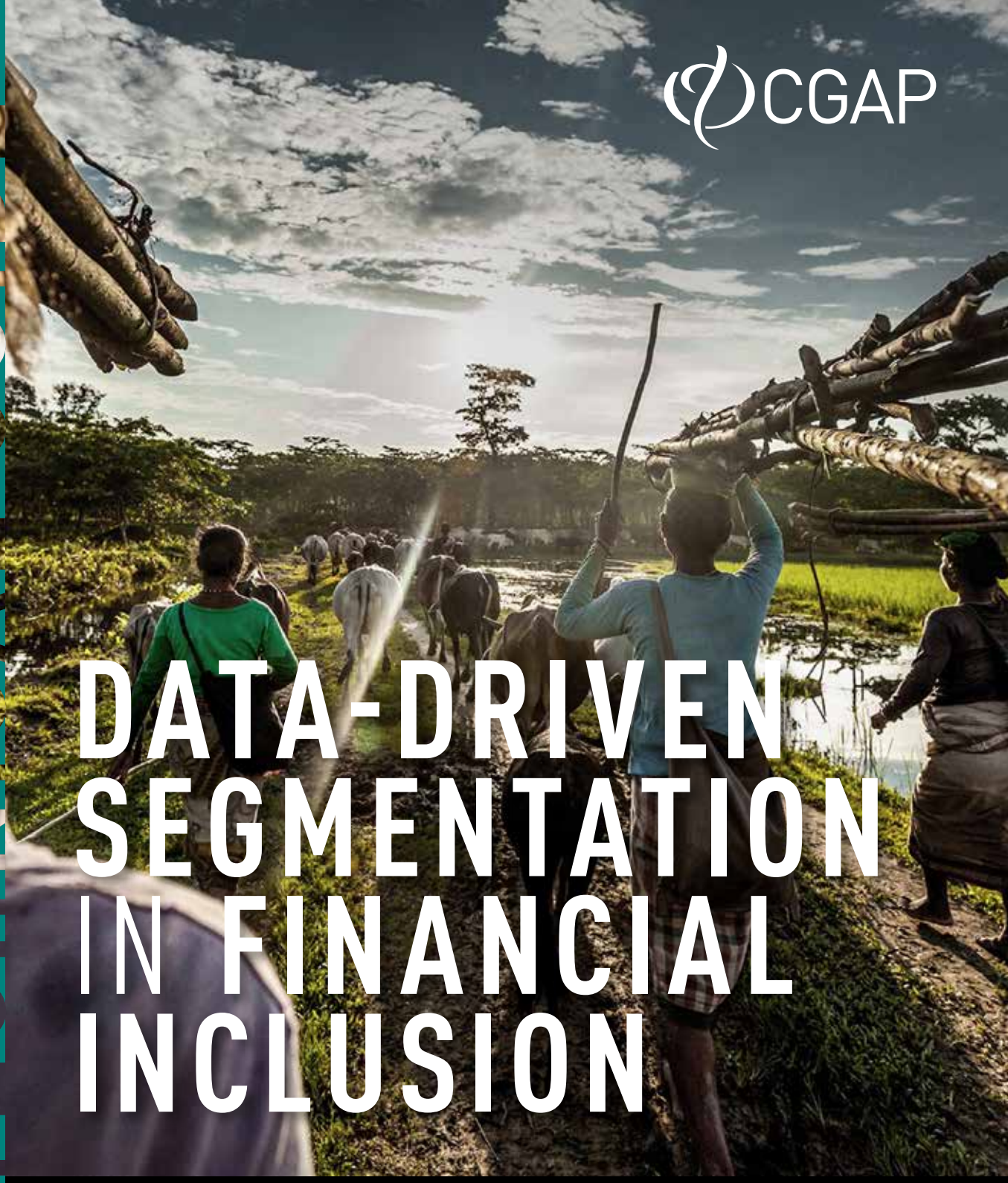
TECHNICAL GUIDE

CGAP



DATA-DRIVEN SEGMENTATION IN FINANCIAL INCLUSION

How financial services providers can use data analytics to better segment and serve customers

July 2019

Maria Fernandez Vidal, Dean Caire, and Fernando Barbon

Consultative Group to Assist the Poor
1818 H Street NW, MSN F3K-306

Washington DC 20433

Internet: www.cgap.org

Email: cgap@worldbank.org

Telephone: +1 202 473 9594

Cover photo by Sujan Sarkar, India.

RIGHTS AND PERMISSIONS

CONTENTS

EXECUTIVE SUMMARY

FINANCIAL SERVICES PROVIDERS CAN IMPROVE THEIR businesses by using segmentation to develop a more accurate understanding of their customers. Segmentation can benefit providers in many ways:

- Stronger and deeper customer relationships
- Improved product uptake
- Greater awareness of new product opportunities
- Flexibility and agility to adapt to customers' needs

It is often the case that providers in emerging markets adhere to traditional methods of segmentation—classifying their customers based on a single characteristic. This guide shows providers how they can use data analytics to understand their customers by performing more complex analyses and extracting insights that were previously hidden.

The first step toward segmentation driven by data analytics is to gather useful data. Most financial services providers have information about their customers' demographics and use of products. This technical guide shows providers how they can clean up and use this data to segment customers in powerful ways.

It also explains how providers can leverage a qualitative understanding of their customers to analyze the data available to them and gain useful customer insights.

INTRODUCTION

TO FEND OFF COMPETITION AND tighten margins, financial services providers (FSPs) are using customer data to learn how they can better serve their customers. They often turn to segmenting their customer base to help them:

- **Build stronger and deeper customer relationships.** Better products and services increase customer satisfaction. Satisfaction drives loyalty, which in turn, reduces churn and increases wallet share.
- **Improve product uptake.** By understanding customers' behaviors and needs, companies can create products that are more suitable for them. They can choose channels that reach their customers and create messages that resonate with them.
- **Identify product opportunities.** Analyzing customers' behaviors, preferences, and needs can help companies identify either new product opportunities in the market or specific features that can be incorporated into existing products to better suit customers' preferences and motivations.
- **Develop flexibility and agility.** Understanding customers will help companies to adapt quickly to meet customers' rapidly changing needs.
- **Develop a data-driven decision-making organizational culture.**

Using data analytics to segment customers can be complicated. Because FSP project managers often are not experts in data analytics approaches and tools, they may not be able to effectively manage data analytics teams. Likewise, data analysts often are not experts in the financial sector, and they may not know how to address the unique segmentation challenges in the financial sector.

This guide can help you to create or review customer segments so that your organization can offer customers a tailored service in a standardized way. It offers practical guidance to project managers and data analytics specialists at financial services organizations, who are often responsible for driving segmentation strategies. It introduces commonly used, time-tested customer segmentation techniques, explains how these techniques can be applied to answer the business questions faced by FSPs, and gives step-by-step instructions for implementation.

BOX 1. **Digging deeper**

Implementation

For readers with some experience in data analytics (or those who want to dig into a little more technical detail), the Appendix includes simplified examples using the open-source software R. Throughout this guide, code that is relevant to the example in the Appendix are [bracketed] in red.

