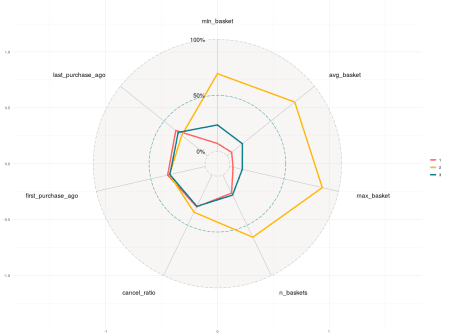
TL; DR

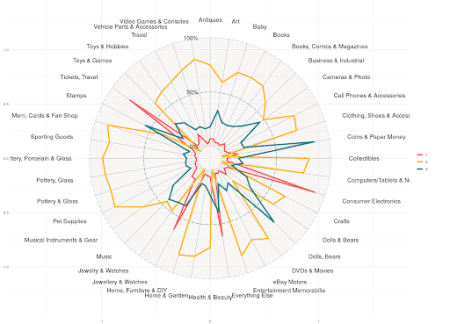
We used a [Kaggle](https://www.kaggle.com/carrie1/ecommerce-data) database to show you how to separate your customers into distinct groups based on their purchase behavior.  With this method, store managers can customize interactions with  existing and potential customers to increase loyalty and eventually, all of the goodies that come with consistent purchases.  For the R enthusiasts out there, I demonstrated what you can do with r/stats, ggradar, ggplot2, animation, and factoextra.

Intro

We find ourselves in a time when humanity has noticed the importance of data collection. Every financial transaction, every trip or meeting with friends can be registered in one of the billions of databases. The tools to collect data points and store them have improved drastically in the last several years, as well as the tools to make sense of the quantitative and qualitative data. We have found that even businesses that collect data points carefully and deliberately are often still sitting on a potential treasure chest of uncovered and, consequently, un-leveraged business intelligence.

Imagine a situation in which you lead an online shop. Wouldn’t it be useful to identify separate groups of clients that show different shopping behaviours? For this blogpost I have put myself in the role of an online shop owner. Spoiler alert: based on the available data and **Machine Learning** methods We extracted three specific customer profiles. You can sneak a peek at the profiles in the radar charts below.





Made with ggradar package.

I detected that my customers fall into three groups.  Each group can be characterized by product choice, frequency and amount of purchases, as well as type of purchases.

I was even able to propose some promotional strategies to encourage each group to visit my shop in the future. If you want to learn the magic that stands behind the conversion of data to pricing and promotion strategy, as well as  what hides behind the above radar charts, I encourage you to read the next sections.

What is customer segmentation?

The magic that allowed me to detect customer profiles is called **customer segmentation**. Customer Segmentation is a series of activities that aim to separate homogeneous groups of clients (retail or business) into sub-groups based on their behavior during the purchase. As a rule, each of the designated groups reacts differently to the product offered, thanks to which we have the opportunity to offer differently to each of them.

Customer segmentation – analysis

Before each analysis, it’s essential to explicitly state questions and expectations about the data and results. During my analysis I’d like to answer the following questions:

1. What do I expect from my analysis?
2. Is the data I have sufficient for my analysis expectations?
3. Which algorithm should I use?
4. Hypothesis: Extracted groups allow me to differentiate customers in a visible way.
5. How am I going to use the results?

What do I expect from my analysis?

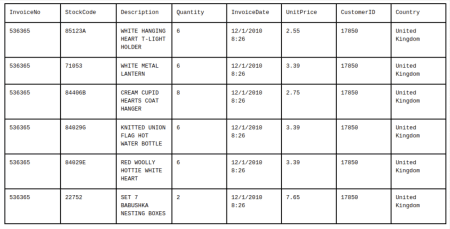
In this case I’m the owner of an online shop. I store details about each order and transaction. I’d like to learn more about my customers and find out how can I attract them and encourage them to use my online shop in the future.

My first idea is to find groups of similar customers based on shopping behaviour, then analyse each group separately and find out what is important for each user while making an order.

Is the data sufficient for my analysis expectations?

For my analysis I’m going to use E-commerce data that you can find here:  <https://www.kaggle.com/carrie1/ecommerce-data>.

A quick spike into the data:



Based on such data I can extract lots of information about a customer’s shopping behavior.

Useful info for my analysis can be:

1. Average basket value.
2. Basket value range (min, max).
3. Order frequency.
4. Tendency to cancel an order.
5. User’s activity (first and last purchase time)

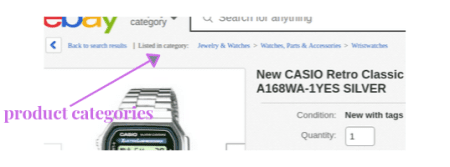
To extract the required information, I aggregated the data twice. The first aggregation is based on “InvoiceNo,” the second one is dependent on “CustomerID,” so each row describes one customer.  I also skipped using “StockCode” and “Country” variables. The “description” column will be used later.

