Hunter College Data: Clustering Analysis

Krista DeStasio

11/06/2018

Table of Contents

# Background information

*Tracy writes:* “As I see it our first goal is testing whether distinct attention bias grouping emerge from the dot probe and questionnaire data using classification algorithms. Once these groups are identified, we can look at whether these are treatment-relevant subgroups (i.e., ABMT training is more effective for one group versus the other).”

“examine the conditions under which ABMT efficacy is boosted or disrupted, and to identify individual differences that impact training gains”

A search for moderators that machine learning may be sensitive enough to detect.

## Introduction

The primary aim of this analysis is to identify whether meaningful subgroups emerge from the anxiety dot probe. If meaningful subgroups do emerge, our secondary aim is to whether cluster membership relates to anxiety as measured by the State-Trait Anxiety Inventory (STAI). To test this aim, group membership will be entered as an independent variable into a between-subjects ANOVA with STAI trait anxiety scores as the dependent measure of interest. This is done with the eventual goal of predicting treatment trajectories in an unrelated sample of ABMT recipients.

## Methods

All analyses will be conducted using R (R Core Team, 2018). I will run both a spectral clustering analysis (connectivity) and a k-means cluster analysis (compactness) to identify subgroups within the Anxiety Dot Probe measures as these approaches are well suited to numeric data. Hierarchical clustering methods, which do not require a priori specification of number of clusters, will also be applied and evaluated. Prior to any analysis, the data will be split into development (80%) and holdout (20%) sets, then the development set will be further split into training (70%) and testing (30%) sets.

## The data

Dot Probe measures that will be included in the analysis are:

* rt\_neutral\_nt: Average RT on all neutral trials
* rt\_threat\_nt: Average RT on all threat trials
* rt\_baseline: Average RT on baseline trials where two neutral faces appear and there is no competition for attention between threat and neutral faces; These trials appear randomly throughout the task
* rt\_outliers: # of trials considered to be outliers based on the following criteria (RTs faster than 150 ms or slower than 2000 ms; any trial RT that was +/- 3 SD from the person’s mean RT for that trial type)
* mean\_pos: mean of positive trial-level threat bias scores
* mean\_neg: mean of negative trial-level threat bias scores
* peak\_pos: highest positive trial-level threat bias score
* peak\_neg: highest negative trial-level threat bias score
* variability: absolute value of the distance across all trial-level threat bias scores / number of pairs

## Data preprocessing

### Outliers and missingness

Outliers were already removed from the data by our collaborators based on the percentage of trials that were answered correctly. Participants with an accuracy of .80 or greater are included in the dataset. There should be no initial missing data for the dot probe measures as the included metrics can be computed for anyone who completed the task.

### Colinearity and transformation

The training, testing, and holdout sets will be independently preprocessed. The datasets will be checked for variables that are linear combinations of other variables and those indicated will be removed. Variables will be standardized using the scale() function to mean = 0, and standard deviation = 1.

# Planned analyses

## K-Means Clustering Analysis

### Development

The number of clusters to begin with will be decided by comparing the results of several different methods:

* Determine the optimal number of clusters to try with k-means by comparing the results of several different methods:
  + The sum of squared error (SSE) scree plot
  + The NbClust function with method kmeans
    - using the nstart option as recommended [here](https://www.r-statistics.com/2013/08/k-means-clustering-from-r-in-action/), which attempts multiple initial configurations and reports on the best one.
  + [factoextra::fviz\_nbclust()](https://www.datanovia.com/en/blog/types-of-clustering-methods-overview-and-quick-start-r-code/)
  + Cross-validation with 10 folds
* A number of different clustering solutions will be tried by using cross-validation
* An importnat note about k-means is that [“it can only detect compact, hyperspherical clusters that are well separated.”](https://www.sciencedirect.com/science/article/pii/S0957417412008767)

## Hierarchical clustering

* hclust() function

# Decision points to address in the pre-registration

**Decisions in bold**

1. Decide whether the outcome measure (anxiety) be STAI trait anxiety, or the STAI total score?
   * **Trait anxiety**
2. Decide if the outcome measure should be continuous, or binary yes/no based on some cut-off (e.g. clinical)? Or should both be tried?
   * **Will do continuous linear analysis first, then a secondary analysis with clinical cut-off of anxiety scores**
   * Need to define clinical cut-off. **Ask Tracy what they use**
     + STAI population mean of ~40 (go 1 standard deviation above population mean of 40)
3. General notes
   * [“Clustering is more of a tool to help you explore a dataset, and should not always be used as an automatic method to classify data. Hence, you may not always deploy a clustering algorithm for real-world production scenario. They are often too unreliable, and a single clustering alone will not be able to give you all the information you can extract from a dataset.”](https://www.datacamp.com/community/tutorials/k-means-clustering-r)
   * [Methods for choosing the number of clusters](https://en.wikipedia.org/wiki/Determining_the_number_of_clusters_in_a_data_set)
   * Hierarchical clustering does not require knowing the number of clusters beforehand. Could be a good option.  
     -[“a hierarchical cluster analysis was performed in order to determine the number of clusters for the K-means approach.”](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4005932/)

# Code

## Data cleaning

### Read in the data.

#data\_matrix <-

* Use janitor to make the dataset names a consistent format.
* Subset the data to retain only the dot probe measures (described above) and STAI Trait total score: stai\_trait\_totalscore
* Subset the data into the set that will be used for the cluster analysis (all variables except STAI score)
* There are no missing data as the dot probe measures exist for all participants who completed the task and who are included in this analysis. I therefore need not worry about biasing the results by omitting participants based on attention measures.