Advanced users

Readers who want more detail should review the list of additional resources provided in the references section that explain the origins and mathematical derivations of the techniques.

Note: See the appendix to this guide for relevant coding examples in the statistical software R. The R software can be downloaded free of charge from https://www.r-project.org/. The code used in the appendix is available as a text file and as an R file. The data used in the exercise are available at https://www.cgap.org/sites/default/files/research_documents/mfi_data.csv.

This guide will help you to:

- Segment customers into homogenous groups based on individual and household characteristics, economic activity, use of financial products and services, and other areas of interest that are sufficiently represented in data.
- Determine how to match products and services to the specific needs of different customer groups.
- Make a range of business decisions, including those regarding product design, risk management, and performance measurement and management.

Other financial services staff—from CEOs to marketing managers—may also find this guide useful. It is written for experienced business professionals and does not assume familiarity with basic data analytics concepts.

The first section of the guide covers the resources needed to conduct a data analytics project. The next two sections address analytical techniques that are used to identify customer segments and tailor product recommendations to individual customers based on their characteristics and past product and service use. The guide concludes with a case study of a large microfinance organization in India. The case study highlights the lessons learned and the challenges the organization faced.

SECTION 1

WHAT RESOURCES WILL YOU NEED?

EVERY DATA-DRIVEN SEGMENTATION project needs two key resources: (i) customer data and (ii) staff or consultants with the skillsets needed to analyze the data. The following is an introduction to different types of customer data and how they can be used.

1.1 Customer Data

The different types of customer data and data sources that can be used for segmentation include:

- Customer registration data (e.g., demographic data recorded for know-your-customer protocols).
- Attitudinal data from customer surveys.
- Historic data on the use of products and services, which are stored in core banking systems.
- Device information (e.g., mobile banking apps can provide access to handset information such as call detail records and GSM calls or SMS texts, which can be used to analyze behavior). A smartphone can provide even more information.

The methods presented in this guide can work with data sets of different formats and sizes. A general rule is that the more relevant the data available for analysis, the greater the likelihood of finding useful and practicable insights—and the more work required to prepare it for analysis.

1.2 Skilled Database Managers

Database managers use the database-management system (DBMS) to extract and prepare data for analysis. They are crucial to all subsequent steps and project success because they do the following:

- Describe how data are captured and stored in the database.
- Extract data in a form that is useful and understandable to the analyst.
- Investigate data irregularities or anomalies discovered in the analysis process.
- Preprocess or transform some of the raw data.[1]

Database managers may also be involved in statistical analysis of the data. The database manager's job is highly unlikely to end when the first data set is delivered; they should expect to be involved throughout the life of the project.

1.3 Data Analytics Expert

Many people working in business and financial services use spreadsheets or similar data analysis software to process and visualize data. While those skills and tools are adequate for some of the techniques discussed in the guide, other techniques involve more complicated and iterative statistics that require specialized statistical software.

At least one team member should have a background in statistics or data science (this may be a database manager who is also familiar with the business). This person should be a good communicator and be comfortable working with at least one

1 For some tasks, data preprocessing may be easier in a database language, like SQL (not covered in this guide), than in statistical software such as R.

statistical analysis software package. Equally important, the data analyst should have (or develop early on) a basic understanding of the business challenges the project seeks to address.

Organizations that see data analytics as a competitive advantage and intend to create a data analytics department should aim to have analysts who can use statistical analysis software and write code of their own to explore the data differently. These analysts should understand the business so that they can generate information that supports business decision-making processes.

1.4 Engagement of End-Users (Customer-Facing Staff)

Feedback from customer-facing representatives is integral to developing data-driven tools that will be useful to them (and used by them). This means that a portion of the business manager's time should be dedicated to the customer data analysis project.

While data analytics has the potential to uncover new insights about customers, it is mostly used to verify (or challenge) things that business managers already believe to be true. The data analytic results should be shared with business managers because business managers usually know and understand their customers best.

When managers echo what neutral or unbiased data show, chances are high that the data analysis is correct. Conversely, when data reveal trends or relationships that surprise or confuse managers, it is a good idea to revisit the data (including identifying and discussing some specific customers the managers know personally) to check the logic of the analytics.

SECTION **2**

PREPARING DATA FOR ANALYSIS

A CENTRAL AND OFTEN TIME-CONSUMING part of any data analysis is preparing the data to be analyzed. Flawed data will lead to flawed analysis. You should take the following steps to help get the data analysis off to a good start: understand the business context, understand where your data come from, and build the data set.

> ***"Data needs to be understood to be a 'sample of reality', recorded as measurements, stored as values and gathered for analysis or reference purpose."***
>
> ***—IFC (2017)***

2.1 Understand the Business Context

You need to understand how the data relate to the business question they will help to answer. You should study how data are collected and used in business operations. For example, if the goal is to design products and services that better meet customers' needs, the analyst might want to ask the following:

- Which customer profile(s) are considered relevant to each service?
- How are customers onboarded?
- What products and services are offered to first-time customers?
- How are products offered to repeat customers?
- When and why do customer data get updated?
- How is customer data used to inform existing products and manage customer lifecycle?
- How do front-line employees decide which products to offer each customer?
- Is there a customer lifecycle?
- What are the main reasons for customer dormancy or drop-out?

The analytical methods described in this guide require the analyst to make decisions that will materially impact the results of the analysis. While these decisions and choices should be vetted with senior managers and business end-users, data analysts should "learn the business" at the earliest stage of preparing the data rather than applying math techniques mechanically to whatever data they are given without understanding their relevance.

2.2 Understand Where Your Data Comes from

Your organization collects a lot of data, and it is useful to know where each piece of data comes from, how it is collected, and when it is captured (Caire et al. 2017, p. 4). Table 1 ranks types of data commonly collected by their expected reliability for predictive analytics.

TABLE 1. **Reliability of different types of data collected by businesses**

Type of Data	Reliability	
Transactional	HIGH	If stored consistently, transactional data (e.g., account deposits and withdrawals, loan payments, bill payments) are usually very reliable. These data present a historic and objective record of a customer's actual behavior or economic activity.
Documentary	HIGH	Identity and basic demographic data are often taken from (or verified by copies of) official documents (e.g., national ID cards).
Collected from devices	HIGH	Device data can be as reliable as financial transaction data.
Psychometric	ABOVE AVERAGE	There are different techniques in psychometric tests to validate candidates' answers (e.g., using slightly similar questions in different sections of the test). However, data reliability is sensitive to the quality of the tests and how they are administered.
Collected by staff	AVERAGE	These data may be influenced by the judgment, work style, or experience of the person collecting it.
Self-reported	BELOW AVERAGE	Data reported by a customer (e.g., on an application or survey) can be less accurate because people have different styles of sharing information. Customers may tailor responses for different purposes.

