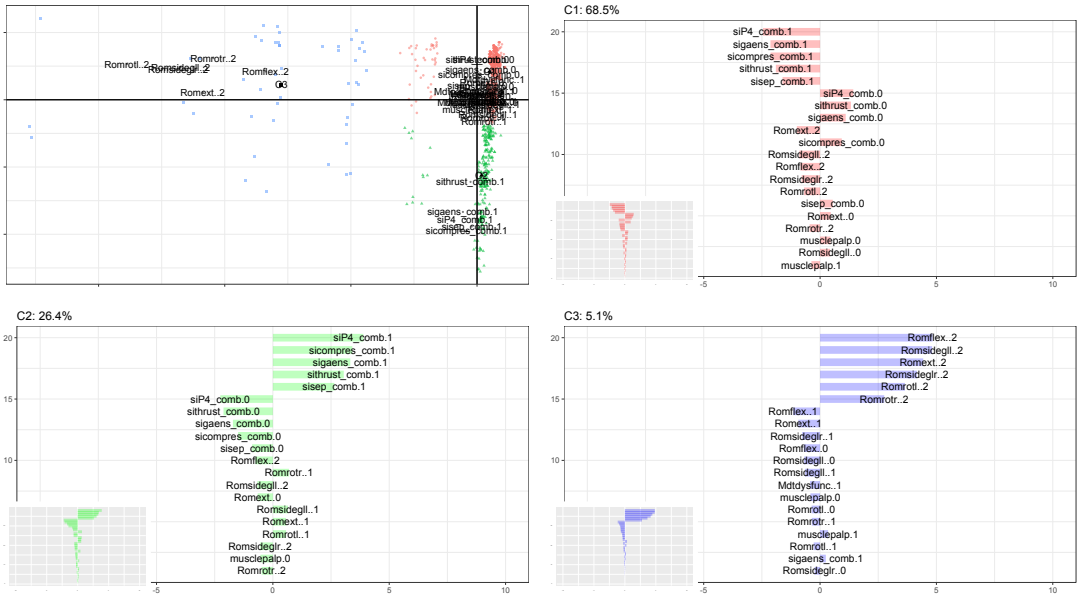

Figure 5: clusCA Output of Physical Impairment domain

- **Psychological impairment domain**
  The output has 3 clusters and 2 components, details of each component are:
  - HX11: good mode and feel energy, good conscience, good sleep
  - LX11: bad mode and feel less energy, bad conscience, bad sleep
  - HX12: not lose interest in daily activities, psychologically believe should do activity
  - LX12: lose interest in daily activities, spycho believe should not do activities

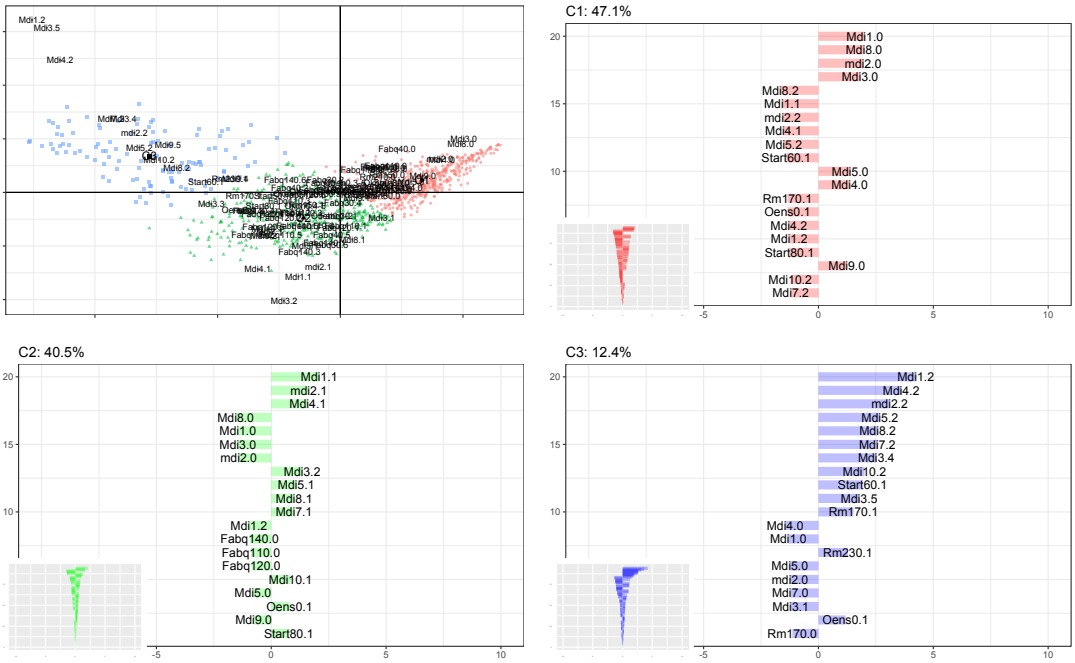

Figure 6: clusCA Output of Psychological Impairment domain

2.3 STEP 2 Reduced K-Means

- **Input for RKM**
  The new component output of clusCA are now in the euclidean space. So the input for reduced K-means are the combined output of X1 - X12 from above clusCA results of 6 domains.

- **Choose the number of clusters**
  Chose the number of clusters based on Within Variance, Silhouette width and Calinski-Harabasz(CH) index. We prefer a lower within variance, and higher Silhouette width and CH index.

  Here, we would like to use all the 12 dimensions instead of any dimension reduction. We also compared the reduced dimension with the full dimension results, the overall measure if better with the result of full dimension. We still used the self tuning RKM function in R package **clustrd** to choose the best clusters and dimensions, by specifying groups to be 2:20, and dimensions to be 12. From the plots of the 3 measures, we can tell that groups with range 5-8 are OK, we tested several of the group options and finalized with 8 groups. Also, we think it will make clusters easier to interpret to use a larger number of groups.

  Note: to make the graph compact, we put all three measures in one plot, with rescaling within variance/10000, Silhouette width*3, and CH index/300.

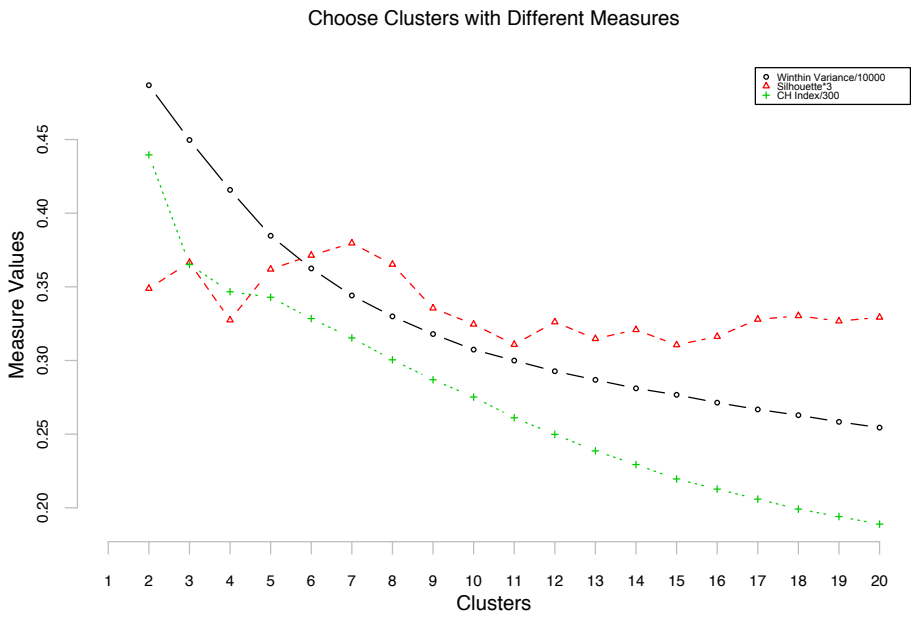

Figure 7: Using 3 Measures to Choose Number of Clusters

3 Results

3.1 Clustering Results

The clustering results are show in this profile plot and the interpretations of all clusters are given below.

