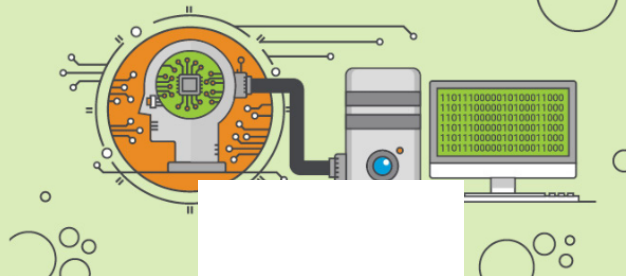


 [Data Science](#)[Leadership](#)[Professional Development](#)[Prospect Research](#)[Relationship Management](#)[About](#)

Creating Clusters for
Strategic Segmentation



DATA SCIENCE

Creating Clusters for Strategic Segmentation

By *Barron Cato* | AUGUST 16, 2018

Michelle Obama once said, “Just try new things. Don’t be afraid. Step out of your comfort zones and soar, all right?”

This quote best describes our foray into something game changing at University of Washington (UW) Advancement: k-means clustering.

K-means clustering is a machine-learning algorithm that takes a diverse and varied population and divides it into a chosen number of homogenous groups. This clustering enables the segmentation of donors by giving tendencies and creates donor personas that are easy to communicate across an organization.

This article reflects on why we ventured into k-means clustering, how we performed it at UW, lessons learned from the process and what we ultimately got out of it.

Why UW Advancement Ventured Into K-Means Clustering

Four years ago, we transitioned to a full-time analytics team, and cluster analysis was one of our first projects. Up to that point, we had built multiple predictive and descriptive models, but had never fully completed a clustering project. The team’s transition into a full-time analytics team also corresponded with my own transition from SPSS to R, which added to the learning curve. For reference, this article reflects the lessons learned as we discovered how to perform k-means clustering in R.

While we had never completed a clustering project, we were already familiar with clustering because we had purchased lifestyle clusters from a vendor. The vendor’s cluster segments were very easily understood when explained to different units in UW Advancement, which was a great sign. Our other segmentation tools, like our engagement model, were not as easily understood across units because they involved calculated scores. This required our team to spend considerable time educating the rest of UW Advancement on the meaning and use of these segments.

The ease of communicating the vendor-based segments motivated our team to find natural groupings based on our own data. We decided to base the groupings on our giving data because our giving data were robust and allowed us to segment a significant portion of our database. To find natural groupings based on our own data, we dove into k-means clustering.

In simple terms, as stated in the introduction of this article, k-means clustering is a machine-learning algorithm that takes a diverse and varied population and divides it into a chosen number of homogenous groups. The algorithm does this automatically; we do not know ahead of time what features the procedure will use to characterize each group. By exploring the resulting clusters, we may develop a better understanding of subgroups in the population to find meaningful and actionable insights to drive business decisions.

K-means is a relatively simple algorithm, but it is still widely used and provides a foundation to understand more advanced clustering algorithms. Like other machine-learning approaches, k-means has strengths and weaknesses, which we assessed for our project. The k-means strengths and weaknesses we identified are as follows:

Strengths	Weaknesses
<ul style="list-style-type: none">• Relatively simple to learn and apply• Interpreting the results requires very little technical or statistical expertise• The resulting clusters generate insights about our population, which can be used for action	<ul style="list-style-type: none">• More sophisticated clustering algorithms exist• Requires a reasonable guess as to how many clusters naturally exist in the data• It is not guaranteed to find the same set of clusters each time it is used—even on the same population!

So, our goal was to find natural groupings using our own data via k-means clustering; and the ultimate question we wanted to answer was, “What segments could we create from our giving data?”

But before we could answer that ultimate question, we had to determine which giving data to use for our clustering.

