CLUSTER ANALYSIS CHALLENGE: LOWER BACK PAIN DATASET

SAN JOSE STATE UNIVERSITY
BY HONGZHE LIU, LE PHAN
ADVISOR DR. CRISTINA TORTORA

May 30, 2017

Contact:
lephan1205@gmail.com
hongzheliu11@gmail.com

Contents

1 Background

On April 18, 2017, the International Federation of Classification Societies (IFCS) issued a challenge to its members and the classification community to analyze a dataset of 928 low back pain patients. In this paper, we present our contribution of cluster analysis for this dataset as a response to the IFCS challenge. We will discuss our data cleaning process, which we view as a two-pronged approach: inferring values missing not at random and imputing values missing at random. We will also discuss the challenges in clustering mixed data types and the required data transformation prior to applying a clustering algorithm. We call this data transformation process "split-then-join". Finally, we offer our interpretation of the clustering results with respect to validation variables and final thoughts on selecting important variables to classify new observations.

2 Data Cleaning Process

2.1 Data Inference

The original dataset contains patient self-reported questionnaires and clinicians recorded examinations resulting in 112 baseline variables and three outcome variables for different time periods (2 weeks, 3 months and 12 months) following the initial clinical consultation. These variables covered various domains from pain history, activity limitation, work-related questions, validated questionnaires, fear avoidance, etc. There are many missing values in the data as some patients and clinicians did not fill out all the questions.

Our first task is to determine the nature of the missing data. If data is missing at random, then the missing values can be imputed using our chosen imputation method—to be discussed in the next section. However, some values are not missing at random and can be deduced based on their relationship with other variables. For instance, many questions inquire whether activities at work impact the patient's pain level or whether their pain limits their activity at work. Obviously, patients who do not have a job (i.e., students, unemployed and pensioners) will not fill out these questions as they are not applicable. Using the patient's employment status, we can infer that some of the missing values are not missing at random (i.e., they can be substituted with a new value indicating the patient did not complete the questionnaires for legitimate reason). Table 1 below demonstrates this idea. Our goal is to fill in as many missing values as possible by inference prior to imputing values missing at random.

Table1: Treatment of Values Missing Not at Random

Variables with missing values	Reason for missing not at random	Inference
fabq60, fabq70, fabq80, fabq90, fab100, fabq110, fab120, fab130, fabq140	Questions involving pain level with respect to work condition; only to be answered if patient is working	Replace NAs with new category -1 if patient's employment situation, barb0, indicates not working
facetextrot, facetsit, facetwalk, paraspin_debut	Questions only to be asked if patients answer yes to having dominating back pain	Replace NAs with new category - 1 if patient does not have dominating back pain (domin_bp = 1 for no)
musclegroup_palp	Question involving pain caused by different muscle groups.	Replace NAs with new category - 1 if patient have no pain referred from triggerpoint(triggerpoint = 0 for no) and no replication of pain during muscle palpation(musclepalp = 0)
musclepalp	Highly correlated with musclegroup_palp	Use musclegroup_palp to update NAs in musclepalp

After filling in missing values based on known relationships between variables, we check the dataset for variables that still contain greater than 30 percent missing values. This enables us to zero in on variables with many missing values that can be filled in by inference, yet whose linkages to other variables may not be obvious. After the second round of inference, many missing values were filled with the appropriate deduced values. We also removed 13 observations with greater than 30 percent missing values. At this point, we believe the remaining missingness are missing at random and thus can be imputed.

2.2 Data Imputation

Our next task is to determine the appropriate techniques to impute the remaining missing values. Since the dataset consists of mixed data types (i.e., binary, continuous and categorical), the selected techniques must be robust for each type. Binary data can be forecasted with logistic regression while continuous data can be forecasted with linear regression. Categorical data can be forecasted with multinomial method. For this reason, we favor the Multiple Imputation by Chained Equations (MICE) method[1]. In R, MICE is implemented with a function by similar name, mice(), which handles both data missing at random (MAR) and missing not at random (MNAR). This gives us an extra layer of comfort in the event that the inference process did not completely remove MNAR missingness. MICE algorithm can be generalized by the following steps:

[1] "mice: Multivariate Imputation by Chained Equations in R | van Buuren ...." 12 Dec. 2011, https://www.jstatsoft.org/article/view/v045i03.

1. Initial imputation are created using random draw from the data as "placeholders".
2. The "placeholders" for the first variable with missing values are set back to missing, this variable is the response in the regression model while the remaining variables are predictors in step 3.
3. The appropriate model (i.e., logistic regression, linear regression, multinomial, etc.) is used to predict the missing values in the response.
4. The missing values in the response is replaced with the predicted values.
5. Repeat Step 2-4 for the next variable with missing values.

The cycling through each variable is considered one iteration. At the end of each iteration, the missing values are all replaced by predicted values. MICE converges when the variance between sequences is smaller than the variance with each individual sequence. Stef van Buuren and Karin Groothuis-Oudshoorn suggest 10-20 cycles. In addition to specifying the number of iteration, users can also set the number of imputation, m, for each missing values resulting in m-datasets. We impute our dataset using ten iterations, each missing values imputed five times, which produce five imputed datasets. We then consolidate the five imputed datasets into one using the mean for numerical variables and mode for categorical variables.