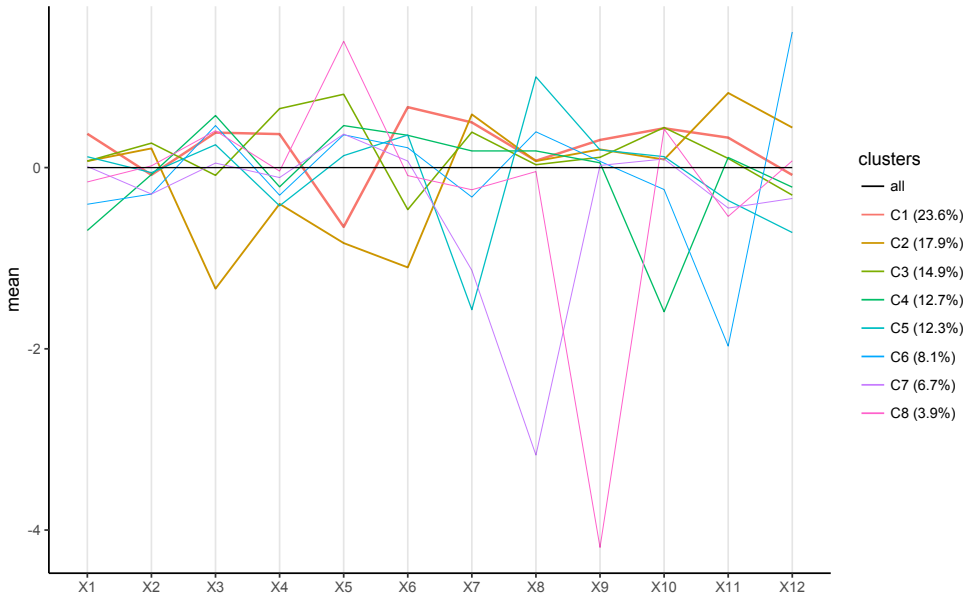

Figure 8: Final Clustering Profile Plot

Cluster	Percentage	Main Features	Interpretation features are separated by ";"
C1	23.6%	HX1, HX4, LX5, HX6, HX7, HX9, HX10, HX11	Mostly male, age34-43, no chronic disease, full-time work; Walk short distance, stand for short time, self-dress normal; Pain has not spread down to legs; LBP pain intensity is high; This pain episode last short; Work not make pain worse, physical work load sitting/walking; no pain on AROM test; Negative on SI-joint tests, no pain on palpation; Good mode and feel energy, good sleep;
C2	17.9%	HX2, LX3, LX4, LX5, LX6, HX7, HX11, HX12	Mostly female, young, no other chronic disease, work, high education; Home activity normal; Walk normal, stand normal; Pain has not spread down to legs, no leg pain; LBP pain intensity is low, this pain episode last long; Work not make pain worse, physical work load sitting/walking; Good mode and feel energy, good conscience, good sleep; Not lose interest in daily activities, psychologically believe should do activity
C3	14.9%	HX2, HX4, HX5, LX6, HX10	Walk short distance, stand for short time, self-dress normal; Pain has spread down to legs, leg pain is intense; LBP pain intensity is low, this pain episode last long; Negative on SI-joint tests, no pain on palpation
C4	12.7%	LX1, HX3, HX6, LX10	Mostly female, old, short, has musculoskel/other chronic disease, retired, high education; Home activity slowly, self dress slowly; LBP pain intensity is high, this pain episode last short; Positive on SI-joint test, back pain on AROM test
C5	12.3%	HX1, LX4, HX6, LX7, HX8, LX12	Mostly male, age34-43, tall, no chronic disease, full-time work, low education; Walk normal, stand normal, self-dress slowly; LBP pain intensity is high, this pain episode last short; Work make pain worse, heavy physical work; Work is heavy, more sick leave and stay home time; Lose interest in daily activities, pycho believe should not do activities
C6	8.1%	LX1, LX2, LX10, LX11, HX12	Old people, has musculoskel/other chronic disease, retired; Positive on SI-joint test, back pain on AROM test; Bad mode and feel less energy, bad conscience, bad sleep; Not lose interest in daily activities, psychologically believe should do activity
C7	6.7%	LX2, LX7, LX8	Mostly male, old(52-66), average height, has musculoskel/other chronic disease,(low education); Work make pain worse, heavy physical work; Unsure if work is heavy, less sick leave and stay home time
C8	3.9%	HX5, LX9, HX10	Pain has spread down to legs, leg pain is intense; Leg pain on AROM test; Negative on SI-joint tests, no pain on palpation;

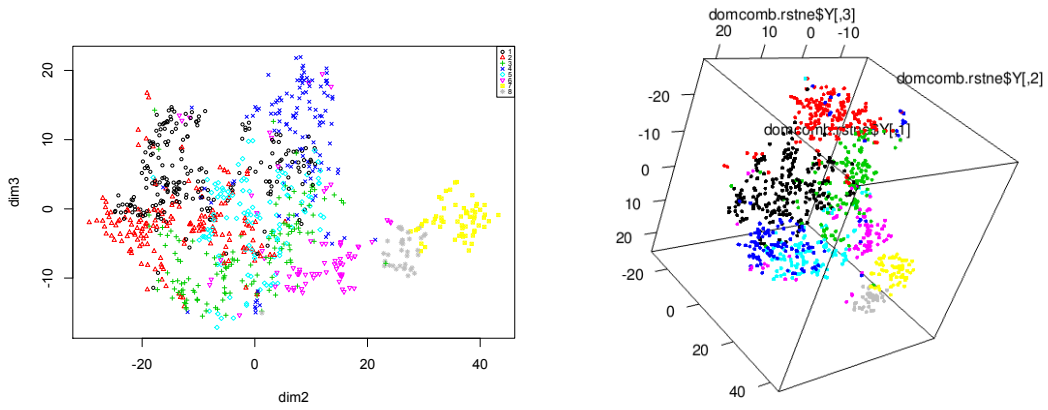

Figure 9: Cluster Results 2D and 3D Visualization with Rtsne

Using the 12 dimensions, we did the clustering visualization by dimension reduction method TSNE. Clusters are shown in 3d visualization and they are well separated, as shown in Fig 9.

