**Replication and Extension Project: Map of Subjective Feelings**

Joey Heffner, Jae-Young Son, & James Willmott

**REPLICATION BACKGROUND**

Emotions play a large role in public health policy as understanding how emotions develop, transition, and are represented are important to improving psychological and physical wellbeing. Daily experience causing large changes in emotions and subjective feelings and humans face the difficult challenge of organizing these experiences. These inner sensations are vital cues to our physical well-being (e.g., sharp pains in the stomach are a reason to seek medical care) and mental well-being (e.g., feelings of pride and success motivate decisions). Furthermore, there is a wide range of inputs to subjective feelings including interoceptive and somatosensory inputs, physiological states such as thirst and hunger, and emotional states. Knowing how these subjective feelings are categorized into emotional states may help people gain insight into these processes which influence wellbeing.

We have chosen to replicate and extend portions of the study “Maps of Subjective Feelings” by Lauri Nummenmaa, Riitta Hari, Jari Hietanen, and Enrico Glerean (2018). This study can be found at the website: <https://www.pnas.org/content/115/37/9198>. The goal of these studies was to elucidate organizing principles of subjective experiences of emotions across individuals, and to uncover whether emotions can be grouped together to form broad categories of subjective experience. Luckily, they include an open source dataset for all experiments. We chose to replicate experiments 1 and 2 as they are the most relevant to the overall structure of subjective feelings into categories.

In Experiment 1, the authors’ specific aim was to map the basic dimensions of emotional states. To do so, the authors presented 339 volunteers with 100 words expressing core feelings (e.g., sadness) and bodily processes (e.g., heartbeat). The volunteers were then asked to rate these words using five different scales corresponding to the following five basic dimensions: 1) bodily saliency, 2) mental saliency, 3) emotional intensity, 4) controllability, and 5) lapse or the relative frequency of experiencing each emotion. This allowed researchers to determine the associations between the core dimensions of subjective feeling (a priori defined by researchers) and the various subjective feelings they sampled. All tested subjective feelings were associated with salient mental and bodily sensations but the relative strength of these differed greatly across feelings. The majority of subjective feelings were associated with emotional intensity (either positive or negative) and were related to the controllability of those feelings (i.e., pleasant feelings are more controllable).

In Experiment 2, the author’s goal was to map the mental feeling space based on similarity ratings. They used a density-based clustering (DBSCAN) which is an unsupervised machine learning non-parametric technique that groups together points based on neighborhoods. This technique has only two free parameters: 1) the minimum distance between two points which defines the points as being neighbors, and 2) the minimum number of points to form a neighborhood or a dense region. After getting clustering from DBSCAN, the authors used a t-distributed stochastic neighbor embedding (t-SNE) to graphical show this clustering in two dimensions. This technique is similar to a principal component analysis (PCA) but is a non-parametric technique for reducing high dimensional data (such as similarity ratings between 100 subjective feelings). The essence of t-SNE is that it finds patterns in the data by observing clusters of similarity in data with multiple features. Because it is a dimensionality reduction technique, it maps multi-dimensional data to lower dimensional space and is primarily used for visualization. The authors combine these two approaches to demonstrate high separability and clustering of the DBSCAN clusters on two factor loadings in t-SNE.

Our specific goals for replication was to reproduce the results and graphs for Experiment 1 and the results and main graph for Experiment 2. We were less interested in the rest of the paper and felt this would be good enough for a replication project. For our extension, we wanted to try additional clustering techniques other than DBSCAN to see how the authors’ results may change with other algorithms.

**REPLICATION METHODS**

*Obtaining data and importing into R*

The authors freely provided their data on a public repository, which we downloaded. However, the original data were encoded in Matlab’s proprietary format. In order to replicate the analysis using R, we wrote custom Matlab scripts to extract the relevant data and save them as CSV files that we could read using R.