3 Clustering Methods

3.1 Data Transformation

The nature of mixed data types presents some challenges with respect to clustering methods. Popular algorithms such as k-means, fuzzy k-means, probabilistic distance clustering and mixture models work well with numeric data but not categorical data. For categorical data, method such as multiple correspondence analysis (MCA) is better suited for studying the links between categories. Thus, data transformation is needed to obtain the final dataset with consistent data type, while preserving the relationships between the variables, before applying a clustering algorithm. To transform the data, we split our dataset into two subsets, one purely categorical and the other purely numeric. Ordinal data are treated as numeric. We then apply MCA to the categorical subset and examined their principal coordinates. With MCA, we are able to reduce the dimension of the categorical subset from 73 to just seven. Since the principal coordinates are numeric linear combinations of categorical data, we append these seven numeric columns to the purely numeric subset, resulting in one numeric dataset. We believe this split-then-join approach allows us to preserve the structure of the original raw data. Figure 1 below illustrates this idea. At this stage, our data is completely cleaned and ready to be clustered. We will refer to this as the **transformed dataset** going forward.

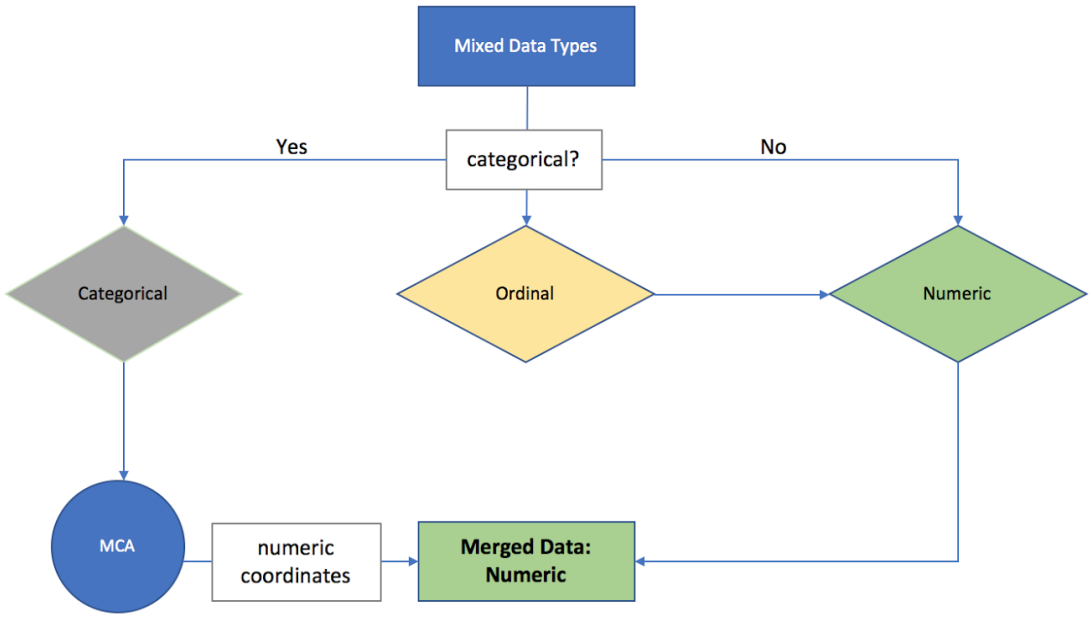


Figure 1: Data Transformation Method

3.2 Determine the Number of Clusters

In order to apply a clustering algorithm, we need to determine the number of clusters in our data. Where appropriate, we use cluster comparison metrics such as Calinski-Harabasz[2] criterion and Bayesian Information criterion[3] (BIC) to analyze the preliminary clustering results for the following clustering algorithms while varying the number of clusters:

1. K-means
2. Partition Around Medoids (PAM)
3. Fuzzy K-means (FKM)
4. Probabilistic Distance Clustering (PDClust)
5. Gaussian Parsimonious Clustering Models
6. Mixture Generalized Hyperbolic Distributions (MGHD)

Clustering solutions with higher Calinski-Harabasz criterion is preferred over those with lower Calinski-Harabasz criterion. In contrast, solutions with smaller BIC is preferred over large BIC. The initial run suggests three to four clusters exist within the data. We then use this known values to rerun these algorithms and evaluate their performance.

[2] Calinski, T., and J. Harabasz. "Communications in Statistics A dendrite method for cluster analysis." 27 Jun. 1974, https://www.researchgate.net/profile/Tadeusz_Calinski/publication/233096619_A_Dendrite_Method_for_Cluster_Analysis/links/555213e108aeaaff3befe29b/A-Dendrite-Method-for-Cluster-Analysis.pdf.

[3] G. Schwarz. "Estimating the Dimension of a Model - ResearchGate." https://www.researchgate.net/publication/38358303_Estimating_the_Dimension_of_a_Model.

3.3 Clustering Algorithm Selection

With the clustering results, we compare their Adjusted Rand Index[4] (ARI). ARI measures the similarity of two data clustering and ranges from zero to one. ARI of zero indicates no match while ARI equals one indicate perfect match between two clustering results. The comparisons in table 2 below enable us to narrow the methods down to three for further analysis: k-means, MGHD, and multivariate skew-t distribution.

Table 2: Algorithm Comparisons Using ARI

Algorithm Comparisons	Adjusted Rand Index (ARI)
ARI(kmeans.out$cluster, pam.out$clustering)	0.80602
ARI(kmeans.out$cluster, fkm.out$clus[,1])	0.87352
ARI(kmeans.out$cluster, pdc.out$label)	0.52930
ARI(kmeans.out$cluster, mvn.out$clust)	0.03592
ARI(pam.out$clustering, fkm.out$clus[,1])	0.70774
ARI(pam.out$clustering, pdc.out$label)	0.58555
ARI(fkm.out$clus[,1], pdc.out$label)	0.54035
ARI(mvn.out$clust, mst.out$clust)	0.71292
ARI(mvn.out$clust, GHD.out@map)	0.51473
ARI(mst.out$clust, GHD.out@map)	0.68374

We then review the effect of these three clustering solutions on the validation variables and found that k-means does the best job at describing these variables. Additionally, the projection of the transformed data onto the first two dimensions of principal component analysis (PCA) reveals that the three clusters are well separated (figure 2).