2.3 Build the Data Set

The following are some questions to ask when you are putting together the data set for analysis:

1. **How far back in time should the analysis go?** While the answer depends on the purpose of the analysis, here are some general guidelines:
   a. **The more recent past is a better indicator of the near-term future.** This is particularly true when countries change or grow rapidly. On the other hand, if it is important to understand how things evolve over time or how behavior changes over an economic cycle, a longer period may be preferable.
   b. **Data quality and availability may change over time.** The content and quality of a database may change (e.g., with changes to core software systems, the roll-out of new products, or changes in the operations that collect data). Look beyond the label of a data column to understand what data it contains over time—descriptive statistics, described below, can help with this. Sometimes the most appropriate period may be limited by when certain system or organizational changes were made.
2. **Which data fields are relevant?** Focus first on what is known or thought to be relevant. Data mining or looking at large data sets without a hypothesis on expected relationships may lead to spurious or chance results and models that may fit too closely to the particular data set and not work well with new data. Another consideration is whether selected data fields can be collected consistently in the future.
3. **How large of a sample does the analysis require?** The most accessible option is to make all relevant data available to the analyst. Then the data can be shaped and sampled as needed, including testing predictive models on cases not used to build the models. However, where there are millions of customer records or where representative data need to be collected specifically for the analysis (at some cost to the organization), random samples of the population are likely to generate representative results (albeit with some degree of error). As a general rule, the more of the population in the sample, the more representative it will be. For example, suppose an organization has 10,000 customers and wants to know what share of them have used insurance products in the past. Unfortunately, the organization has not already collected data on product use. How many customers does it need to survey now to get a representative estimate of past use of insurance products? A sample of 500 customers (for example) will give a more reliable picture than a sample of 50 customers. See Box 2 for an example of trade-offs between sample size and accuracy using a free tool from the Australian National Statistics Service.

BOX 2. **Sample size calculation**

Using a sample size calculator like the one from Australia's National Statistical Service can help you to quickly determine how many customers you may need to survey to generate representative results.[a]

In an approach that relies on an estimate of the share of customers who have used insurance products that is 95 percent accurate with a margin of error of +/- 5 percentage points, you could sample only 356 of the 10,000 customers (see Sample 1).[b] If 40 percent of customers in the sample used outside insurance products, you could be pretty sure (or correct 95 out of 100 times) that between 35 and 45 percent of the rest of your 9,644 customers also have used insurance products.

If you need greater accuracy, say to be 95 percent sure that your sample will have a margin of error of +/- 1 percentage point, you need to sample 4,797—nearly half the population (and likely at a significantly higher cost). If in that much larger sample you find that 40 percent of customers also use outside insurance products, you could be pretty sure (or correct 95 out of 100 times) that between 39 and 41 percent of the rest of your 5,203 customers also have used insurance products.

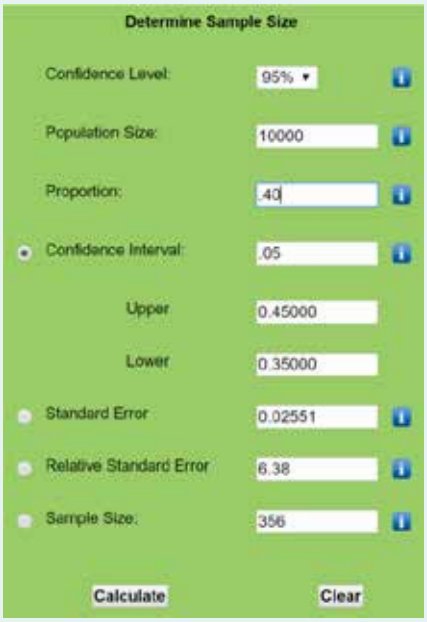
Although a calculator (or calculation) may provide a sense of surety that a sample is "large enough" to be representative, in practice, determining the sample size should be less about achieving an arbitrary level of statistical significance and more about having a representative sample from a well-defined population that is relevant for issues that matter to the organization.[c]

a. See http://www.nss.gov.au/nss/home.nsf/pages/Sample+-size+calculator. The site provides definitions of each term and details about how to use the calculator.

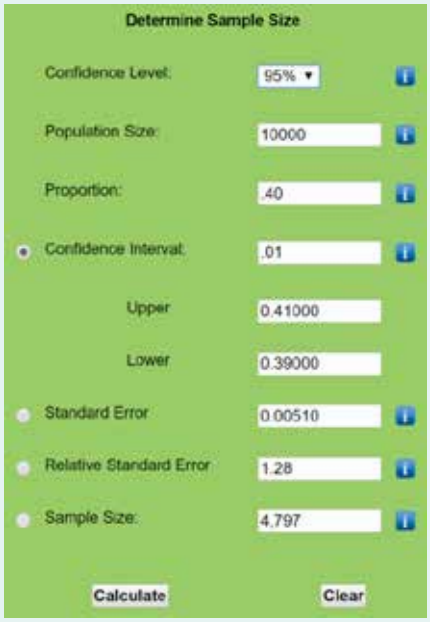
b. For the calculator used in this example, the population proportion defaults to 50 percent if you do not enter a specific number. In this case, 40 percent was entered to illustrate the example.

c. For more on sample sizes for estimating poverty likelihood, see Schreiner (2018, p. 29).

Sample 1: Accurate Enough?



Sample 2: More Accurate



For predictive analytics, you should randomly sample some of the data to look for and model relationships, and then test them to determine whether those same relationships are found in other random samples of comparable sizes—this is sometimes referred to as "out-of-sample" testing. For example, a financial institution with 100,000 customers might randomly sample 10,000 of them to analyze and understand data-driven relationships. Applying the conclusions of (or repeating) this analysis to another random sample of 10,000 customers (of the 90,000 not used in the first analysis) will either confirm that the same relationships are found in the rest of the population or show that some things that looked meaningful in the first sample are not useful because they do not also appear in the second sample.

2.4 Check Data for Accuracy

If the data are at odds with what you know about the company, its customers, and services, then there are probably errors in the underlying data themselves or in the analyst's understanding of the data definitions (i.e., the real meaning of the data in a field or column).

To get a quick basic feel for the data, review the descriptive statistics—sometimes called summary statistics (Caire 2017). These include averages, summations, counts, and aggregations, such as maximum and minimum values. See Table 2 for a list of some popular descriptive statistics.

Statistical software [A-1] makes it quick (mere seconds) and easy to generate descriptive statistics. Descriptive statistics can help the analyst spot potential issues with specific data fields before doing more analysis. For example, if a financial institution knows most of its customers have deposits around the minimum account opening requirement of $200, but a data field named "deposits" has a mode (most frequent) value of $1,000, there is probably an error, either in the data themselves or in how the analyst understands the true content of that column.

Two common issues that seriously affect many types of mathematical analysis are outliers and missing values.

OUTLIERS

Outliers are extreme and unrealistic values in the data. They are often the result of errant data entry or capture and can seriously distort the results of many types of statistical analysis. Even if outliers are not errors, you will want to avoid having the results of your analysis disproportionately influenced by a few, unusual (albeit real) cases.

A simple example of how outliers can affect results and how they may or may not be easily spotted without descriptive statistics is illustrated in Table 3. It shows data on customer age for three different random samples of 10 customers each. The average age statistics for samples A and B are plausible, even though row 10 in Sample A (age 136) is obviously an error. By contrast, the average age for sample C suggests an error that can be traced to row 5—age 422!