Determining Which Giving Data to Use for Clustering

To determine which giving data to use, the director of analytics and I held a brainstorming session. Through this session, we decided to create clusters based on the components of our Recency, Frequency and Monetary Value (RFM) score. See the components in the graphic below:

Gift Recency
Years Giving
Gift Frequency
Lifetime UW Credit
Largest Gift Amount

K-means

For background on our RFM scoring at UW Advancement, we choose to determine our score in a way that provides simplicity for our fundraisers: using a score of 0-100. This score summarizes our donor giving history, and we were curious to see if we could discover more nuanced insights through cluster analysis.

We considered this an experiment, because we did not know if the clusters would actually help us discover more nuanced insights. As a proactive project, we had the freedom to fail; we thought the worst-case scenario would just be learning about k-means analysis and informing future use.

With our background knowledge and expectations in place, we went forward with the clustering process in three steps.

Process Step 1: Transforming the Data Prior to Running K-Means Analysis

Some data transformations were necessary to effectively run the k-means analysis. I learned how to do these data transformations and cluster analysis from books on machine learning and R. These books gave me the foundation to get started, and I used websites like Stack Overflow to find answers for specific coding issues.

Armed with this newfound knowledge, I first used a histogram to visualize our variables and discover the shape of the distribution (e.g., left, right skewed or normally distributed).

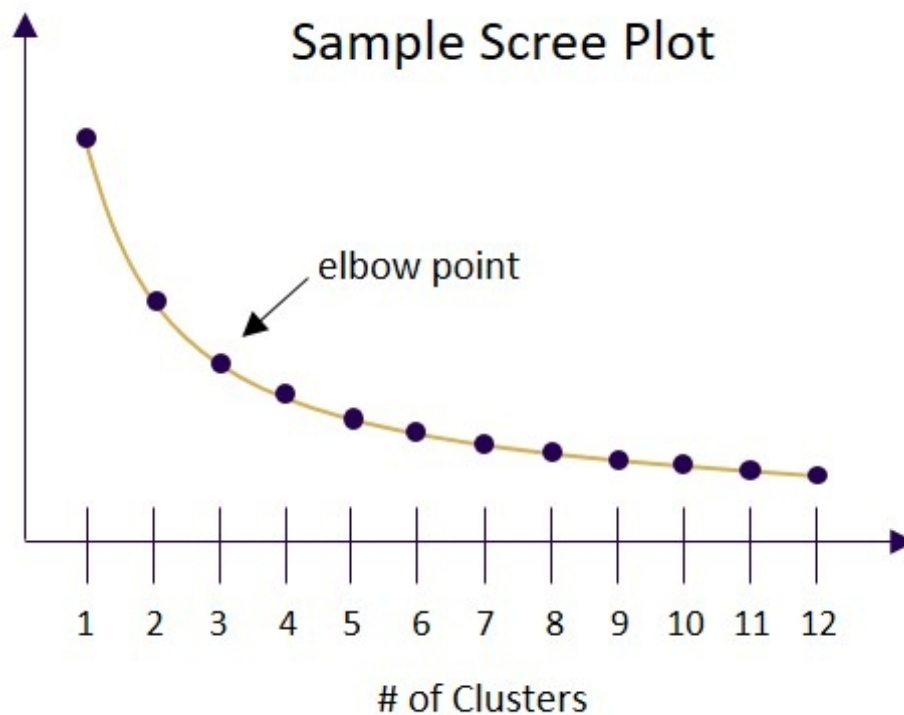
The lifetime credit and gift amount variables were highly skewed, so I transformed them into logarithms to reduce the skew. I then standardized all the variables into z-scores (see below for definition) so they were using the same scale. This is a common practice when performing k-means analysis.

A z-score indicates how many standard deviations a value is from the mean.

Since my population was all donors, I did not worry about missing data. So once my data set was transformed, my next step was to determine how many clusters to build and run the k-means algorithm.

Process Step 2: Determining Initial Cluster Number

To determine my initial cluster number, I created a scree plot, which displays how the measure of model quality changes with the number of clusters. The optimal plot has an “elbow” point, which is where the model improvement slows as the number of clusters increase (see the chart below to see how this point resembles an elbow).