3.2 Evaluation

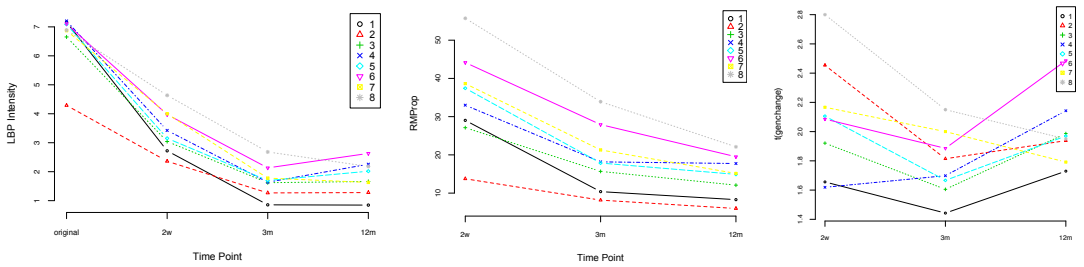

Figure 10: Clustering Evaluation with First 10 Variables

By using the first 10 variables, we compare the clustering results with LBP Intensity (vasl2w, vasl3m, vasl12m), left in Fig 10, RM summary score(rmprop2w, rmprop3m, rmprop12m), middle in Fig 10, global perceived improvements(gen2w, gen3m, gen12m), right in Fig 10.

Two important evaluation qualities we can highlight from here.

- **Out clusters are interpretable and meaningfully match with the evaluation variables**

  In the left plot of LBP intensity, cluster 1, 2 are lowest, in our results, cluster1 is middle-ages men who have intense LBP with a short duration, so it healed quickly and dropped down fast.

  cluster2 is group of young women with light LBP, so the intensity drop is not large but it belongs to the lowest generally. On the contrary, cluster 6 is the group of old people with pain and positive SI-joint and AROM test, and cluster 8 is the group that pain has spread to legs intensely. So their LBP pain is average higher than other group at all the time points.

  Similarly, for the middle plot of RM summary score, cluster 1 and 2 shown as the lowest

and cluster 8 and 6 are the higher than other groups.

We couldn't quite understand the meaning of variable "global perceived improvements". So we don't discuss this variable here, but show the result in the right plot of Fig 10.

So, overall, the interpretation of the clustering results matches with the evaluation,

- **All the clusters are well separated at 2weeks, 3months and 12months at each variable**

  This quality indicates a good separation of the observations.

So we think the results will be useful for the LBP diagnostic purpose.

References

[1] Geert De Soete and J Douglas Carroll. K-means clustering in a low-dimensional euclidean space. In *New approaches in classification and data analysis*, pages 212–219. Springer, 1994.

[2] Michael Greenacre and Jorg Blasius. *Multiple correspondence analysis and related methods*. CRC press, 2006.

[3] Anne Molgaard Nielsen, Peter Kent, Lise Hestbaek, Werner Vach, and Alice Kongsted. Identifying subgroups of patients using latent class analysis: should we use a single-stage or a two-stage approach? a methodological study using a cohort of patients with low back pain. *BMC musculoskeletal disorders*, 18(1):57, 2017.

[4] Anne Molgaard Nielsen, Werner Vach, Peter Kent, Lise Hestbaek, and Alice Kongsted. Using existing questionnaires in latent class analysis: should we use summary scores or single items as input? a methodological study using a cohort of patients with low back pain. *Clinical epidemiology*, 8:73, 2016.

[5] Michel van de Velden, A Iodice D'Enza, and F Palumbo. Cluster correspondence analysis. *Psychometrika*, pages 1–28, 2014.