It is usually good practice to remove extreme outliers from the data. If you have an abundance of data and relatively few outliers, the simplest approach is to drop the cases

TABLE 2. **Common descriptive statistics**

Name	Used to understand ...
Type	What type of data to expect (numeric or text)
Number of unique values	How the population varies
Number of missing values	How well the data have been collected
Mean, median, and mode	What the average customer looks like
Histograms	How the population is distributed
Five highest and lowest values	Outliers or values that are likely to be errors

TABLE 3. **Example of outliers in average age of customers**

	Sample		
	A	B	C
Ref		Age	
1	51	59	22
2	22	57	51
3	34	38	45
4	44	40	45
5	56	58	422
6	18	46	48
7	38	64	57
8	46	38	23
9	25	27	27
10	136	36	46
Average Age	47	46.3	78.6

BOX 3. **Interquartile range rule example**[a]

To identify outliers using IQR:

1. Sort the data from lowest to highest value.
2. Identify the first quartile threshold, Q1, as the value one-quarter of the way through the sorted data column (i.e., 25 percent of values are less than this number and 75 percent of the values are greater than it).
3. Identify the third quartile threshold, Q3, as the value three-quarters through the sorted data column.
4. Calculate the IQR, which is IQR = Q3 – Q1.
5. Multiply the IQR by 1.5 (rule of thumb, or experiment with other values).
6. Subtract IQR x 1.5 from Q1. Any value below this is an outlier.
7. Add IQR x 1.5 to Q3. Any value above this is an outlier.

Numeric example of applying IQR:

For example, if your data column contains 12 numbers, the third value will be one-quarter of the way through the distribution (3/12 = 25%), and the ninth will be three-quarters through the distribution (9/12 = 75%).

In this case: Q1 = 14, Q3 = 22, and IQR = 22 – 14 = 8.

Multiply IQR by 1.5 (8 x 1.5 = 12). Subtract 12 from Q1 (14) and add 12 to Q3 (22). Use the rule to determine that any value less than 2 or greater than 34 is an outlier in the data column. This shows that the last value, 102, is an outlier and the entire row of data should be removed or a different value, such as the average or the upper limit (19), should replace 102 if the row is to be retained.

a. Adapted from Taylor (2018).

Data	Quartile
4	Q1
8	
14	
14	Q2
16	
17	
17	Q3
19	
22	
25	Q4
27	
102	

with outliers. If data are limited, other strategies, such as replacing an outlier with the mean (average) value or with a capped value (e.g., all values over 90 are recoded to be 90), can be used.[2]

One simple way to systematically identify and handle outliers in a large data set uses the Interquartile Range Rule (IQR) rule (see Box 3) [A-2]. This sets boundaries for the highest and lowest values expected in your data set for given data, and values falling outside of this are considered outliers.

MISSING DATA

Data sets usually have missing values. When a nontrivial share of data points is missing for any given data field, it is important to understand why.[3] Possible reasons include:

- The field is not applicable to the customer.
- The information is not available because it was not requested from the customer or the customer did not have or provide it.
- The field was not collected before a specific date (and later was).

2 See "Advanced Handling of Missing Data," Quick R reference library, accessed 15 December 2017, https://www.statmethods.net/input/missingdata.html.

3 A "nontrivial" share is more than 5 percent of the data points in a data field.

Depending on why data fields are missing, some may not be suitable for analysis or should be processed using a technique for working with missing variables. Some of these techniques are presented in Table 4.[4]

2.5 Scale the Numeric Variables

The data-driven segmentation techniques discussed in Section 3 measure distances between data points to identify important individual features and different customer groups in the data. These methods require that numeric data be expressed in common or standardized units so that variables that have large values don't dominate the results. Otherwise, for example, differences in loan values, which may range from $500 to $5,000 will appear much more important than differences in household size, which may range from 1 to 15.

A column of variables is scaled by subtracting the mean, or average value, for the column from each value and dividing it by the standard deviation for the column. An example of 10 rows of standardized values for household size (hh_size) and loan size (loan_size) is shown in Table 5. For Customer 1, the standardized value of household size is calculated as (7 – 5.2) / 2.20 = 0.82.

The standardized columns hh_size_std and loan_size_std are expressed in common units with a mean of 0 and a standard deviation of 1. Statistical software can calculate this quickly and consistently for all numeric data columns [A-3].

4 See Quantitative Finance Collector (2010).

TABLE 4. **Working with missing values**

Strategy	Use	Pros	Cons
Remove rows	• Data are plentiful • Missing records are relatively few • Complete data will always be required in future	• Uses complete records • No assumptions	Loss of data means sample is less representative
Replace with unique value (such as -1 or 999999999)	• Data are not numeric (or numeric data have been put into groups) • High percentage of missing values across different columns • Missing values may be possible for same columns in future	• Preserves data • Can accommodate missing data in future • Controls for possible nonrandom occurrence of missing data	May introduce bias if past reasons for missing data will not persist
Replace with an average value	• Data are limited • Missing values are relatively few	• Preserves data	Assumes past missing cases were no different than average cases
Replace with predicted value	• Data are limited • Missing values are relatively few	• Preserves data • May be more realistic/accurate than using average values	Increases complexity of model

TABLE 5. **Example of scaling numeric values**

Customer	Hh_size	hh_size_std	loan_size	loan_size_std
Customer 1	7	0.82	$2,400	-0.40
Customer 2	5	-0.09	$3,600	0.50
Customer 3	4	-0.55	$4,600	1.25
Customer 4	5	-0.09	$1,800	-0.85
Customer 5	5	-0.09	$800	-1.60
Customer 6	4	-0.55	$2,000	-0.70
Customer 7	2	-1.45	$4,500	1.18
Customer 8	3	-1.00	$2,000	-0.70
Customer 9	8	1.27	$3,100	0.13
Customer 10	9	1.73	$4,500	1.18
TOTAL	52	0.00	$29,300	0.00
MEAN	5.2	0.00	$2,930	0.00
Standard Deviation	2.20	1.00	$1,334	1.00

SECTION 3

DATA-DRIVEN CUSTOMER SEGMENTATION

YOU CAN USE CUSTOMER SEGMENTATION to group customers based on common characteristics, behaviors, attitudes, needs, or interests. Segmentation will help you to serve each group with different products, services, or promotional strategies that are appropriate for each group.[5]

Segmentation can be rules-based, data-driven, or a combination of both. Table 6 outlines some of the strengths and weaknesses of each approach.

Section 3.1. presents four steps to implementing clustering techniques for customer segmentation. Clustering algorithms analyze customer data to identify groups of customers that are most like each other and that are most different to the customers in other groups. These data-driven insights can augment or complement rules-based segmentation and learnings from traditional marketing research methods, such as focus groups and surveys.

3.1 Clustering Analysis

STEP 1: SELECT POTENTIAL SEGMENTATION VARIABLES

Clustering results depends on the data in the algorithms. Financial institutions may collect a wealth of data about customers, but not all of it will be relevant or appropriate for clustering.