Such information is presented in the table below:



I still haven’t used the very important variable “Description”. It stores information about which products interest my customers the most. How can we use this information in the analysis?

There are currently 3883 distinct products within the data. It would be useful to group the product by category, but this data point wasn’t included in the set.  Since I didn’t want to come up with product categories on my own, I decided to scrape the data from a popular online shop that has the notion of a “product category”  (I decided to use eBay.

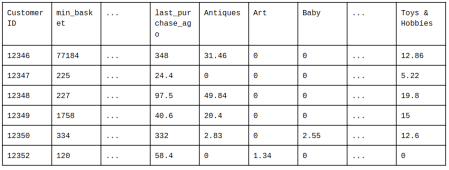


Below is a list of selected products and the groups we matched after scraping:



Now we can switch from 3883 “Description” values to 41 “Category” values.

Let’s use this information to create new sets of variables that store information about how much each customer spends in each category.  We now have our final dataset:



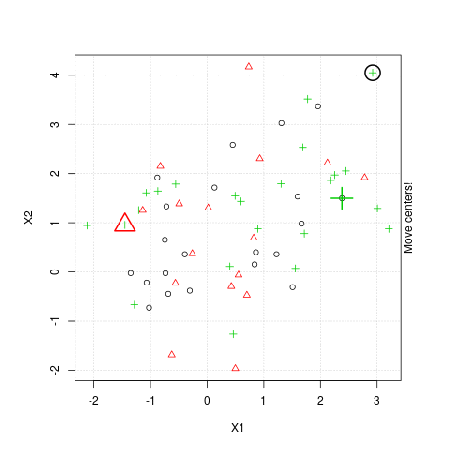
Going back to the topic question: Is the data I have sufficient for my analysis expectations? The answer is Yes.  We now store information about the users’ spending behavior, their products of interest and some basic information about the users’ activity.

Which algorithm should I use?

There are plenty of algorithms that are commonly used for segmentation. You might have heard about the very popular k-means, hierarchical clustering, latent class analysis, or even self-organizing maps.

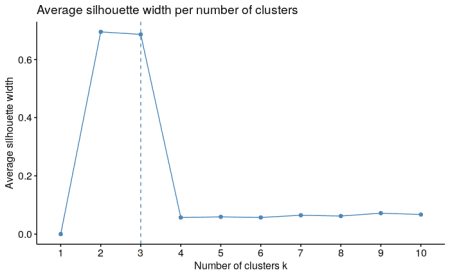
The question is which algorithm is best for my particular data set. The standard approach is to test out each algorithm and compare them according to existing measures.

The most popular algorithm used for partitioning a given data set into a set of k groups is k-means. We’ll use this in our case.  The k – parameter is set pre-specified, but the post-analysis can help you choose the best value (**silhouette** or **gap statistic**). The algorithm tends to minimize inter-cluster variation that should result with separating homogeneous groups. The way the algorithm works is shown below:



Applying the k-means algorithm

As we learned before, the k-means algorithm doesn’t choose the optimal number of clusters upfront, but there are different techniques to make the selection. The most popular ones are **within cluster sums of squares**, **average silhouette** and **gap statistics**.  The silhouette statistic for a single element compares its mean inner-cluster distance to the mean distance from the neighbouring cluster. It varies from -1 to 1, where high positive values mean the element is correctly assigned to the current cluster, while negative values signify it’s better to assign it to neighbouring one.  Here we present average silhouette across all data points:



Made with factoextra package

As you can see above, the optimal number of clusters is 2 or 3. Even if 2 clusters shows the maximum of the average silhouette statistic, 3 clusters shows similar value and we tend to find more groups in our analysis. So let’s choose 3.

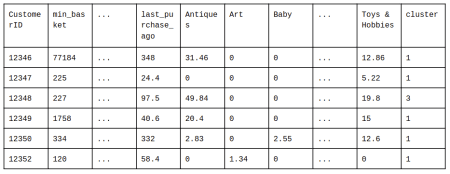
To sum up, we’re going to use the k-means algorithm with 3 clusters. We can do it with one line of code:

model <- kmeans(customer\_data\_scaled, 3)

Let’s extract the chosen clusters from the created model and take a look at the data again:

customer\_data$cluster <- model$cluster

head(customer\_data)

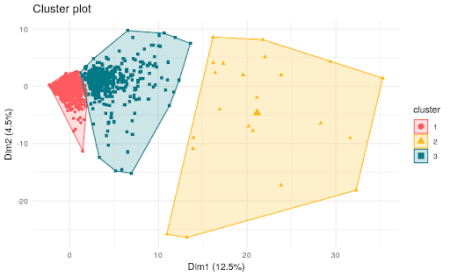


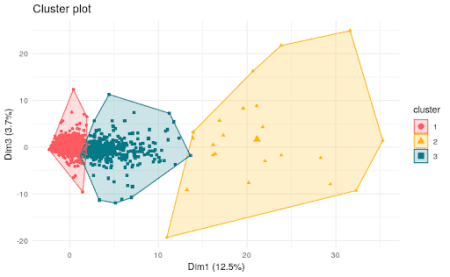
How can we verify if the clusters were extracted correctly?