Appendix: R-code

```
#*********************************
#*   IFCS Data Challenge
#*  Team: Fengmei Liu, Sucharu Gupta
#*  Date: 05082017
#*********************************

# libraries needed, pls install them first if you don't have them
#install.packages("missForest", dependencies=TRUE)
#install.packages("mice", dependencies=TRUE)
library(gdata)
library(mice) # for data imputation
library(randomForest) # for data imputation
library(VIM) # for data imputation
library(data.table) # for data processing
library(dplyr) # for data processing
library(missForest) # for imputation
library(Rtsne) #for visualization
library(rgl) # for 3d visualization
```

```r
library(clustMD) # for model based clustering
library(irlba) # for svd visualization
library(Matrix)
library(fpc) # for cluster.stats
library(GGally) # for one of ggplot
library(clustrd) # for MCA
library(cluster) # for daisy-gower distance
library(clustMixType)


## Read in data
# setwd("~/SJSU_CIASSES/Math285_Clustering/project")
dataset <- xls2csv("data_challenge2.xlsx")
data <- read.csv(dataset, header=TRUE,strip.white=TRUE)
varsset <- xls2csv("variables_data_challenge2.xlsx")
vars <- read.csv(varsset, header= TRUE, strip.white=TRUE)
colnames(data) <- sapply(vars$Variable, as.character) # to make the variables names same

## ------------------------------------------------------------------
## explore variable -
##-------------------------------------------------------------------

unique(vars$Domain) # 6 domains (blank values for top 10 variables)

# Domain
vars[11:122,] %>% group_by(Domain) %>% dplyr::summarize(a=n())

#1              Activity      23
#2  Contextual factors      16
#3                  Pain      14
#4         Participation       8
#5 Physical\nimpairment      24
#6         Psychological      27

# Type
vars[11:122,] %>% group_by(Type) %>% dplyr::summarize(a=n())
#1          Continuous       8
#2         Dichotomous      64
#3 Multistate nominal       9
#4             Ordinal      30
#5       Trichotomous       1

# Group
vars[11:122,] %>%
  group_by(Baseline.questionnaire..BQ..or.phy..examination..PE..or.outcome..OU.
) %>% dplyr::summarize(a=n())
#1                                                                BQ    77
#2                                                                PE    35

# check missing values
par(mar = c(3,3,1,1))
hist(as.numeric(levels(vars$Missing ....[11: nrow(vars)]))[vars$Missing ....[11: nrow(vars)]],
  breaks=40, right=FALSE, xlim = c(0,40), ann=FALSE, axes=FALSE,col = "royalblue")
axis(1, font.axis=1 , cex.axis=0.75, family = "sans")
```

```r
axis(2, font.axis=1, cex.axis=0.75, family = "sans")
title(main="Missing Proportion Distribution",font.main = 1, cex.main= 1, family="mono")
    # create title
mtext("Missing Proportion", 1, line=2, cex=1, family="sans") # create x label
mtext("Frequency", 2, line=2, cex=1, family="sans") # create y label

# >30: 2
# >20: 2
# >10: 11

##
    ------------------------------------------------------------------
#  Variables Processing
##------------------------------------------------------------------

###### Special missing values
# sum of missing and non-missing
which((vars$N + vars$Missing..N) != 928)
 # 67 68 69 70 71 72 73 74 75 86 87 88 89 98 need extra treatements

# 1) Create extra level for 67 68 69 70 71 72 73 74 75
colnames(data)[67:75] # see if the variables are what we want
stumiss <- which(data[,14] %in% c(4,5,6,7))
for (i in (67:75)){
   data[stumiss,i][is.na(data[stumiss,i])] <- 0
}
#data[stumiss,67:75] # check

# 2) Create extra level for 86 87 88 89
colnames(data)[86:89] # see if the variables are what we want
dbpmiss <- which(data[,100] %in% c(1))
# make 9 non-BP to be 1
data[dbpmiss,86][is.na(data[dbpmiss,86])] <- 1
for (i in (87:89)){
   data[dbpmiss,i][is.na(data[dbpmiss,i])] <- 0
}

##### Seperate ordinal to categorical and continuous
## group to categorical and continuous
# variable #18 we treat it as categorial
catvars <- c(which(vars$Type %in% c("Dichotomous","Multistate
    nominal","Trichotomous")),18)
bivars <- which(vars$Type %in% c("Dichotomous"))
nomivars <- c(which(vars$Type %in% c("Multistate nominal","Trichotomous")),18)
ordcontivars <- which(vars$Type %in% c("Ordinal","Continuous"))[-8]
ordvars <- which(vars$Type %in% c("Ordinal"))[-3]
contivars <- which(vars$Type %in% c("Continuous"))
# make categorical to be the right type
data[,catvars] <- lapply(data[,catvars], factor)
data[,ordvars] <- lapply(data[,ordvars], ordered)

#### summary scores will not incldued: 85, 122
#### missing values over 20% not included: 91, 98
#### check if any level dominate one variable, using 85% cutoff line
# exclude: 35 57 69 93 94 95 100 109 111 112 113 114 115
```

```r
nonconti.ind <- which(vars[1:122,]$Type !="Continuous")
maxprop <- NULL
for (i in nonconti.ind){
    maxprop[i] <- (max(summary(as.factor(data[,i]))))/982)
}
print(which(maxprop >= 0.85))
vars$Variable[which(maxprop >= 0.85)]
# Start70, Rm190 , heartdisease, asthma, psychdisease, Domin_bp, Mdtnonreduce, Herndiscr,
    Herndiscl, Affstrenght, Affsens, Affdtr
# 35  57  69 93  94  95 100 109 111 112 113 114 115

## Select the features based on above analysis
    ----------------------------
delvars <- c(1:10, 85, 122,91, 98, 35, 57 , 69 ,93 , 94,  95 ,100 ,109 ,111 ,112 ,113
    ,114, 115)
data.new <- data[,-delvars]
# dim(data.new) # 928 95

##
    ------------------------------------------------------------------------
# Impute using MICE random forest
##-------------------------------------------------------------------------
library(mice)
library(randomForest)
library(VIM)
# !!!!! below code will take few minutes
imp.data <- mice(data.new, m=1, maxit=5, meth=c("rf"), seed=500)
# summary(imp.data)
comp.data <- complete(imp.data,1)

##
    ------------------------------------------------------------------------
# Clustering - 1. Transform all to categorical - clusCA
##-------------------------------------------------------------------------
bivars2 <- which(colnames(comp.data) %in% vars$Variable[bivars])
nomivars2 <- which(colnames(comp.data) %in% vars$Variable[nomivars])
ordvars2 <- which(colnames(comp.data) %in% vars$Variable[ordvars])
contivars2 <- which(colnames(comp.data) %in% vars$Variable[contivars])

mca.data <- comp.data
## transform continuous variables to categorical based on quantiles.
# lapply(mca.data[, contivars2 ], summary)

# Age
mca.data$Age <- ordered(cut(mca.data$Age, c(17, 34, 43, 52,66),include.lowest= TRUE))
levels(mca.data$Age) <- c("17-34", "34-43","43-52","52-66")
# Bhoej0/Height
mca.data$Bhoej0 <- ordered(cut(mca.data$Bhoej0,
    c(153.0,170.0,176.0,182.0,201.0),include.lowest= TRUE))