"Segmentation, at its most basic, is the separation of a group of customers with different needs and characteristics into subgroups of customers with similar needs and characteristics."

—GAVETT (2014)

Example data set

The analytical techniques described in this section use an example of data that might be collected by a financial institution.[6] It includes household, product use, and attitudinal information. The data table (available at https://www.cgap.org/sites/default/files/research_documents/mfi_data.csv) contains 200 rows of data and 11 columns, which are described in the data dictionary in Table 7. You can download the data set if you want to try the R coding examples in the appendix.

This data set is purposefully kept small to provide simple examples of how to read and interpret the output of the techniques covered in this guide. It is not representative of specific relationships found in a specific microfinance institution (MFI), and real data sets typically contain many more data columns, require many more analytical choices, and present more computational challenges.

5 For more information, see CGAP (2016).

6 For a detailed explanation of the mathematics of each technique, information on other methods (also with coding examples in R software), and practice data sets, see James et al. (2013).

TABLE 6. **Comparing rules-based and data-driven segmentation**

Segmentation Type	Strengths	Weaknesses
Rules-based • Usually in 1 or 2 dimensions • Based on existing beliefs	• Easy to perform • Choice of rules guided by strategy • Makes sense to managers	• Sometimes there is little learning from the data • More customer-centric approaches are needed for a deeper understanding
Data-driven • Use machine learning to identify clusters of similar customers • Can consider many dimensions (or customer attributes) at once	• Can identify new groups on more than two dimensions • Can provide new data-based insights	• Requires specialized skills • May cost more and take longer • Sensitive to spurious data

TABLE 7. **Example data set dictionary**

Field Name	Description	Type
hh_id	Unique household identity number	Numeric (used as label)
hh_size	Number of household members	Numeric
hh_monthly_income	Household income per month	Numeric
years_mfi	Years working with MFI	Numeric
max_loan	Largest past loan taken	Numeric
satisfaction_survey	Customer satisfaction survey score	Numeric
Assets	Value of household assets	Numeric
occupation_type	Trade related or not	Text (Categorical)
health_insurance	Health insurance policy (yes/no)	Text (Categorical)
Location	Name of region of residence	Text (Categorical)
rent_own	Own or rent property of residence	Text (Categorical)
has_deposit	Has a balance of > 0 on deposit account	Numeric (0 = No Deposit, 1 = Has Deposit)

Column header names: Column header names should be short and simple, yet as descriptive as possible. In syntax-based software programs like R, the column names may be used repeatedly. If they are long, the code will be harder to read and work with. They also should be descriptive enough for the analyst to remember what data are in each column.

Before doing the clustering, ask experts questions you may have about correlation analysis and principal component analysis, for example, to select the variables to explore.

Expert judgment. Good variables for clustering-based segmentation typically are:

- Available for most customers (or where missing values have a specific meaning, e.g., when a missing value in the field "collateral" always means that there is no collateral).
- Collected consistently across the organization and over time.
- Well understood by management and users.

Machine-learning methods use math to identify combinations of factors to include in models, but they do not consider qualitative information, such as the cost, reliability, and the practicality of using various types of information.

Correlation analysis. If many columns of numeric data measure things that are related to one another—things that are correlated—you may want to generate a correlation matrix as a first step in selecting the variables to be used

in clustering. The Pearson Correlation Coefficient ranges from -1 to 1 and measures the degree and direction of the relationship between two variables:[7]

- Higher negative values (closer to -1) indicate that two variables move in opposite directions—as one value rises the other falls (or vice versa).
- Higher positive values (closer to 1) indicate that two variables move together—as one value rises (or falls) the other also rises (or falls).
- Values closer to zero (positive or negative) indicate that the two variables contain different information and are basically independent of one another.

For example, time-based measures of stability such as "years in job", "years in residence", "years of professional experience", and "time as customer" are generally correlated with one another as well as with customer age. The same is often true for various measures of income, expenses, or assets.

Table 8 is a correlation matrix [A-4] for the six numeric data columns of the example MFI data. The diagonal of the matrix (with the values of 1) is the intersection of each column with itself. (For clarity, only values below the diagonal are presented.) Every other correlation coefficient is found at the intersection of the two columns (e.g., household size [hh_size] and household monthly income [hh_monthly_income] have a positive correlation of 0.80).

If there are many numeric variables, say 20 or 30, it might make sense to drop one or more of the strongly correlated (red) numeric data fields from further exploratory analysis. Too much of the same information adds little value to the analysis and can obscure from view other potentially useful, but less powerful, relationships in the data. The choice for which variables to keep and which to drop in exploratory analysis is subjective and should be based on how given data fields are collected and used in the organization rather than strictly on quantitative rules or algorithms.

Too much of the same information adds little value to the analysis and can obscure from view other potentially useful, but less powerful, relationships in the data.

Given that there are only six numeric fields in this example, all variables will be kept in each stage of analysis.

Principal component analysis. Principal component analysis (PCA) is widely used in exploratory data analysis. It is particularly useful with larger data sets—those with many features or variables—to identify a smaller number of features that contain most of the information of the full data set. This is sometimes called feature reduction, and it can be a useful first step in clustering analysis to identify which features in a data set best describe differences in the individual rows (in this case, customers) in the data set. See Box 4 for guidelines on clustering methods for customer segmentation.

For example, say we have 50 or more possibly relevant numeric variables in the data. In the most common case, where many of those variables will be at least somewhat correlated with one another, PCA can help to identify which of those best explain the differences (sometimes called

TABLE 8. **Correlation matrix example**

	hh_size	hh_monthly_income	years_mfi	max_loan	satisfaction_survey	assets
hh_size	1.00	0.80	0.09	0.57	0.05	-0.04
hh_monthly_income	0.80	1.00	0.25	0.67	0.06	-0.03
years_mfi	0.09	0.25	1.00	0.41	0.00	0.01
max_loan	0.57	0.67	0.41	1.00	0.03	-0.02
satisfaction_survey	0.05	0.06	0.00	0.03	1.00	0.02
Assets	-0.04	-0.03	0.01	-0.02	0.02	1.00

7 See James et al. (2013, p. 70) for the correlation coefficient formula.

BOX 4. **Unsupervised learning and rules of thumb**

This guide presents some simple guidelines for data-driven customer segmentation for practitioners. Note that both PCA and the clustering methods described next fall under "unsupervised learning." If you generally call your data fields "inputs" and any one particular field you would like to predict in a learning exercise an "output," then:

- **Supervised learning** estimates an output based on one or more inputs
- **Unsupervised learning** looks at the structure and relationships of inputs, but with no output variable[a]

A challenge of working with unsupervised learning methods (like clustering and PCA) is that there are no universally or globally "right" answers. This means all (or any) answers can potentially be right or at least subject to debate and discussion.

The more decisions are left to judgment and discretion, the more challenging it is to create guidelines that can be applied to many possible situations and data sets. A general guideline for unsupervised learning is to check results with those who know the business well. Since unsupervised models are derived from samples of reality (data), they should make sense to those in the business.

a. See James et al. (2013, p. 1).

variation) in the data. These are most often the variables included in the first (particularly) or first few components. As James et al. (2013, p. 384) note:

"In practice, we tend to look at the first few principal components to find interesting patterns in the data. If no interesting patterns are found in the first few principal components, then further principal components are unlikely to be of interest."