[4] W. Rand. "Objective Criteria for the Evaluation of Clustering Methods - jstor." https://www.jstor.org/stable/2284239.

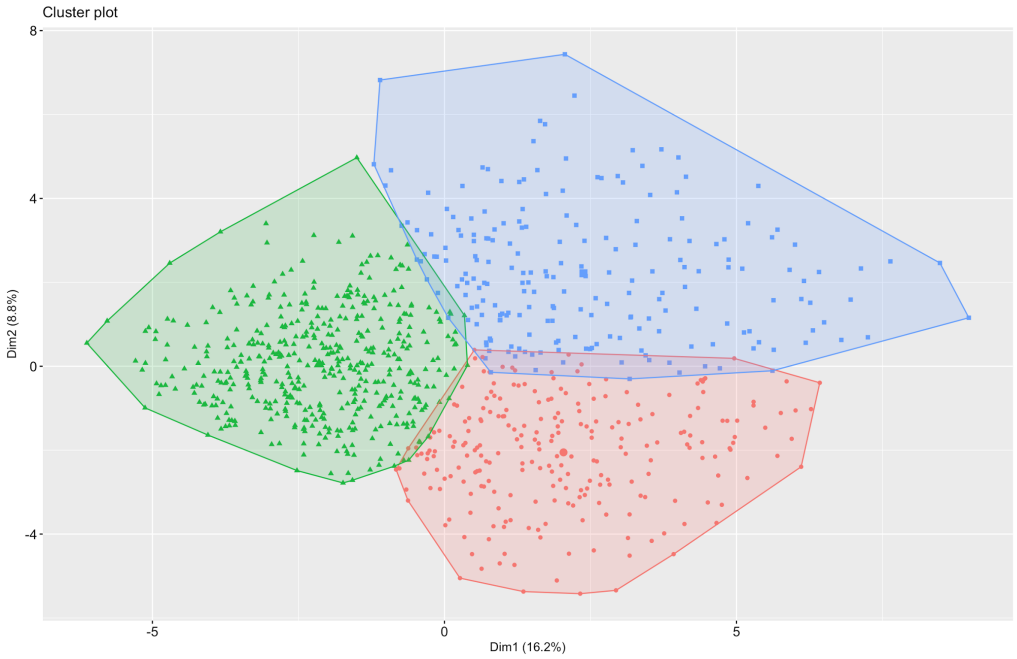
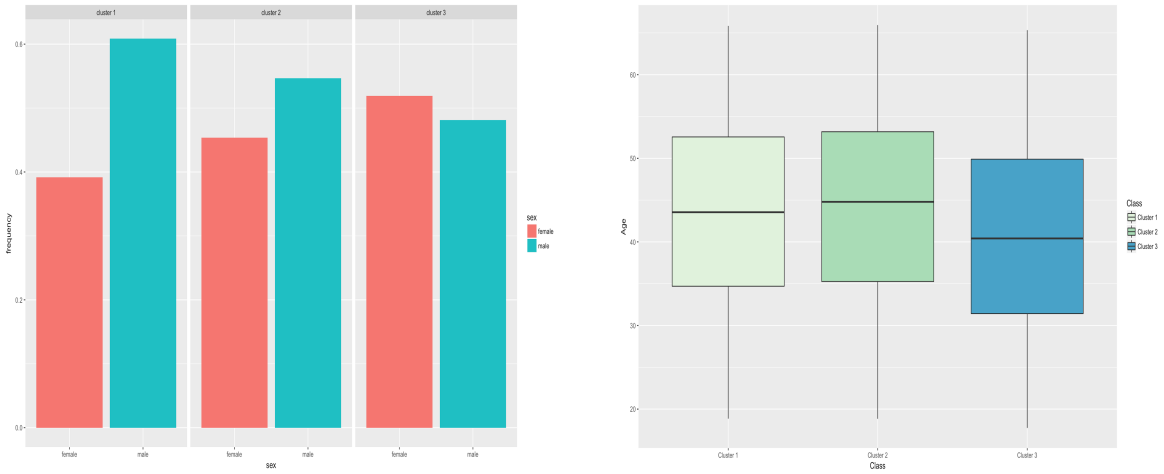


Figure 2: Data Projection onto Dimension 1 and 2 for 112 Variables

4 Interpreting Clustering Results

4.1 Cluster Characteristics

K-means clustering result reveals some interesting cluster characteristics. From figure 3 below, we can see that cluster one and two have more male than female patients while cluster three's male-to-female ratio is nearly even. Patients in cluster three, on average, are slightly younger than patients in cluster one and two while the height and body mass index (BMI) of all three clusters are similar. This indicates that height and BMI cannot be used to differentiate cluster membership.