levels(mca.data$Bhoej0) <- c("153.0-170.0","170.0-176.0","176.0-182.0","182.0-201.0")
# Vasl0/LBP intensity
mca.data$Vasl0 <- ordered(cut(mca.data$Vasl0, c(0,5,7,8,10), include.lowest= TRUE))
levels(mca.data$Vasl0) <- c("0-4", "5-6","7","8-10")
# Okon0/Able to decrease pain
mca.data$Okon0 <- ordered(cut(mca.data$Okon0, c(0,2,4,6,10),include.lowest= TRUE))
```

```r
levels (mca.data$Okon0) <- c("0-1", "2-3","4-5","6-10")
# Obeh0/Treatment not essential
mca.data$Obeh0 <- ordered(cut(mca.data$Obeh0, c(0,1,3,6,10),include.lowest= TRUE))
levels (mca.data$Obeh0) <- c("0","1-2", "3-5","6-10")
# Htil0/Self-rated general health
mca.data$Htil0 <- ordered(cut(mca.data$Htil0, c(5, 57, 68, 80,100), include.lowest= TRUE))
levels (mca.data$Htil0) <- c("5-57", "57-68","68-80","80-100")
# bmi/BMI
mca.data$bmi <- ordered(cut(mca.data$bmi, c(18, 23, 26, 28.3,60), include.lowest= TRUE))
levels (mca.data$bmi) <- c("18-23", "23-26","26-28.3","28.3-60")


# Try clusCA on all data
# allmca.out <- tune_clusmca(mca.data,nclusrange=3:12, ndimrange= 2:9, method="clusCA",
    criterion= "asw")

allmca.out <- tune_clusmca(mca.data,nclusrange=3, ndimrange= 2, method="clusCA",
    criterion= "asw")
plot(allmca.out$clusmcaobj,cludesc=TRUE)

## ------------------------------------------------------------------
# Clustering - 2. clusCA -Domain Clustering
##-------------------------------------------------------------------

contextual_vars <- which(colnames(comp.data) %in% vars$Variable[vars$Domain ==
    "Contextual factors"])
activity_vars <- which(colnames(comp.data) %in% vars$Variable[vars$Domain ==
    "Activity"])
pain_vars <- which(colnames(comp.data) %in% vars$Variable[vars$Domain == "Pain"])
parti_vars <- which(colnames(comp.data) %in% vars$Variable[vars$Domain ==
    "Participation"])
physicalimp_vars <- which(colnames(comp.data) %in% vars$Variable[vars$Domain ==
    "Physical\nimpairment"])
psycho_vars <- which(colnames(comp.data) %in% vars$Variable[vars$Domain ==
    "Psychological"])

###### STEP1: each domain fit clusCA clustering ------------

### Contextrual Factor domain ----------
# factors: 1  2  3  4  5  7  8 27 77 78 79 95
# names: "Bsex0","Age", "Budd0","Barb0", "Bfor0","Bhoej0","Bryg0","Htil0",
    "nootherdisease", "musculoskeldisease"," otherchronicdisease", "bmi"
contextual.data <- mca.data[,contextual_vars]
#contex.out <- tune_clusmca(contextual.data,nclusrange=3:12, ndimrange= 2:9,
    method="clusCA", criterion= "asw")
contex.out <- tune_clusmca(contextual.data,nclusrange=4, ndimrange= 2, method="clusCA",
    criterion= "asw")
plot(contex.out$clusmcaobj,cludesc=TRUE, what=c(TRUE, TRUE))
# 4 clusters, 2 dims
#       2      3      4      5      6      7      8      9
#3 0.186
#4 0.211 0.186
#5  0.17 0.199 0.181
#6 0.138 0.158 0.197 0.178
```

```
#7 0.107 0.133 0.152 0.194 0.177
#8 0.039 0.075 0.147 0.156 0.188 0.089
#9 0.039 0.066 0.092 0.139 0.087 0.115 0.124
#10 0.036 0.05 0.077 0.074 0.105 0.109 0.069 0.092
#11 0.034 0.066 0.053 0.075 0.066 0.067 0.103 0.127
#12 0.022 0.049 0.061 0.055 0.088 0.111 0.094 0.113

# Important factors: "Bsex0","Age", "Budd0","Barb0","Bhoej0","nootherdisease",
    "musculoskeldisease","otherchronicdisease"
# 1 2 3 4 8 77 78 79
#
# X1: + male, age34-43, tall, no chronic disease, full-time work, low education
#       - female, old, short, has musculoskel/other chronic diseas, retired, high education
#
# X2: + female, young, tall, no other chronic disease, work, high education.
#      - male, old(52-66), average height, has musculoskel/other chronic disease,(low
    education)
#
# cluster1: male, age34-43, tall, no other chronic disease, full-time work, lower education
# cluster2: female, age 17-34, (women average height), no other chronic disease,
    part-time work/student, higher education
# cluster3: male, old(52-66), average height, has musculoskel/other chronic disease,(low
    education)
# cluster4: female, old(52-66), (average height),has musculoskel/other chronic disease,
    retired, high education


sel_contxt_vars <- c("Bsex0","Age", "Budd0","Barb0","Bhoej0","nootherdisease",
    "musculoskeldisease","otherchronicdisease")


### Activity domain ----------
# factors: 21 22 29 30 31 32 33 34 35 36 37 39 40 41 43 45 48 54 61 73 74
# names: "Start30" "Start40" "Rm20" "Rm30" "Rm40" "Rm50" "Rm60"
        "Rm70" "Rm80" "Rm90" "Rm100" "Rm120" "Rm130" "Rm140"
    "Rm160" "Rm180" "Rm220" "Fabq50" "Fabq130" " facetsit " "facetwalk"
activity.data <- mca.data[,activity_vars]
# activity.out <- tune_clusmca(activity.data,nclusrange=3:12, ndimrange= 2:9,
    method="clusCA", criterion= "asw")
activity.out <- tune_clusmca(activity.data,nclusrange=3, ndimrange= 2, method="clusCA",
    criterion= "asw")
plot( activity.out$clusmcaobj,cludesc=TRUE)
activity.out$ critgrid

# 3 clusters, 2 dims
#       2      3      4      5      6      7      8      9
#3 0.247
#4 0.177 0.203
#5 0.189 0.153 0.185
#6 0.123 0.169 0.169 0.095
#7 0.115 0.103 0.094 0.103 0.105
#8 0.113 0.103 0.124   0.1 0.085 0.087
#9 0.102 0.101 0.106  0.05 0.086 0.075 0.071
#10 0.091 0.099 0.068 0.069 0.037 0.038 0.052 0.049
#11 0.075 0.074 0.118 0.024 0.057 0.028  0.03 0.044
```

```
#12 0.062 0.063 0.068 0.025 0.024 0.034 0.013 0.037

# Important factors: "Rm30", "Rm70" ,"Rm80","Rm100" , "Rm130" ,"Rm140","Rm180",
    "Rm220", "Start30","Start40"

# X1: + home activity slowly, self dress slowly ,
#     - home activity normal, self dress normal
# X2: + walk short distance, stand for short time, self-dress normal
#     - walk normal, stand normal, self-dress slowly

# cluster1: walk short distances because of back pain, stand for short time, do less home
    activity
# cluster2: not walk short distances, stand long, get dressed slowly, trouble put on socks
# cluster3: dress normal, normal home activity


sel_activity_vars <- c("Rm30", "Rm70" ,"Rm80","Rm100" , "Rm130" ,"Rm140","Rm180",
    "Rm220", "Start30","Start40")

### Pain domain ----------

pain.data <- mca.data[,pain_vars]
# pain.out <- tune_clusmca(pain.data,nclusrange=3:12, ndimrange= 2:9, method="clusCA",
    criterion= "asw")
pain.out <- tune_clusmca(pain.data,nclusrange=3, ndimrange= 2, method="clusCA",
    criterion= "asw")
plot(pain.out$clusmcaobj,cludesc=TRUE)
# pain.out$ critgrid

# 3 clusters, 2 dims
#       2      3      4      5      6      7      8      9
#3 0.247
#4 0.177 0.203
#5 0.189 0.153 0.185
#6 0.123 0.169 0.169 0.095
#7 0.115 0.103 0.094 0.103 0.105
#8 0.113 0.103 0.124   0.1 0.085 0.087
#9 0.102 0.101 0.106  0.05 0.086 0.075 0.071
#10 0.091 0.099 0.068 0.069 0.037 0.038 0.052 0.049
#11 0.075 0.074 0.118 0.024 0.057 0.028  0.03 0.044