## Data partitioning

Split data into development and holdout, then training and testing

* 80/20 development/holdout split of the full data set.
* 70/30 train/test split of the initial development set.

## Pre-processing

Preprocess the train/test sets

* Check for and remove variables that are linear combinations of other variables.
* Standardize variables since they vary in range (if using more than just RT).
  + Using the scale() function, variables are scaled to mean = 0, sd = 1.

## Clustering algorithms

### K-means cluster analysis

#### Development

Run k-means with 1 through 6 clusters. Record the total within-cluster sum of squares for each iteration with a different cluster number. Plot as scatterplots and as a scree plot. Look for the elbow in the scree plot to determine the number of clusters to use.

set.seed(50)  
  
# Initialize the total within sum of squares error  
wss <- 0  
  
for (i in 1:6) {  
 kmeans\_out <- kmeans(data\_matrix, centers = i, nstart = 20, iter.max = 50)   
 wss[i] <- km\_out$tot.withinss  
   
 plot(x, col = kmeans\_out$cluster, main = round(km\_out$tot.withinss, 2), xlab = "", ylab = "")  
}  
  
# Scree plot  
plot(1:15, wss, type = "b",   
 xlab = "Number of Clusters",   
 ylab = "Within groups sum of squares")

Select the final model

# Number of clusters selected based on scree plot  
#k <-   
kmeans\_out <- kmeans(data\_matrix, centers = k, nstart = 20)   
  
# Plot the final model  
plot(data\_matrix[, c("var1", "var2")],  
 col = km\_out$cluster,  
 main = paste("k-means clustering of anxiety dot probe with", k, "clusters"),  
 xlab = "var1", ylab = "var2")

Identify the number of clusters

#### Apply the solution to the test data set

* Repeat the above train/test steps until a satisfactory solution is reached.

#### Finalize the k-means model

Pre-processing of the holdout data set

* Check for and remove variables that are linear combinations of other variables.
* Standardize variables since they vary in range so they have mean = 0, sd = 1.

#### Apply the k-means model

Apply the finalized cluster analysis to the holdout data set

* Evaluate cluster solution performance on the independent holdout data.
  + [“Since the centroids provided by the function are based on standardized data, the aggregate() function is used along with the cluster memberships to determine variable means for each cluster in the original metric.”](https://www.r-statistics.com/2013/08/k-means-clustering-from-r-in-action/)

# Analysis Plan, Part 2: Between-subject analysis of variance (ANOVA)

Do this, or not?

* Use the data frame of dot probe measures and STAI Trait total score: stai\_trait\_totalscore
* Remove rows with missing values on the stai\_trait\_totalscore using na.omit()

aov(STAI\_anxiety ~ cluster\_groups + Error(subject/cluster\_groups), data=full\_dataset)

# Resources

* [Datacamp tutorial:](https://www.datacamp.com/community/tutorials/k-means-clustering-r) Describes basics of clustering & provides a tutorial of k-means clustering, including interpretation of output in R.
* [A discussion of the relationship betweek K-means and PCA](https://stats.stackexchange.com/questions/183236/what-is-the-relation-between-k-means-clustering-and-pca)
* [An overview of K means and hierarchical clustering, along with a good description of distance metrics](https://www.hackerearth.com/practice/machine-learning/machine-learning-algorithms/clustering-algorithms-evaluation-r/tutorial/)
* [Von Luxburg, U. (2007). A tutorial on spectral clustering. Statistics and Computing, 17(4), 395–416. http://doi.org/10.1007/s11222-007-9033-z](https://arxiv.org/pdf/0711.0189.pdf): Preprint of a tutorial on spectral clustering by the Max Planck Institute
* [Luxburg, U. V., Bousquet, O., & Belkin, M. (2005). Limits of spectral clustering. In Advances in neural information processing systems (pp. 857-864).](http://papers.nips.cc/paper/2692-limits-of-spectral-clustering.pdf): Preference for normalized over nonnormalized spectral clustering
* Unsupervised learning: [Behavioral segmentation](https://www.datascience.com/blog/k-means-clustering) by attention bias measures (Dot Probe)\*\*
* [Spectral clustering R implementation](https://rpubs.com/nurakawa/spectral-clustering)
* [Drawbacks of K mean clustering](https://stats.stackexchange.com/questions/133656/how-to-understand-the-drawbacks-of-k-means)
* [Alternatives for segmenting noisy data](https://www.datascience.com/blog/k-means-alternatives)