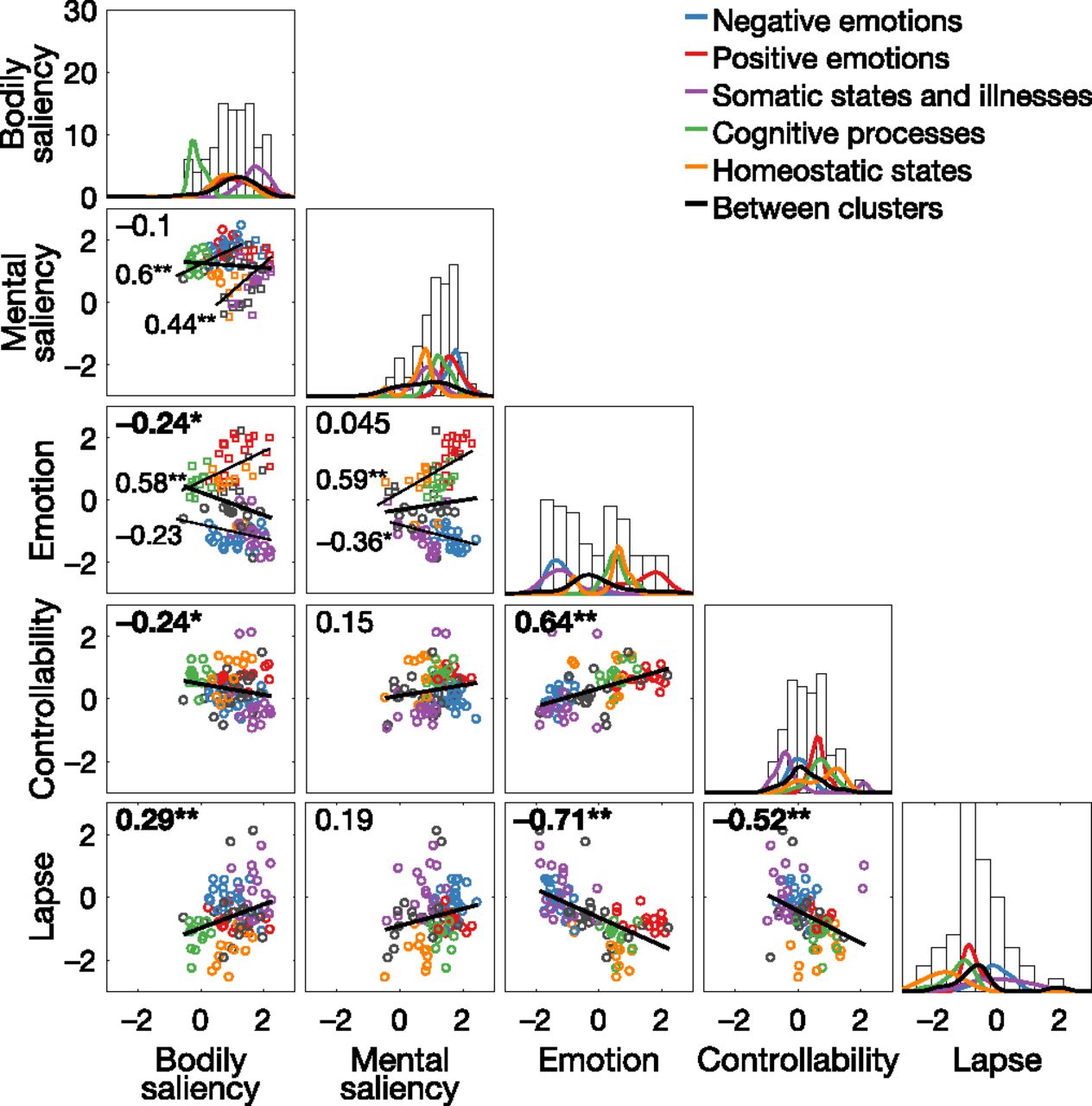
First, we extracted subjects’ mean ratings of how 100 common sensations varied along the following five dimensions: 1) the intensity of body sensations, 2) the salience of the mental experience, 3) the emotional valance (positive vs negative), 4) the perceived controllability of the sensation, and 5) how frequently the emotion occurs (operationalized as the time elapsed since the last experience of the sensation).

Second, we extracted results from the authors’ clustering analysis, which used machine learning techniques (i.e., DBSCAN, K-means, and hierarchical clustering) to categorize these 100 sensations into groups of related sensations. Third, we extracted the Cartesian coordinates for where each of the 100 sensations were placed in two-dimensional space using a dimensionality-reduction algorithm (t-SNE), which allowed us to visualize the results of the clustering analysis.

Finally, in order to replicate the clustering analysis using R’s implementation of the DBSCAN algorithm, we extracted the dissimilarity matrix for the 100 sensations. This dissimilarity matrix contains Euclidean distances quantifying subjects’ ratings for how similar (or equivalently, how dissimilar) any two sensations were judged to be.