For traditional PCA, the variables should be numeric and scaled, and the outliers should be removed. If you put only the six numeric variables from the example MFI data set into PCA software [A-5], you will get a list of principal components and the percentage of variance explained by each component, as shown in Table 9.

TABLE 9. **Example of variance explained by principal component**

Component	Variance Explained	Cumulative Variance Explained
PC1	42%	42%
PC2	17%	59%
PC3	17%	76%
PC4	16%	91%
PC5	6%	97%
PC6	3%	100%

TABLE 10. **Loadings on the first three principal components**

	PC1	PC2	PC3
hh_size	0.54	-0.25	0.06
hh_monthly_income	0.58	-0.11	0.04
years_mfi	0.28	0.58	-0.17
max_loan	0.54	0.13	-0.06
satisfaction_survey	0.05	-0.32	0.81
Assets	-0.03	0.68	0.56

To move this example further, assume that you decide to keep and examine the first three principal components, which explain 76 percent of the variation in the data. At this point, it is common to look at which data fields most influence each component. Table 10 shows the "loadings," or influence, of each of the six data columns on the principal components.[8] Higher numbers indicate greater importance in describing the component.

In Table 10, PC1, which indicates the largest share of variance (42 percent as per Table 9), is described by household size, monthly income, and largest past loan—the largest loadings are in red. Assets and years as a customer contribute most to PC2, and the customer satisfaction survey is most important to PC3.

PCA can be used in clustering to understand which features in the data are most important in describing it with the goal of reducing the number of features (columns) in further analysis. In this example, it is clearly of limited value because there are only six numeric variables, all of which will be used in the examples that follow.

8 Calculations of loadings requires matrix algebra. The can be found in many statistical texts, for example in Jolliffe (1986).

STEP 2: EXPLORE CLUSTERS

Clustering is a mathematical technique for finding distinct and relatively homogenous groups in a data set. One of the best-known approaches is K-means clustering [A-6]. James et al. (2013) describe K-means as a simple and elegant approach for partitioning a data set into K distinct, nonoverlapping clusters. Like PCA, K-means calculates distances to determine clusters, hence, it requires numeric, scaled variables as inputs. Other clustering methods handle a mix of numeric and categorical variables and are presented in the appendix [A-7].

To perform K-means clustering, you must first specify the number of clusters desired. Then, the K-means algorithm will assign each observation to one of the K clusters (James et al. 2013, p. 400). There are several mathematical methods for determining the optimal number of clusters (see the appendix) [A-8]. However, for practitioners, it may be more expedient to explore cluster sizes that are between three and seven on a trial-and-error basis, because many business use cases benefit from a manageably low number of clusters or segments.

Entering the six quantitative data fields from the MFI data set into a K-means clustering algorithm looking for three, four, or five clusters results in the three graphs in Figure 1. Each of these solutions shows relatively distinct clusters, albeit with some overlap—of clusters 1 and 2 in the three-factor solution; 1, 2, and 3 in the four-cluster solution; and clusters 2, 4, and 5 and 1 and 3 in the five-factor solution.

Table 11 illustrates another way of looking at the potential usefulness of the three clustering solutions in terms of how well customers are spread among the number of clusters. With no rules to say what is right, some possible interpretations are:

Start Small. The example data set has only 200 rows and its variables happen to work well for clustering. Real data (particularly bigger data with many more rows and columns) will take longer to process and will likely lead to results that are more difficult to interpret. Try starting small:

- If you have 100,000 rows of data, test your clustering approaches with 1,000 or fewer. Processing times can be considerable for clustering algorithms with large data sets.
- Select a few priority columns of the data you expect to be most important for separating your customers (based on expert judgment, correlation, and/or PCA). Add and remove subsequent columns on a trial-and-error basis. Doing this and reviewing other descriptive statistics can help you understand how different types of data are influencing the results.

- The three-cluster solution is the simplest; however, its usefulness may be limited because Cluster 1 is quite small (20 [of 200] customers).
- The four-cluster solution has more balance of customers across clusters.
- The five-cluster solution has the most even distribution of customers across clusters.

To investigate further which solution might be best, you need to profile the identified clusters, see how they differ, and consider the implications through a business strategy lens.

FIGURE 1. **K-means three-, four-, and five-cluster solutions**

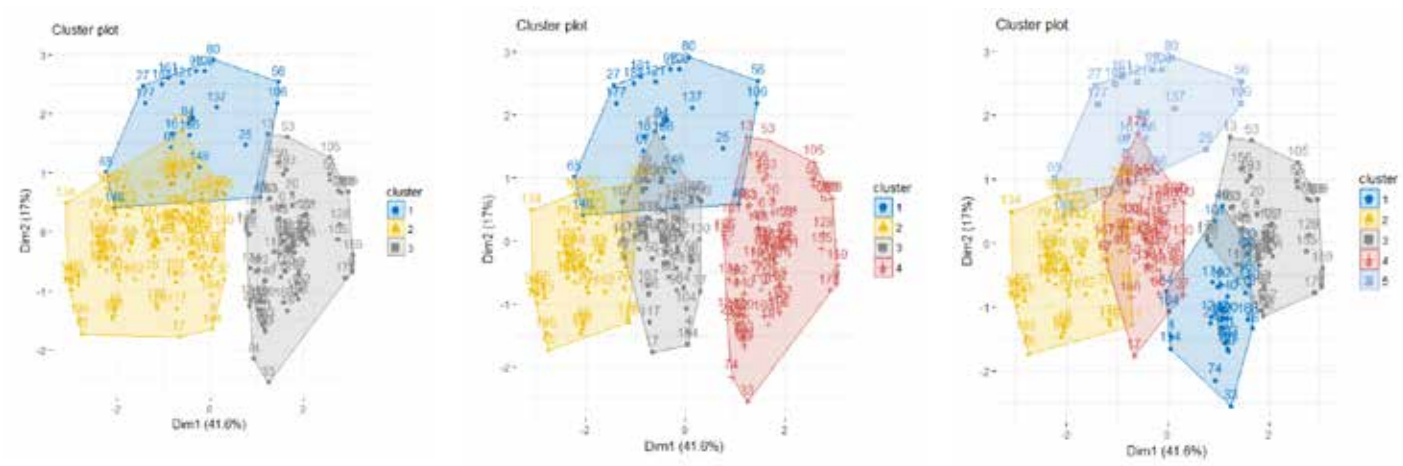


TABLE 11. **Cluster sizes for three-, four-, and five-cluster solutions**

No. of Customers					
	Cluster				
	1	2	3	4	5
Cluster of 3	20	104	76		
Cluster of 4	20	45	59	76	
Cluster of 5	32	48	49	52	19

STEP 3: PROFILE THE CLUSTERS

Which (if any) of the three clustering solutions is preferable for a business application should be determined not by any charts or statistics, but on the number of clusters that:

- Are obviously (meaningfully) different, in a business sense.
- Are actionable in terms of devising business strategies to address the clusters.