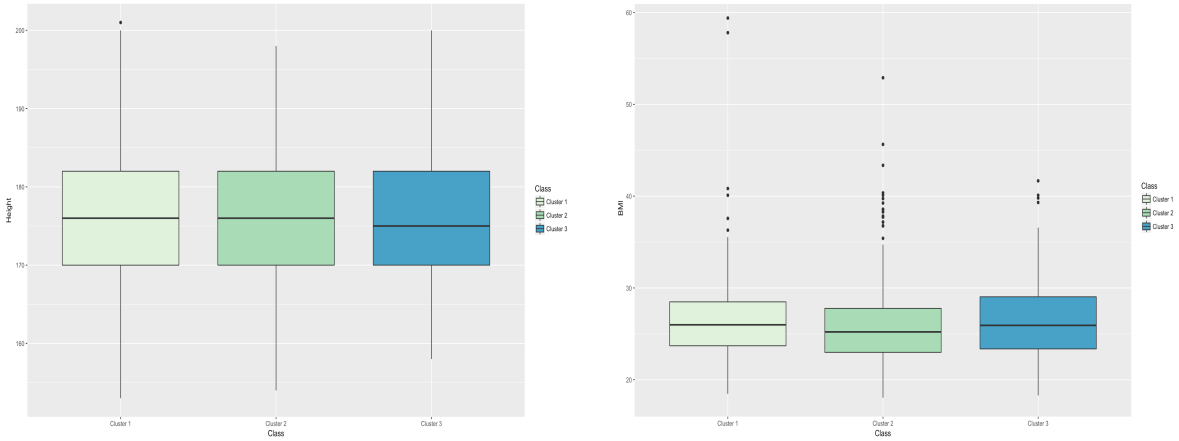


Figure 3: Cluster Characteristics (Sex, Age, Height, BMI)

Another interesting finding is that there may be a relationship between full-time employment status and the duration of lower back pain at the time of consultation. Figure 4 and 5 below show there are significantly more patients with full-time work status and short-term lower back pain (1-2 weeks) in all three clusters.

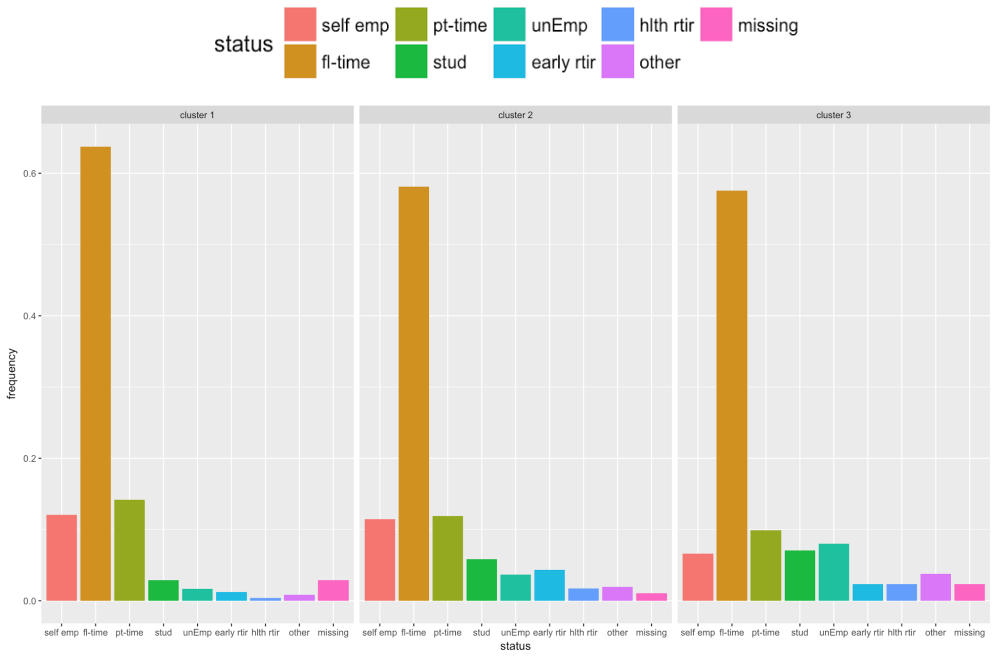


Figure 4: Working Status

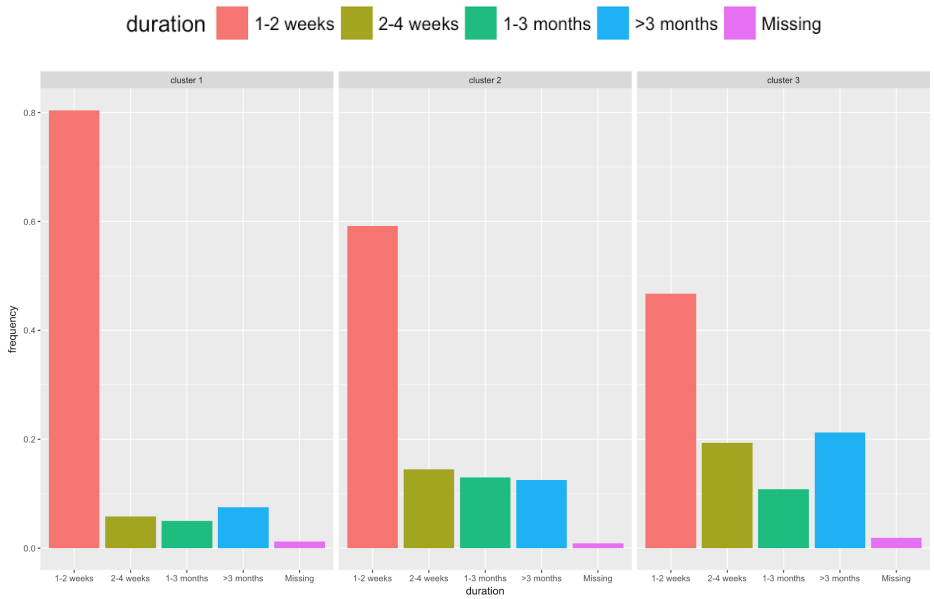


Figure 5: Pain Duration

4.2 Validation Variables

K-means clustering algorithm separates the clusters quite well as evident by distinguishable patterns for all three clusters with respect to the validation variables. From figure 6 below, it is obvious that patients in all three clusters experienced a decline in both LBP intensity and Roland Morris scores—twelve months after the initial consultation. Patients in cluster 3 experienced the highest LBP intensity and Roland Morris scores but they have the greatest perceived improvements compared to the other two clusters. Interestingly, cluster 3 has highest percentage of members considered at "high risk" for having back pain disability, as shown by figure 7.