*Goal 1: replicate Figure 1*

In the original paper, Figure 1 showed results from two analyses conducted in Experiment 1. First, for each of the 100 sensations, the figure plotted the Spearman correlation between subjects’ ratings for five author-defined dimensions (body saliency, mind saliency, emotional valence, controllability, and lapse). These scatterplots also included the ordinary least squares linear regression line drawn through the points. Each of the sensations visualized as a scatterpoint was color-coded according to how a machine learning clustering algorithm grouped related sensations in Experiment 2. Second, the figure showed histograms showing how 100 sensations’ mean ratings were distributed along each of the five author-defined dimensions.

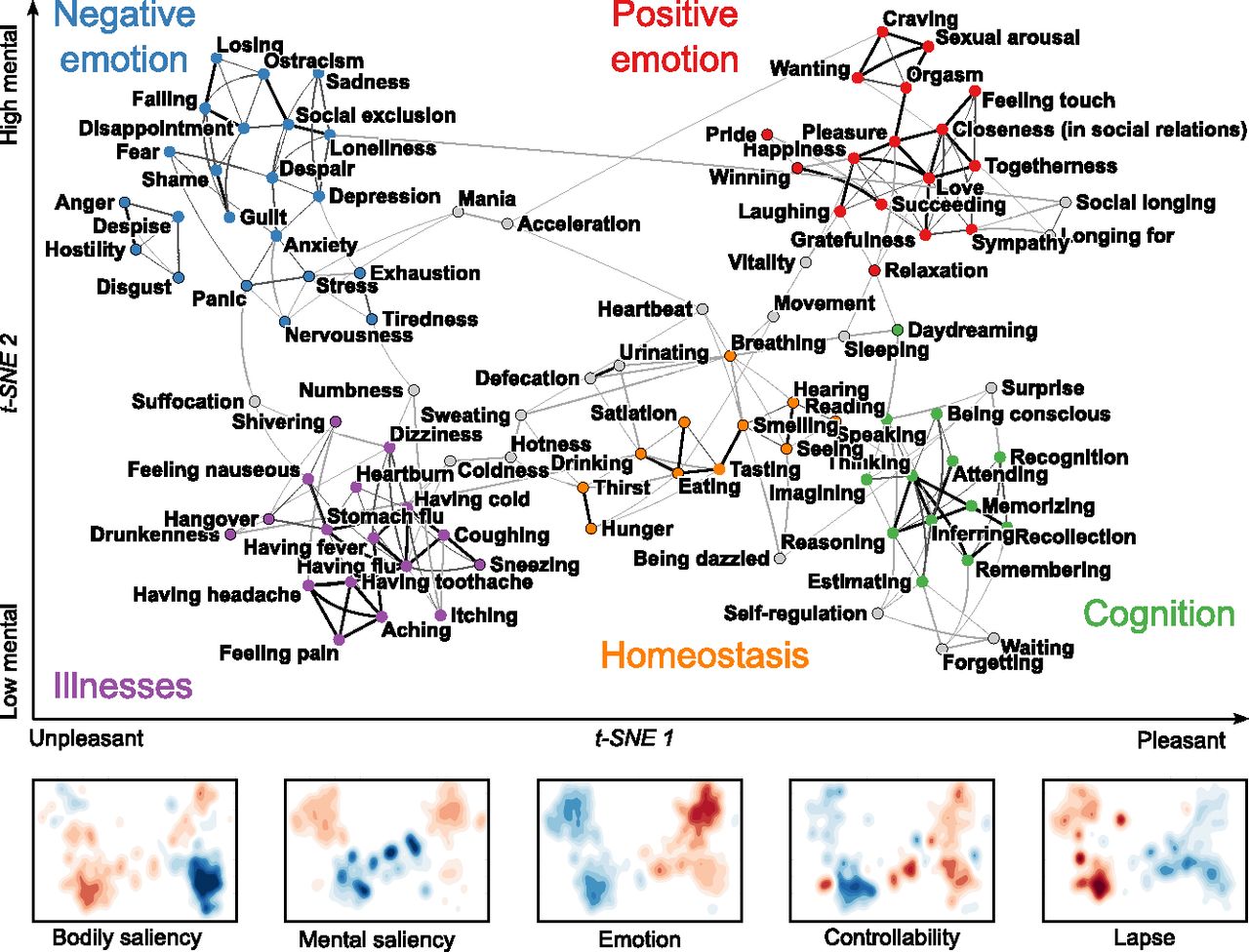


**Figure 1.** The figure that was reported in the original manuscript, provided as a visual reference.

In order to replicate this figure, we used the tidyverse library to tidy and visualize the data. We then used the cowplot library to “merge” all subplots into a single figure. Spearman correlations and p-values were calculated using the cor and cor.test functions in base R. For the scatterplots, we chose to report only the overall Spearman correlation, whereas the authors also reported Spearman correlations for each algorithm-defined cluster. Similarly, for the histograms, we chose to visualize only the overall distribution, whereas the authors also visualized distributions for each algorithm-defined cluster.

*Goal 2: replicate Figure 2*

In the original paper, Figure 2 visualized the results of the authors’ clustering analysis, which grouped the 100 sensations according to participants’ judgments of their similarity. They used a Matlab implementation of the DBSCAN algorithm to perform this analysis. In short, DBSCAN is an unsupervised machine learning algorithm that can identify irregularly-shaped clusters in density-based datasets, even in the presence of noise and outliers. Compared to other unsupervised machine learning approaches like K-means, DBSCAN has the following advantages: 1) it does not require the researcher to define a specific number of clusters to extract from the dataset; 2) it can find irregularly-shaped clusters, which is useful when the data is high-dimensional; and 3) it identifies outliers and is particularly robust to outliers.



**Figure 2.** The figure that was reported in the original manuscript, provided as a visual reference.