#12 0.062 0.063 0.068 0.025 0.024 0.034 0.013 0.037

# Important factors: "Dlva0","Tlda0", "Vasl0","Vasb0", "Start10", "Start90", "Pain_dis"
# X1 + pain has spred dwon to legs, leg pain is intense
# - pain has not spred down to legs, no leg pain
# X2 + LBP pain intensity is high, this pain episode last short
# - LBP pain intensity is low, this pain episode last long

# cluster1: pain has not spread down, leg pain 0, back pain only, very extreme backpain in
    last 2wks, this pain episode last 2 wks, pain caused#
# by phisical activity and phisical activity make it worse
# cluster2: pain has spred down to legs, leg pain is serious, back pain and pain in 1 or 2
    legs, last year has >30 days LBP, caused by physical # work, pain intensity is above
    medium, pain bothersome in last 2 weeks, back/leg pain almost all the time
```

```
# cluster3: pain not spread down, not caused by work, not very bothersome in the last 2
    wks, pain intensity is light, not pain all the time


sel_pain_vars <- c("Dlva0","Tlda0","Vasl0","Vasb0","Start10","Start90","Pain_dis")

### Particification domain ------------

parti.data <- mca.data[,parti_vars]
# parti.out <- tune_clusmca(parti.data,nclusrange=3:8, ndimrange= 2:7, method="clusCA",
    criterion="asw")
parti.out <- tune_clusmca(parti.data,nclusrange=4, ndimrange= 2, method="clusCA",
    criterion="asw")
plot(parti.out$clusmcaobj,cludesc=TRUE)
# parti.out$ critgrid

# 4 clusters, 2 dims
#      2      3      4      5      6      7
#3 0.261
#4 0.264 0.209
#5 0.164 0.215 0.167
#6 0.215 0.164 0.245 0.209
#7 0.132 0.143 0.167 0.128 0.075
#8 0.126 0.147 0.199 0.074 0.086 0.157

# Important factors:  "Bfbe0","Fabq70","Fabq90","Fabq100"

# X1: + work not make pain worse, phisical work load sitting/walking
#     - work make pain worse, heavey phisical work
# X2: + work is heavey, more sick leave and stay home time.
#     - unsure if work is heavy, less sick leave and stay home time.

# cluster1: work not make pain worse, work is not heavy, physical workload is
    sitting/walking
# cluster2: not sure if it's work make the pain worse(but prune to believe so), prune to
    heavy work load
# cluster3: work caused pain and make it worse, heavey work load, long sick leave last
    month, stay home most of the time
# cluster4: unsure if work is heavy(prune to believe so), but work make pain worse, heavy
    work load

sel_parti_vars <- c("Bfbe0","Fabq70","Fabq90","Fabq100")

### Physical impairment domain ------------

physi.data <- mca.data[,physicalimp_vars]
# physi.out <- tune_clusmca(physi.data,nclusrange=3:12, ndimrange= 2:9, method="clusCA",
    criterion="asw")
physi.out <- tune_clusmca(physi.data,nclusrange=3, ndimrange= 2, method="clusCA",
    criterion="asw")
plot(physi.out$clusmcaobj,cludesc=TRUE)
# physi.out$ critgrid
# 3 clusters, 2 dims
#        2      3      4      5      6      7      8      9
#3  0.219
```

```
#4    0.13   0.16
#5   0.122 0.146 0.147
#6    0.07 0.141 0.146   0.15
#7    0.07 0.107   0.11 0.149 0.145
#8   0.041 0.102 0.125 0.114 0.149 0.147
#9   0.026 0.091 0.101 0.105   0.11   0.14 0.138
#10 0.003 0.085 0.098 0.089 0.102 0.138 0.112 0.134
#11 −0.002 0.054 0.082 0.054 0.114 0.128 0.126 0.107
#12 −0.006 0.038 0.04 0.096 0.098 0.125 0.105 0.115

# Important factors:  "Mdtpartlyreduce",
    "Romrotl","siP4_comb","sicompres_comb","sigaens_comb", "Mdtdysfunc","Romext",
    "Romsidegll", "Romflex"

# X1:  + no pain on AROM,

#       − leg pain on  AROM test

# X2: + negative on SI−joint tests, no pain on palpation
#     − postive on SI−joint test, back pain on AROM test

# cluster1: negative on SI−joint tests, no pain on palpation, no pain on AROM,
# cluster2: postive on SI−joint test, back pain on AROM
# clsuter3: leg pain on AROM test

sel_physi_vars <− c("Mdtpartlyreduce",
    "Romrotl","siP4_comb","sicompres_comb","sigaens_comb", "Mdtdysfunc","Romext",
    "Romsidegll", "Romflex")

### Psychological impairment domain ––––––––––––
# psycho_vars
# factors: 15 16 17 18 23 24 25 42 44 47 49 52 53 59 60 62 63 64 65 66 67 68 69 70 71
# names: "Okon0" "Okom0" "Oens0" "Obeh0" "Start50" "Start60" "Start80" "Rm150"
    "Rm170" "Rm210" "Rm230" "Fabq30" "Fabq40" "Fabq110" "Fabq120" "Fabq140"
    "Mdi1" "mdi2" "Mdi3" "Mdi4" "Mdi5" "Mdi7" "Mdi8" "Mdi9" "Mdi10"
psycho.data <− mca.data[,psycho_vars]
# psycho.out <− tune_clusmca(psycho.data,nclusrange=3:12, ndimrange= 2:9,
    method="clusCA", criterion= "asw")
psycho.out <− tune_clusmca(psycho.data,nclusrange=3, ndimrange= 2, method="clusCA",
    criterion= "asw")
plot(psycho.out$clusmcaobj,cludesc=TRUE)
# psycho.out$critgrid

# 3 clusters, 2 dims
#         2       3       4       5       6       7       8       9
#3   0.154
#4   0.076   0.142
#5   0.061   0.066   0.056
#6   0.024   0.056    0.05   0.049
#7   0.015   0.058   0.033   0.046    0.04
#8  −0.009  0.026   0.009   0.031  0.025  0.031
#9   −0.04  0.014    0.03   0.027   0.02  0.024  0.017
#10 −0.017 0.007 −0.003  0.018  0.006  0.021  0.017   0.01
#11 −0.029 −0.006  0.01   0.021  0.002  0.037  0.018  0.006
#12 −0.045 −0.016     0 −0.004  0.012 −0.02  0.006  0.008
```

```r

# Important factors:  "Fabq120","Fabq140", "Mdi1" ,"mdi2","Mdi3" ,"Mdi4","Mdi5", "Mdi8"

# X1 + good mode and feel energy, good conscience, good sleep
#      - bad mode and feel less energy, bad conscience, bad sleep
# X2 + not lose interest in daily activities , psychologically believe should do activity
#       - lose interest in daily activities , spycho believe should not do activities
# cluster1: feel good and energy most of the time, good conscience, good sleep, should do
    some activity
# cluster2: feel bad and less energy some of the time, some guilty, socaliy isolated
# cluster3: feel bad and less energy most of the time, feel restless , have trouble sleep

sel_psycho_vars <- c("Fabq120","Fabq140", "Mdi1" ,"mdi2","Mdi3" ,"Mdi4","Mdi5",
    "Mdi8")

##
    ----------------------------------------------------------------
# Clustering - 3. RKM - ALL Clustering
##----------------------------------------------------------------------

###### STEP2: use the output each domain as input of new RKM clustering
    ----------------------------
## functions from R library clustrd  https://github.com/cran/clustrd/tree/master/R
# 1) from https://github.com/cran/clustrd/tree/master/R
clu_means<-function(x, id, disp=TRUE, center=TRUE, scale=TRUE){



  clu = NULL

  funs = NULL



  x = data.frame(scale(as.matrix(x), center = center, scale = scale))



  p=ncol(x)

  gm=apply(x,2,mean)



  id=factor(id)

  csize=as.vector(table(id)/sum(table(id)))



  x$clu=id

  clum=(x %>% group_by(clu) %>% summarise_all(funs(mean)))