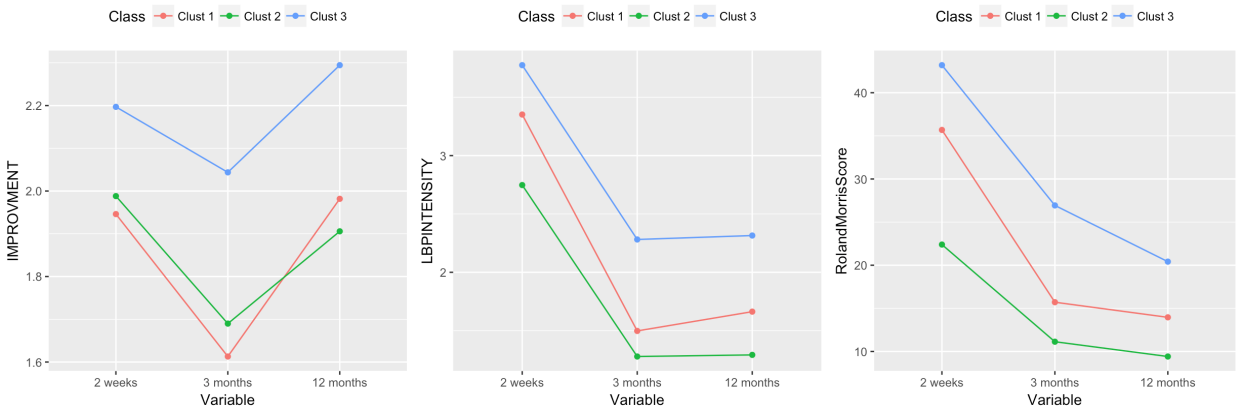


Figure 6: Validation Variables

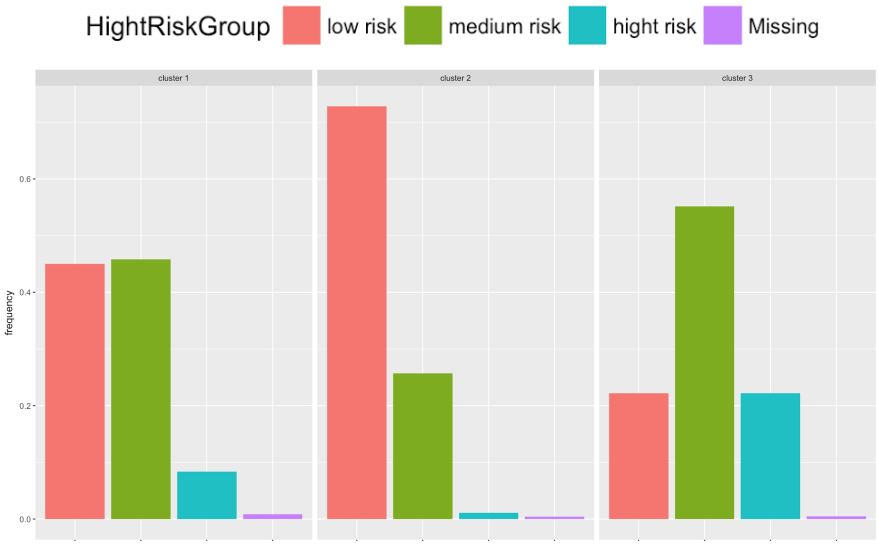


Figure 7: High Risk Group

We also notice an interesting pattern in the validation variables. While all three clusters exhibit a decline in LBP intensity and Roland Morris score from the period of 2-week to 3-month, their

perceived improvement actually declined in this period. One would think that as physical condition improves (i.e., declining LBP intensity and Roland Morriss score), patients should experience an increase rather than decline in perceived improvement. We suspect there is a "time lag" between when patients feel better versus when their conditions improved. This is evident by the flattening out of LBP intensity while the perceived improvements rises from 3-month to 12 month period.

5 Final Thoughts

5.1 Variables Selection

Up to this point, we have been performing all analyses based on the full set of 112 variables. Plotting the first seven principal components shows that the clusters are well separated using the first two principal components (figure 8). Consequently, we believe it is possible to describe the full dataset using fewer variables.