Starting with the plot's suggested cluster number at the elbow, I tweaked it as necessary to generate the best distribution of clusters.

Process Step 3: Running the First K-Means Analysis

With great anticipation, I ran my first k-means analysis. The initial results were quite a surprise: skewed clusters! I thought my transformations would eliminate skew, but it turned out we had one donor who gave at such a high level, they were a cluster of one. This impacted the other clusters and created a skewed distribution. I attempted to change the cluster number to see if this would bring the lone wolf into another cluster, but that was not successful. Ultimately, I decided to remove that one donor and run the analysis again. This resulted in a better distribution of clusters. I then manually added the donor to the cluster with similar donors. As an aside, I've run this analysis in subsequent years and, as a result of our capital campaign, we have enough donors close enough to our top donor to eliminate the solo cluster.

After the clusters were created, it was up to me as the analyst to supply a meaningful, actionable label to each cluster. In order to do this, I needed to interpret the cluster analysis results and communicate them to our non-technical stakeholders. I learned three distinct things during this labeling process that I would like to share:

Labeling Lesson 1: Simplify Cluster Interpretation

The clusters are generated with the average z-score per variable in each cluster. Each cluster includes all the variables, but they are at different magnitudes, which differentiates the clusters. I initially attempted to create cluster labels through interpreting the z-scores, but this became challenging when clusters had similar z-scores on multiple variables.

I eventually found an easier way to label. I appended the clusters back to the original data set (something I would do anyway) and used the summary statistics of the non-transformed variables, which were more intuitive and easy to interpret. I also appended the median age and RFM scores for each cluster to aid in labeling them.

Cluster #	Recency Z-Score	Recency Median Value (months)
1	2.92753636	339
2	0.02663416	54
3	-0.25576311	9
4	0.39004035	90
5	-0.11705960	47
6	1.02118074	150
7	-0.25706205	12
8	-0.45010797	5

Labeling Lesson 2: Don't Include Cluster Numbers

The k-means algorithm generates a number for each cluster it creates. These numbers are the “labels” and have no rank or order — even though it looks like they do. When I first started, I included the generated cluster numbers along with the donor cluster labels I created. This was fine until I re-ran the clusters a year later and saw that while the actual clusters stayed the same, the number labels changed. This confused my target audience even when I tried to explain to them that the numeric order did not impact the clusters.

In hindsight, my error was including the cluster numbers at all. They were not necessary once I had created my own cluster labels, and they changed each time the analysis was refreshed with new data.

Labeling Lesson 3: Labels Matter

I was initially more concerned with being technically accurate than reaching my target audience when I first generated the cluster labels and descriptions. In the back of my mind, I knew the cluster descriptions were based on median summary statistics, which meant there would be cases in each cluster that didn't perfectly match up with the description. I tried to compensate for that in my cluster labels by hedging the names and thoroughly describing the summary stats used to create the label. While this made me feel good as an analyst, the director of analytics pointed out that I had artificially inserted a hierarchy. When I introduced the term "Best" in one of our clusters, that implied that was the group to focus on. This wasn't my intent in labeling the clusters, but I hadn't thought through how our target audience would interpret them.

So, we changed the hedging in the titles (e.g., "Newer" to "New"), removed the hierarchy (e.g., "Best" to "Invested") and made the descriptions less technical. We made sure the labels were terms non-technical people could relate to. For example, we used "Lapsed Donors" to differentiate the cluster representing the recent former donors from former donors who had not given for many years ("Former Donors"). The cluster average did not meet our normal definition of a lapsed donor, but that cluster label was a good representation of that cluster. We made sure to explain this to our target audiences so they understood.