The summary statistics for each of the clusters give a snapshot of the average customer in each cluster. Table 12 shows the average values for characteristics for each cluster solution.

For the three-cluster solution, you can see the following:

- **Cluster 1**—a small share (10 percent) of customers have relatively large households and higher monthly income and have had larger loans.
- **Cluster 2**—a larger share (52 percent) of customers are from small households with small income and have had smaller loans.
- **Cluster 3**—other customers (38 percent) tend to have medium household sizes and incomes and past loans in between the highs and lows in clusters one and two. They also have the highest assets.

Table 12 shows that, with three clusters, there is little difference in customers' years with the MFI, customer satisfaction survey scores, or household assets.

Table 13 shows the same statistical breakdown for a five-cluster solution.

With five clusters, the larger groups from the three-cluster solution are further divided to provide additional differentiation in the small to medium (1.6 to 3) household size bands. Cluster 5 is now unique for identifying a medium-sized household with much larger assets. Whether this difference is important depends on the products and services the MFI plans to offer.

Although it is common in marketing to use a selection of "most important" numeric data fields for clustering, the remaining data fields can be used to better understand and profile the clusters. Table 14 further explores the three-cluster solution in terms of the non-numeric data fields in the MFI data set. The three clusters are labelled large, medium, and small households to reflect the most clearly distinguishing characteristics from the analysis.

TABLE 12. **Average customer statistics for three-cluster solution**

	Share	hh_size	hh_monthly _income	years_mfi	max_loan	satisfaction _survey	Assets
Cluster 1	10%	6.2	1,180	4.2	2,060	33.6	10,791
Cluster 2	52%	1.5	372	3.2	826	32.5	9,948
Cluster 3	38%	3.1	649	4.4	1,307	33.5	12,766

TABLE 13. **Average customer statistics for five-cluster solution**

	Share	hh_size	hh_monthly _income	years_mfi	max_loan	satisfaction _survey	Assets
Cluster 1	16%	7.2	1,141	3.3	1,514	33.9	10,595
Cluster 2	24%	1.6	375	2.2	850	31.0	9,443
Cluster 3	25%	5.7	1,224	4.6	2,335	33.8	9,852
Cluster 4	26%	3.0	642	4.5	1,309	34.9	8,717
Cluster 5	10%	3.1	636	4.2	1,482	32.1	27,368

TABLE 14. **Profiles on non-numeric indicators for three-cluster solution**

		Cluster		
		1	2	3
		Large	Small	Medium
Occupation	Trade	55%	55%	43%
	Non_trade	45%	45%	57%
Health Care	Health Care Yes	90%	69%	75%
	Health Care No/Missing	10%	31%	25%
Location	Countryside	20%	19%	21%
	Semi-Urban	30%	35%	36%
	Urban	50%	39%	38%
	Missing/Other	0%	7%	5%
Property Ownership	Rent	20%	34%	26%
	Own	80%	63%	74%
	Missing/Other	0%	4%	0%

In this example, the differences across clusters for the non-numeric information are relatively minor. Some differences that stand out include the following:

- The small household cluster is more likely to own property.
- The small household cluster is more likely to have health care coverage.
- The medium household cluster has more nontrade businesses.
- The large household cluster is more likely to be in an urban location.

This example is arguably more realistic than many textbook examples that present obvious solutions. While data-based clustering may lead to some revelations about customers and their behaviors, it is just as likely that there will be few new insights and that learnings will be subtle and nuanced, requiring a deeper dive into the data and analysis.

The next section discusses how additional qualitative research can help you better understand and personify segments in ways that bring to life the type of customer in each of the clusters.

STEP 4: CREATE PERSONAS

Creating personas can help you design marketing campaigns and help your customer-facing staff understand their customers better.

> "Creating persona profiles, rather than relying on tables and charts, for example, helps your audience digest the most important segment characteristics. Also, prepare examples of the types of things these customers might say if you spoke with them. Actual and illustrative quotes bring segments to life" (CGAP Customer Segmentation Toolkit).

In practice, cluster personas are best created with business managers who understand the business needs of the company (see Box 5). Sharing data insights with customer-facing staff and listening to their feedback will help to create personas that everyone believes in—transparency and buy-in are two key drivers of success in implementing data-driven business tools.

3.2 Keeping It Real

A clustering algorithm will always find clusters, whether they are valid or not. Therefore, it is important to validate the real-world plausibility and usefulness of clusters.

In the Dvara Trust example, focus groups and field tests were used to solicit qualitative feedback and build personas from the data-driven clusters. While mathematical calculations and data helped to identify the drivers of similarities and differences across customers, the perspectives of frontline staff and their contributions improved the chances of successfully implementing data-driven customer segmentation tools.

BOX 5. **Persona creation in Dvara Trust**

Dvara Trust (formerly IFMR Trust) operates a group of companies that provide access to financial services in remote, rural India through a branch-based model called Kshetriya Gramin Financial Services (KGFS).

In partnership with CGAP, Dvara sought to apply data-driven approaches to:

- Create customer segments
- Track the impact of KGFS branch products and services over the past decade
- Arrive at more relevant product recommendations to customers by the frontline staff

The data-driven segmentation resulted in four occupation-based segments that were further tiered according to starting per capita household income (high, medium, and low ranges).[a]

To bring these segments to life, the project team worked with KGFS wealth managers to turn the quantitatively defined segments into personas—simplified descriptive summaries of customers from each segment. These summaries include details of the customer's situation, context, needs, and motivations.

First, more extensive segment profiles were built based on household demographics, asset ownership, expenses (including education, health, and festival expenses), and KGFS product use over time. The following is an example of a resulting profile:

Age: 32 years
Occupation: Works abroad (Singapore)
Education: Class 5
Family: Mother (60 years), brother (25 years), wife (28 years)
Household income: Rs. 500,000 per annum
Household expenses: Rs. 100,000

- **Brother:** agrilabor + 100 days work
- **Wife:** agrilabor + 100 days work

Remittances: Rs. 25000 per month
Channels: Bank account, Western Union, informal

KGFS wealth managers and area managers participated in focus groups where they were given information about the different segments and asked to think of (and name) people they had worked with who matched the characteristics of the segment.

Naming the persona is an important step in humanizing the segment.

As wealth managers discussed customer income and expenses in detail, they mentally mapped the patterns to households and businesses from their own experience. This allowed managers to add detail and color to the segments in terms of their perception of these people's needs, aspirations, worries, and wants.

Finally, they were asked to combine all characteristics discussed into a persona bio, including quotes that capture the key drivers for the persona's behavior. See Figure B5-1.

To test the usefulness of the personas, the research team met with randomly selected customers and asked them about household demographics. They compared their answers with their latest financial information from the KGFS system, which helped to show how accurate the data-driven clustering-based segments were.

FIGURE B5-1. **Example of a persona bio**

a. Total household income divided by the number of people in the household.