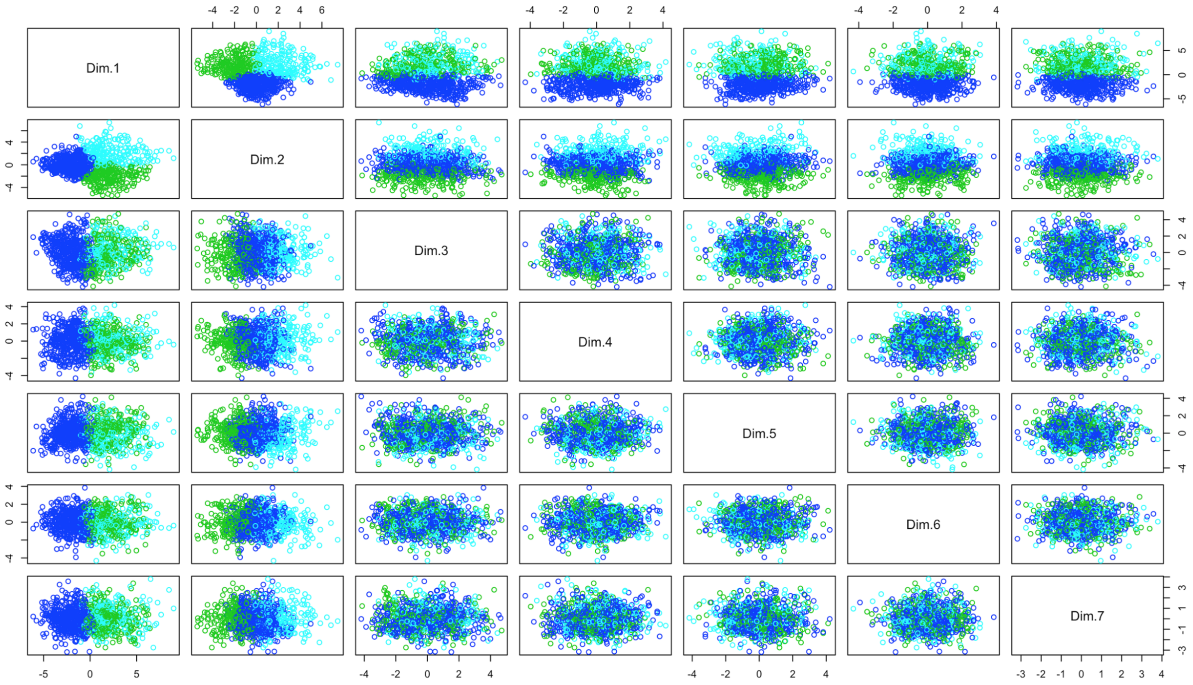


Figure 8: Pair-plots of First 7 Principal Components Using Transformed dataset

To identify the important variables in the first two principal components, we analyze their contributions to each dimension as well as examining their biplot. We found that the data can be reduced to 27 numeric and 28 categorical variables (see Appendix). We will refer to this smaller set of selected variables as the **reduced dataset**. Table 3 and figure 9 below summarize these important variables.

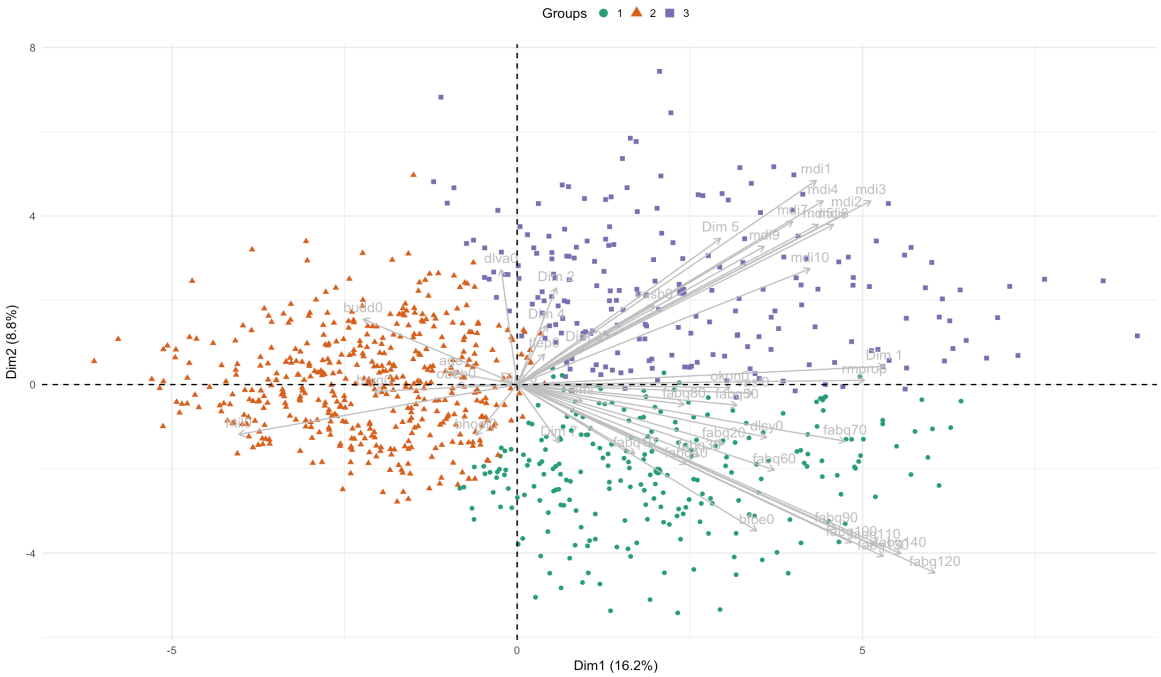


Figure 9: Important Variables in Each Cluster

Table 3: Variable Contribution Analysis

Variable Contribution Analysis - Part 1				
Ranking	**Variables**	**PC1**	**PC2**	**Contribution Score**
1	fabq120	0.065	0.066	0.733
2	fabq140	0.055	0.053	0.609
3	mdi3	0.047	0.063	0.586
4	fabq130	0.050	0.055	0.580
5	mdi1	0.033	0.077	0.548
6	fabq110	0.048	0.048	0.537
7	mdi2	0.041	0.055	0.510
8	mdi4	0.035	0.063	0.501
9	fabq100	0.042	0.046	0.486
10	mdi8	0.037	0.047	0.459
11	mdi5	0.034	0.048	0.434
12	fabq90	0.038	0.039	0.429
13	mdi7	0.028	0.049	0.401
14	MCA 1	0.050	0.001	0.369
15	mdi10	0.032	0.025	0.331
16	rmprop	0.045	0.000	0.328
17	fabq70	0.040	0.006	0.316
18	bfbe0	0.021	0.040	0.312
19	mdi9	0.023	0.035	0.306
20	MCA 5	0.015	0.040	0.269
21	fabq60	0.025	0.014	0.233
22	htil0	0.029	0.005	0.229
23	dlsy0	0.023	0.005	0.190