```

```
451
452 am=rbind(clum[,-1],gm)
453
454 bm=data.frame(t(am))
455
456 names(bm)=c(paste("C",1:nrow(clum),sep=""),"all")
457
458 bm$names=row.names(bm)
459
460
461
462 par_bm=data.frame(t(bm[-ncol(bm)]))
463
464 gnam=paste(names(bm)[-ncol(bm)]," (",round(csize*100,digits=1),"%",")",sep="")
465
466 #  cnm=paste(cnames,": ",round(csize*100,2),"%",sep="")
467
468
469
470 gnam[length(gnam)] = "all"
471
472 par_bm$clusters=gnam
473
474 par_bm$csize=c(csize,1/length( csize ))
475
476
477
478 gg_color_hue <- function(n) {
479
480   hues = seq(15, 375, length = n + 1)
481
482   hcl(h = hues, l = 65, c = 100)[1:n]
483
484 }
485
486
487
488 mypal=gg_color_hue(length( csize ))
489
490 mypal=c("black",mypal)
491
492
493
494 # if ( scale == T) {
495
496 #
      pco=ggparcoord(par_bm[1:(dim(par_bm)[1]-1),],columns=1:p,groupColumn=p+1,scale="globalminmax",mapping
      = ggplot2::aes(size = 5*csize))
497
498 # } else {
499
500 pco=ggparcoord(par_bm,,columns=1:p,groupColumn=p+1,scale="globalminmax",mapping =
      ggplot2::aes(size = 3*csize))
501
502 # }
```

```
503
504     pco=pco+scale_size_identity()
505
506     pco=pco+scale_colour_manual(values=mypal)
507
508
509
510     #  if (scale == T) {
511
512     #    pco=pco+geom_vline(xintercept=1:p,alpha=.5) + xlab("variables") + ylab("z-score")
513
514     # } else {
515
516     pco=pco+geom_vline(xintercept=1:p,alpha=.1) + xlab("") + ylab("mean") + theme_classic()
517
518     # }
519
520
521
522     return(pco)
523
524
525
526   }
527
528   # 2) from https://github.com/cran/clustrd/tree/master/R
529   outOfIndependence=function(data,Gvec,labs,nolabs=F,fixmarg=T,firstfew=0,minx=-2.5,maxx=2.5,segSize=4,textSize=6)
530
531     #  require(ggplot2)
532
533     #  require(dummies)
534
535     value = NULL
536
537     newplace = NULL
538
539      lbls = NULL
540
541
542
543     data=data.frame(data)
544
545     data=dummy.data.frame(data, dummy.classes = "ALL")
546
547
548
549     K=max(Gvec)
550
551     C=matrix(0,nrow(data),max(Gvec))
552
553     # print(dim(C))
554
555     for(j in 1:max(Gvec)){
556
557       C[which(Gvec==j),j]=1
```

```
558
559     }
560
561
562
563     P=t(data) %*% C
564
565     n=nrow(data)
566
567
568
569     #   B=t(data) %*% data
570
571     #
572
573     P=P/sum(P)
574
575
576
577     c=apply(P,2,sum)
578
579     c=t(t(c))
580
581     r=apply(P,1,sum)
582
583     r=t(t(r))
584
585
586
587     invsqDc=diag(as.vector(1/sqrt(c)))
588
589     invsqDr=diag(as.vector(1/sqrt(r)))
590
591     eP= r%*% t(c)
592
593     devP=invsqDr %*% (P-eP) %*% invsqDc
594
595
596
597
598
599     ###### HERE STARTS THE FOR LOOP
600
601     dfP=list()
602
603     sortOp=list()
604
605     bp=list()
606
607
608
609     colorPal=rainbow(K)
610
611
612
```

```
613  for(jj in 1:K){
614
615    #topfew=which(abs(devP[,jj]*sqrt(n))>1)
616
617    #print(labs[topfew])
618
619    dfP[[jj]]=data.frame(value=devP[,jj]*sqrt(n),place=1:nrow(devP),lbls=labs)
620
621    sortOp[[jj]]=sort(abs(dfP[[jj]]$value),decreasing=T,index.return=T)
622
623    #    sortOp2=sort(abs(dfP2$value),decreasing=T,index.return=T)
624
625    #    sortOp3=sort(abs(dfP3$value),decreasing=T,index.return=T)
626
627
628
629    dfP[[jj]]=dfP[[jj]][sortOp[[jj]]$ix,]
630
631    dfP[[jj]]$newplace=nrow(devP):1
632
633    xran=c(min(dfP[[jj]]$value)-.5,max(dfP[[jj]]$value)+.5)
634
635    if(firstfew>0){
636
637      dfP[[jj]]=dfP[[jj]][1:firstfew,] #names(dfP[[jj]])
638
639      dfP[[jj]]$newplace=firstfew:1
640
641    }
642
643
644
645    bbp=ggplot(data=dfP[[jj]], aes(x=value,y=newplace),labels=lbls)
646
647
648
649    if(fixmarg==T){
650
651      bbp=bbp+geom_segment(data=dfP[[jj]],aes(x=0,xend=value,y=newplace,yend=newplace),colour=colorPal[jj],size=se
652
653      bbp=bbp+theme(legend.position="none")+xlab("")+ylab("")+xlim(c(minx,maxx))
654
655      bbp=bbp+theme(axis.text.x = element_text(size=textSize), axis.text.y =
           element_text(size=textSize))
656
657      if(firstfew==0){bbp=bbp+theme(axis.line=element_blank(),axis.ticks =
           element_blank())}
658
659    }
660
661    else{
662
663      bbp=bbp+geom_segment(data=dfP[[jj]],aes(x=0,xend=value,y=newplace,yend=newplace),colour=colorPal[jj],size=se
664
665      bbp=bbp+theme(legend.position="none")+xlab("")+ylab("")+xlim(xran)
```

```
666
667         bbp=bbp+theme(axis.text.x = element_text( size =textSize) , axis . text . y   =
                element_text( size =textSize))
668
669         if ( firstfew ==0){bbp=bbp+theme(axis.line=element_blank(),axis.ticks =
                element_blank())}
670
671       }
672
673       if (nolabs==F){
674
675         bbp=bbp+geom_text(data=dfP[[jj]],aes(label=lbls) , size =textSize)
676
677       }
678
679       bp[[ jj ]]=bbp
680
681     }
682
683     #
684
685     #
686
687     out=list ()
688
689     out$G=bp
690
691
692
693     out
694
695   }
696   # 3) from   https://github.com/cran/clustrd/tree/master/R
697   clusval <-function(x,dst="full"){
698
699     if (dst=="full"){
700
701       if ( class (x)=="cluspca"){
702
703         data  =  scale(x$odata,  center  = x$center,  scale  = x$scale)
704
705         oDist  = daisy(data,metric="euclidean")
706
707       }else{
708
709         oDist=daisy(x$odata,metric="gower")
710
711       }
712
713     }else{
714
715       oDist=daisy(x$obscoord,metric="euclidean")
716
717     }
718
```

```
719
720
721     clu_res=cluster.stats(d=oDist,x$cluID,wgap=F,sepindex=F,sepwithnoise=F)
722
723
724
725     out=list()
726
727
728
729     out$ch=clu_res$ch
730
731     out$asw=clu_res$avg.silwidth
732
733     #out$crit=x$criterion
734
735     class(out) = "clusval"
736
737     return(out)
738
739 }
740
741 #### overall MCA
742
743 domout.data <- cbind( contex.out$clusmcaobj$obscoord, activity.out$clusmcaobj$obscoord,
        pain.out$clusmcaobj$obscoord, parti.out$clusmcaobj$obscoord,
        physi.out$clusmcaobj$obscoord,psycho.out$clusmcaobj$obscoord)
744
745 # svd visualization