Cluster Label	Description
"One and Done's"	Often Memorial Gift donors but many other reasons for these single gift donors
New Donors	Just started giving in the last three years but consistently in that time
Leadership Donors	Give consistently and over a long period of time with high lifetime credit and leadership level gifts
Lapsed Donors	Donors who haven't been giving to the UW for over 10 years
Young Donors	Similar to New Donors but younger and giving at lower levels
Former Donors	Consider these individuals almost non-donors. Haven't made a gift for over 20 years
Loyal Annual Donors	Loyal annual donors who look like Leadership Donors but giving at lower amounts
Invested Donors	Major gift and planned giving donors with long history and/or large gifts

Insights Gleaned From the Clusters

The insights created from the clusters enabled us to see the nuance between different donors like our “One and Done’s” and “Young Donors.” They both had similar RFM scores, but they were two different donor types. The insights also made the case to not always reject low RFM scores because that could unnecessarily remove donors in our “Young Donors” cluster.

Cluster Label	Median RFM	Median Age
“One and Done’s”	12	66
New Donors	17	48
Leadership Donors	40	65
Lapsed Donors	5	64
Young Donors	13	33
Former Donors	12	54
Loyal Annual Donors	33	55
Invested Donors	70	71

Putting the Clusters to Work

Our clusters were put to good use.

We rolled out our donor clusters to our UW Marketing team, who used them for a major advertising campaign. They only wanted to advertise to certain segments of our database and used the clusters to determine which populations to target. I segmented the key zip codes by cluster, and the marketing team prioritized and targeted their outreach based on the selected cluster descriptions. They reported back that the advertising campaign successfully reached their target audience based on online click-thru stats and event attendance. They credited the donor clusters with playing a vital role in their success.

Our Annual Philanthropy team also used the donor clusters to segment a March mailing. We took a risk and removed the “One and Done’s” and “Former Donor” clusters and then used a predictive model to prioritize the remaining population. This reduced the overall population by two-thirds compared to the previous year, which reduced mailing costs while raising a similar amount of donations. We

intend to continue using the clusters for future solicitations in Annual Philanthropy.

Through our experiment, we discovered that clustering by giving data was a great way to segment donors by giving tendencies; the clusters created donor personas that were easy to communicate and could be used in all of UW Advancement.

In the end, it turned out Michelle Obama was right.



Data Science

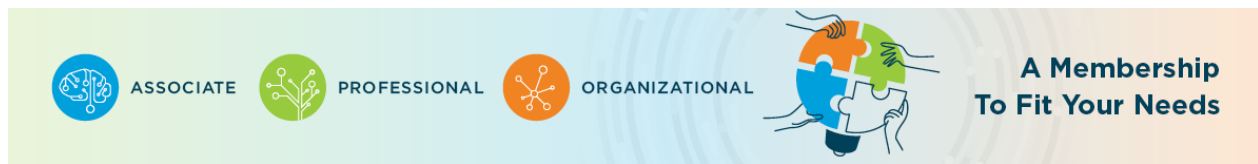
Education



**Barron Cato**

Assistant Director of Fundraising and Engagement Data Science, University of Washington

Barron Cato is the assistant director of fundraising and engagement data science at the University of Washington. His 20 years in non-profit development work has focused on fundraising, prospect research and analytics. Barron has served on the Northwest Development Officers Association Board (NDOA), the Apra Data Analytics Symposium Committee and the Apra Awards Committee (as chair). He has presented on prospect research and analytics at NDOA, UW fundraising certificate courses, Apra-NW, Apra Data Analytics Symposium and Apra OverDrive. Barron received the Apra Margret Furhy Award in 2014.



MORE RECENT ARTICLES FROM *CONNECTIONS*



Apra Fundamentals: An Energizing Way to Build Your Prospect Development Career



Painting the Big Picture: Tips for Prospecting in the Arts



Closing Time: Calculating Risk to Encourage

Proposal Velocity

330 N. WABASH AVE. SUITE 2000, CHICAGO, IL 60611

312.321.5196 | INFO@APRAHOME.ORG



[LOGIN](#)