Variable Contribution Analysis - Part 2				
Ranking	**Variables**	**PC1**	**PC2**	**Contribution Score**
24	vasl0	0.021	0.000	0.152
25	fabq20	0.016	0.007	0.142
26	fabq50	0.018	0.001	0.134
27	fabq30	0.012	0.010	0.128
28	fabq40	0.011	0.012	0.124
29	okon0	0.017	0.000	0.123
30	dlva0	0.000	0.025	0.097
31	vasb0	0.007	0.012	0.097
32	budd0	0.009	0.008	0.096
33	fabq80	0.010	0.001	0.079
34	fabq10	0.005	0.009	0.073
35	MCA 2	0.001	0.017	0.072
36	bryg0	0.008	0.000	0.056
37	MCA 7	0.001	0.006	0.029
38	MCA 4	0.000	0.007	0.028
39	bhoej0	0.001	0.005	0.023
40	MCA 3	0.002	0.003	0.022
41	bmi	0.002	0.001	0.014
42	age	0.002	0.000	0.013
43	obeh0	0.001	0.000	0.010
44	tlep0	0.000	0.002	0.009
45	MCA 6	0.000	0.000	0.000

- Variable X_k's contribution to principal component Y_i is defined as: $\frac{P^2_{Y_i,X_k}}{\sum_{i=1}^{p} P^2_{Y_i,X_k}}$ where $P^2_{Y_i,X_k}$ is the square correlation between variable X_k and principal component Y_i and $\sum_{i=1}^{p} P^2_{Y_i,X_k}$ is the total sum square of the correlation coefficients of the components Y_i. Larger variable contribution has greater influence on a component than smaller variable contribution.

- Contribution score for each variable is defined as: $\sum_{i=1}^{p} C_{ki}\lambda_i$ where λ_i is the i^{th}-eigenvalue and C_{ki} is the k^{th}-variable contribution to the i^{th} component.

To validate this, we rerun the k-means algorithm using only the selected variables and compare its clustering result against the result from the full set of 112 variables using the adjusted rand index. This comparison produces a high ARI of 0.92, which indicates that the two clustering results are very similar. The projection of this reduced data onto the first two principal components (figure 10) confirms that the selected variables performs just as well as the larger set with respect to separating the clusters and therefore sufficient enough to describe the original dataset.

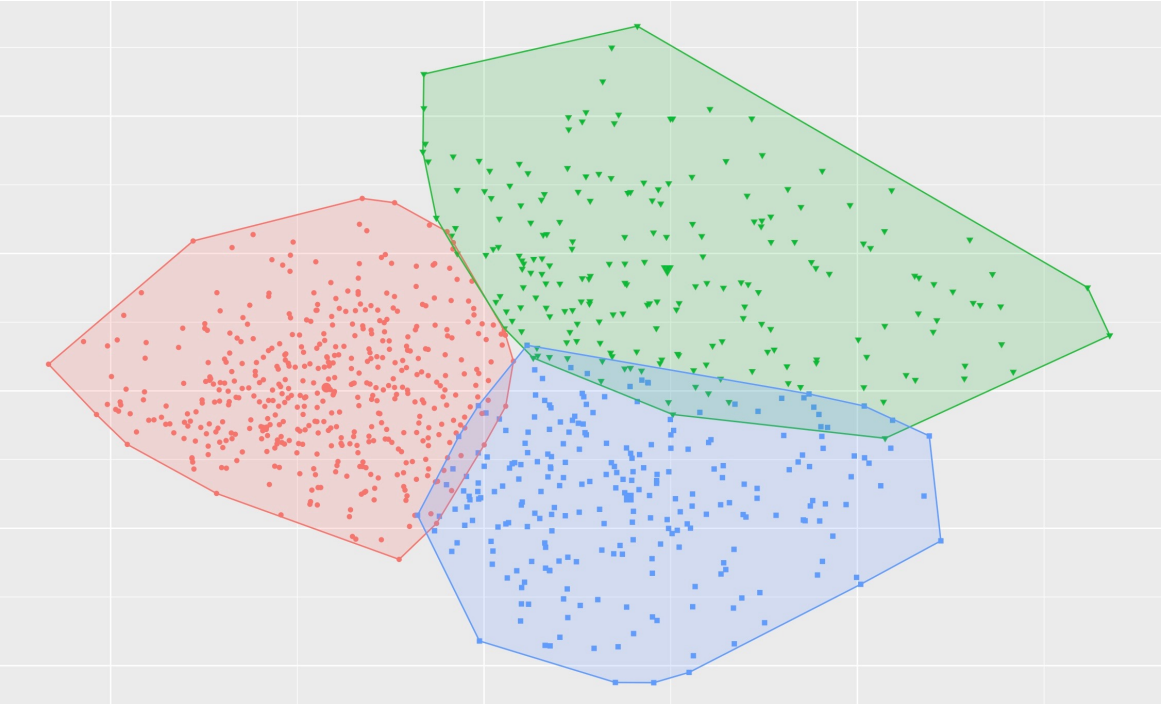


Figure 10: Projection of Reduced Data onto the First Two Principal Components

5.2 Classification

Figure 11 below shows how the three clusters differ with respect to the numeric variables from the transformed dataset. Note these variables have been scaled to eliminate the effect of different ranges in their values. Cluster 1 is described by high score on Fear-Avoidance Beliefs Questionnaire (*"fabq's"*) and Roland Morris Summary (*"rmprop"*). Similarly, cluster 3 is described by high score on self-reported mood questionnaires—Major Depression Inventory (*"mdi's"*). High *mdi* score indicates poor mood or mental health. In contrast, patients in cluster 2 has the lowest score in these questionnaires. In words, cluster 1 is characterized by patients who experienced above average back pain and limited functional activities. Patients in cluster 3 experience higher level of depression characterized by having low spirit, sadness, loss of appetite and inability to sleep at night and are also considered to be a higher risk group (recall Figure 7). In contrast, cluster 2 patient's back pain is not affected by physical activities and they generally have better mental health compared to cluster 1 and 3. Keen observers may wonder why we