746  library ( irlba )
747 domcomb.svd <- irlba(domout.data,3)
748 domcomb.svd.data <- domout.data %*% domcomb.svd$v
749
750 # r-tsne visualization
751 domcomb.rstne <- Rtsne(domout.data,3)
752
753 ## RKM # use functions : clustval(), clu_means()
754 outpca2 =princomp(domout.data)
755 plot(outpca2)
756
757 # outRKM3 <- tune_cluspca(as.matrix(domout.data), nclusrange = 8:12, ndimrange = 5:10,
        criterion = "asw", dst = "full", alpha = NULL, method = "RKM", center = TRUE, scale
        = TRUE, rotation = "none", nstart = 10, smartStart = NULL, seed = 1234)
758
759 # iterate 2:20 clusters using RKM to decide number of clusters
760 RKMcrit <- c()
761 RKMasw <- c()
762 RKMch <- c()
763 for (i in (2:20)) {
764     outRKM2 = cluspca(as.matrix(domout.data), i, 12, method = "RKM", rotation =
            "varimax",nstart=20)
765     RKMcrit[i]<- outRKM2$criterion
766     RKMasw[i] <- clusval(outRKM2)$asw
767     RKMch[i] <- clusval(outRKM2)$ch
768 }
```

```
769
770 # make a plot of the 3 measures
771 dat <- cbind(RKMcrit/10000,RKMasw*3, RKMch/300)
772 matplot(dat, type = c("b"),pch=1:3,col = 1:3, cex=0.5, xlab = "Clusters", ylab=
        "Measures", ann=FALSE, axes=FALSE)
773 legend("topright", legend =c("Winthin Variance/10000","Silhouette*3","CH Index/300"),
        col=1:3, pch=1:3, cex = 0.5)
774 axis(1, font.axis=1 , at= 1:20, cex.axis=0.75, family = "sans", col ="grey")
775 axis(2, font.axis=1, cex.axis=0.75, family = "sans", col="grey")
776 title(main="Choose Clusters with Different Measures",font.main = 1, cex.main= 1,
        family="sans") # create title
777 mtext("Clusters", 1, line=2, cex=1, family="sans") # create x label
778 mtext("Measure Values", 2, line=2, cex=1, family="sans") # create y label
779
780 # We choose 8 clusters as the final number of clusters
781
782 outRKM812 = cluspca(as.matrix(domout.data), 8, 12, method = "RKM", rotation =
        "varimax", nstart=20)
783 library(GGally)
784 cdsc812 = clu_means(outRKM812$odata,outRKM812$cluID, center=outRKM812$center)
785 cdsc812
786
787 ### meaning of each cluster
788 # see documents
789
790 # ids and labels
791 summary(as.factor(outRKM812$cluID))
792 # 1   2   3   4   5   6   7   8
793 # 212 173 148 114 114 66  64  37
794
795 ##
        ------------------------------------------------------------
796 # Clustering - Visualization
797 ##------------------------------------------------------------
798 # using R-tsne for 2d embedding
799 domcomb.rstne <- Rtsne(domout.data,3)
800
801 plot(domcomb.rstne$Y[,2:3], col=outRKM812$cluID,pch=outRKM812$cluID,cex=0.5, xlab =
        "dim2", ylab="dim3" )
802 legend("topright", legend =1:8, col=1:8, pch=1:8, cex = 0.5)
803
804 # 3d visualization
805 plot3d(domcomb.rstne$Y, col=outRKM812$cluID,pch=outRKM812$cluID,cex=0.5)
806
807 ##
        ------------------------------------------------------------
808 # Clustering - Evaluation
809 ##------------------------------------------------------------
810 #LBP intensity at 2weeks,3months,12months
811 #Verify the clustering results using first 10 columns of data
812
813 data.wlabel <- cbind(data[,2:10], outRKM812$cluID)
814 # Pain intensity changes
815 data.wlabel.va <- data.wlabel[,which(colnames(data.wlabel) %in% c("vasl2w","vasl3m",
        "vasl12m","outRKM812$cluID"))]
```

```
# md.pattern(data.wlabel)
data.wlabel.va.nona <- na.omit(data.wlabel.va)

vaslchange <- cbind (as.data.frame(comp.data %>% group_by(outRKM812$cluID)%>%
    dplyr::summarize(a=mean(Vasl0)))$a, as.data.frame(data.wlabel.va.nona %>%
    group_by(data.wlabel.va.nona[,4])%>% dplyr::summarize(a=mean(vasl2w)))$a,
    as.data.frame(data.wlabel.va.nona %>% group_by(data.wlabel.va.nona[,4])%>%
    dplyr::summarize(a=mean(vasl3m)))$a, as.data.frame(data.wlabel.va.nona %>%
    group_by(data.wlabel.va.nona[,4])%>% dplyr::summarize(a=mean(vasl12m)))$a)

matplot(t(vaslchange), type = c("b"),pch=1:8,col = 1:8, cex=0.5, ylab= "LBP Intensity",
    xlab ="Time Point", axes = FALSE)
axis(1, font.axis=1 , cex.axis=0.75, at = c(1,2,3,4),
    labels =c("original","2w","3m","12m"), family = "sans")
axis(2, font.axis=1, cex.axis=0.75, family = "sans")
legend("topright", legend =1:8, col=1:8, pch=1:8, cex = 1)

# RM prop change
data.wlabel.rm <- data.wlabel[,which(colnames(data.wlabel) %in%
    c("rmprop2w","rmprop3m", "rmprop12m","outRKM812$cluID"))]
data.wlabel.rm.nona <- na.omit(data.wlabel.rm)

rmchange <- cbind (as.data.frame(data.wlabel.rm.nona %>%
    group_by(data.wlabel.rm.nona[,4])%>% dplyr::summarize(a=mean(rmprop2w)))$a,
    as.data.frame(data.wlabel.rm.nona %>% group_by(data.wlabel.rm.nona[,4])%>%
    dplyr::summarize(a=mean(rmprop3m)))$a, as.data.frame(data.wlabel.rm.nona %>%
    group_by(data.wlabel.rm.nona[,4])%>% dplyr::summarize(a=mean(rmprop12m)))$a)

matplot(t(rmchange), type = c("b"),pch=1:8,col = 1:8, cex=0.5,ylab= "RMProp", xlab
    ="Time Point", axes = FALSE)
axis(1, font.axis=1 , cex.axis=0.75, at = c(1,2,3), labels =c("2w","3m","12m"), family =
    "sans")
axis(2, font.axis=1, cex.axis=0.75, family = "sans")
legend("topright", legend =1:8, col=1:8, pch=1:8, cex = 1)

# gen
data.wlabel.gen <- data.wlabel[,which(colnames(data.wlabel) %in% c("gen2w","gen3m",
    "gen12m","outRKM812$cluID"))]
data.wlabel.gen.nona <- na.omit(data.wlabel.gen)

genchange <- cbind (as.data.frame(data.wlabel.gen.nona %>%
    group_by(data.wlabel.gen.nona[,4])%>% dplyr::summarize(a=mean(gen2w)))$a,
    as.data.frame(data.wlabel.gen.nona %>% group_by(data.wlabel.gen.nona[,4])%>%
    dplyr::summarize(a=mean(gen3m)))$a, as.data.frame(data.wlabel.gen.nona %>%
    group_by(data.wlabel.gen.nona[,4])%>% dplyr::summarize(a=mean(gen12m)))$a)

matplot(t(genchange), type = c("b"),pch=1:8,col = 1:8, cex=0.5, axes=FALSE)
axis(1, font.axis=1 , cex.axis=0.75, at = c(1,2,3), labels =c("2w","3m","12m"), family =
    "sans")
axis(2, font.axis=1, cex.axis=0.75, family = "sans")
legend("topright", legend =1:8, col=1:8, pch=1:8, cex = 0.8)
```