To replicate this analysis in R, we first extracted the Euclidean distances between all pairwise combinations of the 100 sensations, thereby creating a dissimilarity matrix. Next, we needed to define the two parameters used by DBSCAN. The first parameter, minimum points (MinPts), defines the minimum number of neighbors that should be contained in a neighborhood, with a radius defined by the second parameter, epsilon (eps). Therefore, eps defines the radius of a neighborhood surrounding specific points in density space. To summarize in simple English, our DBSCAN clustering method is a recursive process of finding clusters that containing at least MinPts members, then finding surrounding points that are within the radius eps and adding them to existing clusters. Points that do not fall into clusters using this method are therefore identified as outliers.

In order to search for the optimal value of eps, we used k-Nearest Neighbor Distances on the dissimilarity matrix, as implemented in the function dbscan::kNNdistplot. We used MinPts=5, the value used by the authors according to their Matlab code. This resulted in a plot of points sorted by distance, and their corresponding 5-Nearest Neighbors Distances. The optimal value of eps can then be found by searching for where there is a “knee” in the plot. Our visual analysis was concordant with the original analysis, which used eps=0.2565 according to the authors’ Matlab code.

With these DBSCAN parameter values, we used the fpc::dbscan function on our dissimilarity matrix to identify clusters of related sensations. To visualize our DBSCAN algorithm results, and to compare our results with the authors’ original clusters, we used the t-SNE algorithm (t-distributed stochastic neighbor embedding). This algorithm is a dimensionality reduction technique conceptually akin to a nonlinear PCA, and aims to represent dissimilarity in high-dimensional space by placing points closer together/further apart in two-dimensional space. Importantly, the use of t-SNE was not an analytic technique, but rather a convenient method for visualizing dissimilarity between sensations, and clusters of sensations. We loaded the results of the authors’ t-SNE analysis, which provided us with Cartesian coordinates for where each of the 100 sensations should be placed in two-dimensional space.

To recap in plain English, Figure 2 is a two-dimensional representation of a potentially high-dimensional “sensation space” and is therefore a map of sensations arranged by the t-SNE algorithm and grouped using the DBSCAN algorithm.

**EXTENSION METHODS**

*General strategy*

Using the DBSCAN algorithm, the authors drew strong conclusions about how sensations might be structured. However, this leaves open the question of whether alternative clustering methods would have led to similar interpretations of sensation structure. If not, this would imply that the conclusions drawn by the authors are highly contingent on the analytic method. Curiously, the authors’ public repository contained results from two other clustering techniques, which were not reported in either the manuscript or supplement. Therefore, we extended the published work by comparing the published clustering results against alternative clustering methods.