chose to discuss the differences in these clusters based on only numeric variables and excluded categorical variables. We do so because many important variables related to Roland Morris questionnaires are summarized by a single numeric summary score, *"rmprop"*, and it is easier to represent cluster differences using this numeric variable. Additionally, we choose to examine important categorical variables (*"start's"*)—those that stratified patients according to their risk of having back pain disability—separately using their representative variable *start_risk*.

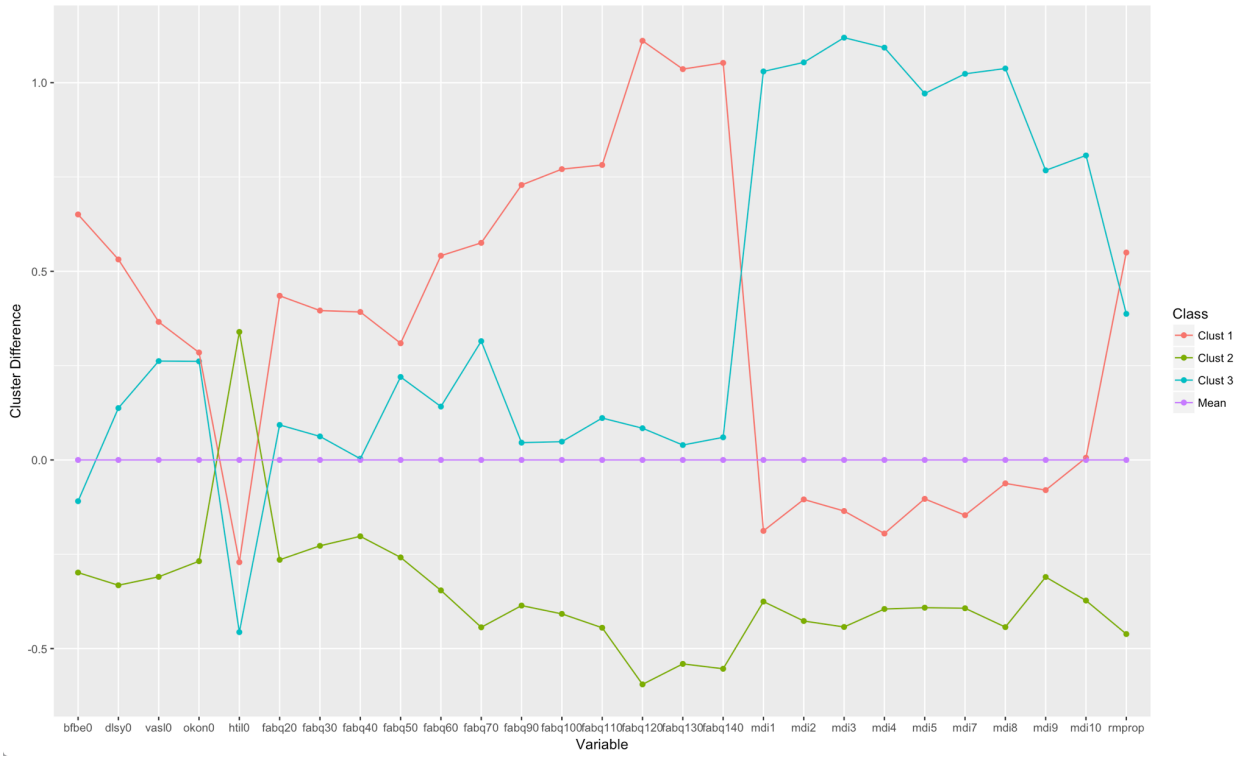


Figure 11: Cluster Difference Using Scaled Variables

Consequently, we believe these characteristics can be used to classify new patient cluster membership. Table 4 below provides a general guideline to classify new patients.

Table 4: New Observation Classification Guide

Fear-Avoidance Beliefs Questionaires (fabq's)	Major Depression Inventory (mdi's)	Roland Morris Summary Score (rmprop)	Back Pain Disability Risk Profile (start_risk)	Cluster Membership
Above Average		Above Average	Low - Medium	1
Below Average	Below Average	Below Average	Low - Medium	2
Above Average	Above Average		Medium - High	3

Appendix

Table 5: List important variables

List of Important Variables

Continuous & Ordinal Variables			Categorical Variables		
fabq120	mdi8	fabq60	start_risk	rm50	rm90
fabq140	mdi5	htil0	rm30	start90	rm110
mdi3	fabq90	dlsy0	rm100	rm120	start80
fabq130	mdi7	vasl0	rm180	start30	rm80
mdi1	mdi10	fabq20	rm220	rm160	rm150
fabq110	rmprop	fabq50	rm140	rm200	romflex
mdi2	fabq70	fabq30	rm70	start40	start60
mdi4	bfbe0	fabq40	rm60	rm40	rm170
fabq100	mdi9	okon0	rm130	rm10	start50
				romrotl	