Our dataset stores 47 variables, so it’s impossible to compare assigned clusters across all variables (readable visualisations are restricted to a maximum 3 dimensions).

One of the most popular approaches that helps solve the problem is [Principal Component Analysis (PCA)](https://towardsdatascience.com/understanding-pca-fae3e243731d). PCA combines variables of a provided dataset to create new ones, called PCA components, that capture most of the dataset variation. Plotting clusters distribution across first PCA components should allow us to see if the clusters are separated or not.

For this case, let’s plot how clusters were distributed comparing the 1st vs. the 2nd, as well as the 1st vs.  the 3rd PCA components.





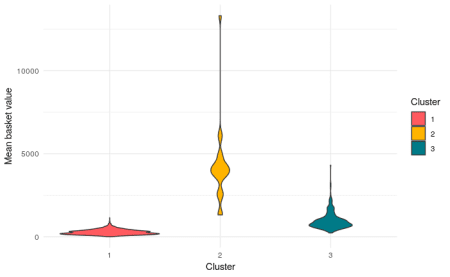
Made with factoextra package.

From the above plots we can certainly conclude that the 2nd (yellow) cluster is separate from the remaining ones. Clusters 1 and 3 are slightly overlapping, but each one covers high concentration groups of data points which is successful information in this analysis. As the PCA for the first three dimensions covers only 21% of the variance we may still expect that the remaining dimensions show even more exact separation of the clusters. To sum up, we’re happy with this result and we can now move to the next part of our analysis.

Hypothesis: Extracted groups will allow me to differentiate customers in a visible way

The plots above show cluster assignments across the first three PCA components (dim1, dim2 and dim3). How can we detect which indicators along 47 variables distinguish our customers?

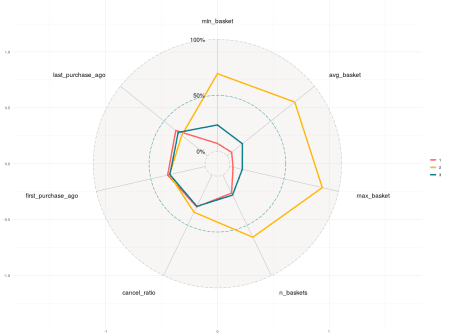
In general, it’s necessary to analyse distributions for each variable grouped by calculated cluster. A good approach that could be of use here are **violin plots**. Below we present a violin plot to show the differences of “avg\_basket” in each cluster:

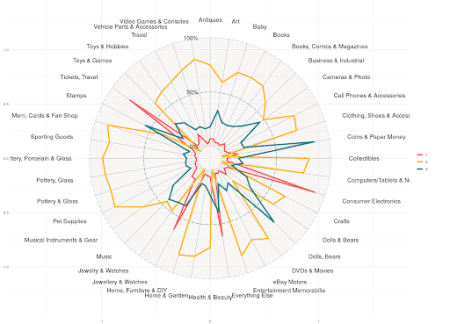


Made with ggplot2

For this variable we can detect significant differences in “avg\_basked” spending for each group. Nevertheless comparing all 47 variables profiles can be burdensome approach.

For simplification and the needs of this blogpost we’ll just check how the average value for each variable was distributed in each group; to do so I created radar charts that show all of the variables at once.





Made with ggradar package

The first chart sums up basket indicators (such as average basket value or total number of baskets) across the 3 groups of customers. The second one shows the tendency for buying a product in a specific category.

From the above summary we can detect a few simple characteristics about customers in each group.

Group no. 1 (red):

* Tends to spend a low amount of money for each basket.
* Is mostly focused on ordering electronics, tickets/travel and jewelry.
* The clients on average are also least active in the recent past.
* We can classify the group as typical **bargain hunters.**

Group no. 2 (yellow):

* Tends to spend a lot of money for each basket.
* They also order the highest number of baskets.
* Products of interest for the group are varied.
* The clients on average are also the most active in the recent past.
* We can classify the group as **regular customers**.

Group no. 3 (green):

* Mean value for basket based indicators (no specific behaviour shown).
* Strong interest of general group in product category “Collectibles and Art.”
* We can classify the group as **collectors**.

How am I going to use the results?

We were able to group our customers based on their purchase behaviour and we managed to detect meaningful factors for each group. The best way forward is to prepare specific interactions for each one.

Here are some ideas:

1. For group no. 1, we can offer selected promotions for products from their groups of interest. We could periodically send the discount offers by email or show the message right after the user logs in to our shop.
2. A big part of regular customers may be entrepreneurs, so they order wholesale quantities of products. We can prepare an offer for them to get an extra discount when they buy in bulk.
3. Collectors might be encouraged to return if we inform them about new and/or unique products from our line. We could even include recommendations from the appropriate **influencers**.