*K-means clustering*

One of the simplest and most popular clustering techniques in unsupervised machine learning is K-means clustering. The intuition behind this technique is conceptually related to ordinary least-squares regression: the researcher specifies that they want to extract K=k clusters from the data, and the algorithm assigns data points to clusters by minimizing the within-cluster sum of squares until an “optimal” clustering is found that minimizes the sum of squares. Here, the authors specified that they wanted k=5 clusters, to match the number of clusters obtained by DBSCAN. To compare the results of the K-means clusters against the DBSCAN clusters, we mapped sensations to the same Cartesian locations in t-SNE coordinate space, then color-coded them according to their clusters. We assigned qualitative labels to these clusters based on our best guess of what real-world structure found by the algorithm.

*Hierarchical clustering*

Another method for clustering dissimilarity data is hierarchical clustering, an algorithm whose goal is to create a dendrogram reflecting the hierarchical structure of multilevel data. Therefore, in a three-level dendrogram, two distinct clusters at the third level would be subcomponents of the same cluster at the second level. The authors used an agglomerative approach (i.e., a bottom-up approach), in which each datapoint is first treated as its own cluster. The algorithm then recursively merges the two clusters that are closest together until all points have been merged into a single cluster (i.e., the first-level parent cluster). After implementing this algorithm, the authors identified the Nth level where the minimum number of points within the smallest cluster was equal to the smallest number of points within the DBSCAN clusters. Since they set the DBSCAN parameter MinPts=5, this ended up being a minimum of five points within the smallest cluster. As we did with K-means clustering, we mapped the sensations using t-SNE coordinates, then color-coded according to the identified clusters. Once again, we assigned qualitative labels to these clusters based on our best guesses about the underlying structure.

**REPLICATION RESULTS**

**EXTENSION RESULTS**

**CONCLUSIONS**

We were able to successfully replicate both Experiment 1 and Experiment 2 as well as extend the authors original findings using different clustering algorithms. While we experienced many of the joys of open science, there were major difficulties in successfully reproducing the author’s results. The authors stored their repository in an unusual format and most of their data was stored in an inaccessible format (MatLab files). Unless researchers have access to a working copy of MatLab, which is not a free software, replication using just the provided files would be difficult. We created CSV files for all of the replication figures and believe the authors should have done so to increase accessibility.

Aside from the difficulty of retrieving the data, the authors failed to report critical free parameters in their clustering and dimensionality reduction algorithms. For example, DBSCAN requires the user to specify the minimum distance between points for those points to be considered part of the same neighborhood. While the authors define that point in their MatLab analysis file, they do not report that value in the Methods for the paper. While these parameter values can be justified, the number of total “clusters” from subjective feelings can drastically differ depending on these free parameters. Without knowing these details, it would be hard to make strong claims about the universality of the subjective feelings categories authors pulled out using this technique. Furthermore, the authors failed to mention how t-SNE algorithm will produce different results depending on the seed in the paper. While they do describe that t-SNE was done for graphing purposes, using different seeds when running this non-parametric algorithm will produce vastly different results and ultimately the graph will look less convincing depending on the random seed. We feel that the authors should have mentioned this aspect because their t-SNE is just one visualization to show how the clusters can be shown on two dimensions but is not the only representation.

Finally, while the authors could not have predicted these differences, we found clear discrepancies between the results of our R DBSCAN and their MatLab DBSCAN. They do report that they used MatLab software and we found it surprising how simply changes in the software running similar algorithms can produce different results. For example, in our DBSCAN we found that hunger, pride, and tiredness all were differently clustered than the authors original. This is largely acceptable because the authors make no specific claims about these feelings but rather the general structure of the map of subjective feelings, however it is important to note.

In our extension, we used different clustering algorithms including KMEANS and hierarchical clustering to demonstrate two different methods for organizing the map of subjective feelings. What is impressive about this multi-faceted approach is that the general clustering seem very consistent across algorithms and lends support to the idea that there is consistent structure in the participants’ similarity ratings. However, it is important to note since there are inconsistencies regarding specific feelings as noted above. For example, the KMEANS clustering seems to identify significantly more negative emotions than the DBSCAN originally classified and in doing so loses any classification for “cognitive” feelings. Given the authors had a priori reasons to believe some feelings were cognitive it is accessible to use the DBSCAN clustering, but it would make us hesitant to make any strong claims about the universality of these clusters as different techniques seem to pick out different groupings. Overall everyone in the project enjoyed this opportunity to learn new machine learning methods and replicate an important and impressive paper on the structure of emotions.