SECTION 4

CUSTOMER-CENTERED PRODUCT OFFERS

THERE ARE MANY WAYS TO USE data to determine which products and services to offer customers in specific segments (or even without formal segmentation). Which methods work best depends on the intended business use case, what information was captured and stored in the past, and what new data can be captured (e.g., via customer survey) now.

The next sections present analytical techniques that may help you make decisions on customer-centric products. We start with basic summary statistics, move to well-known, more complex tools of regression and classification trees, and finally to an introduction to recent and complex machine-learning network techniques.

4.1 Basic Descriptive and Summary Statistics

While descriptive statistics do not grab headlines, they are the workhorse of business intelligence units around the world. According to IFC's Data Analytics Handbook (Caire et al. 2017):

> "The least complex analytical methodologies are descriptive in nature; they provide historical descriptions of the institutional performance, analysis around reasons for this performance and information on the current institutional performance. Techniques include alerts, querying, searches, reporting, visualization, dashboards, tables, charts, narratives, correlations, as well as simple statistical analysis."

A lot can be learned from basic summary statistics using any statistical software, including popular products like Microsoft Excel. These programs can provide summary statistics and reports, visualization with charts and graphs, and in the case of Excel, powerful pivot table tools.

Pulling together information generated from these simple tools is popularly referred to as a customer-level dashboard. See Figure 2 for an example of a customer-centric product recommendation dashboard.

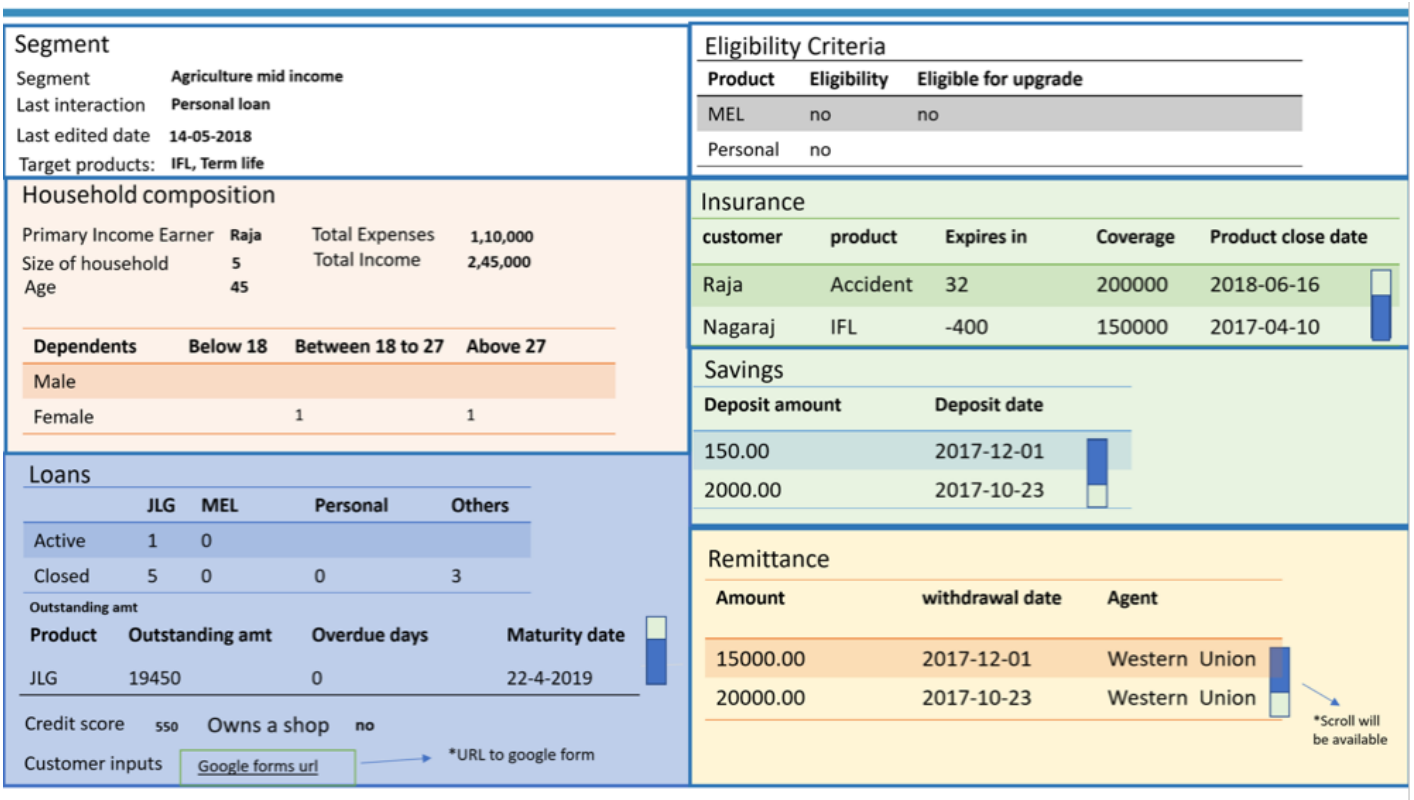
Dashboards for customer-centric product recommendations may present a variety of information, including the following:

- Customer's segment/persona
- Products and services currently used by the customer
- Customer's most recent (date sorted) activity
- Comparison of customer key performance indicators to benchmarks: average for all customers; average within segment
- Eligibility and/or likely suitability for other services based on a customer's segment, product eligibility criteria, and predictive models built on other characteristics in the data

Product recommendations can be customer-centric even if they are based on nothing more complex than a few business rules related to segments and/or other selected customer characteristics. In a model, this would be programmed as a series of logical arguments, for example:

If customer is in Cluster A
AND
If customer has had a deposit account for > 1 year
AND
If customer is self-employed
THEN
Offer new fixed-interest term-deposit product

FIGURE 2. **Example of customer-centric product recommendation dashboard**



If several rule-based recommendations are configured in software (such as Excel), a dashboard can automatically list the appropriate products to recommend to each customer.

A rules-based approach to product recommendations can be used to promote certain strategies—for example, to sell certain new products or to expand a certain segment of the business. However, unlike the data-driven methods discussed in the next section, these approaches tend to rely on existing beliefs rather than new information from data.

Simple techniques, such as summary statistics, can be adequate and appropriate when:

- Working with the most obvious associations in data—those truths about the business that are already known with certainty.
- There is too little customer data available for rigorous analysis.
- Customers are homogenous (outside of a few obvious indicators) and/or few products or services have been offered in the past.

Under these conditions, the analysis of summary statistics is likely to lead to the same conclusions as would more complex methods. For example, if you know from experience that the need for a certain credit product depends heavily on type of employment, then you are likely to find that relationship in the data whether you look at simple summary statistics—such as a pivot table of credit product take-up versus occupation type—or use any of the more complex techniques discussed in the next section. When relatively simple tools can get the job done, the understanding and acceptance of the tools by frontline staff can improve the likelihood that they will be successfully implemented improves the likelihood that these tools will be successfully implemented.

When relatively simple tools can get the job done, "frontline" user understanding and acceptance of those tools can improve the likelihood that they will be successfully implemented.

4.2 Predictive Analytics

Predictive analytics go beyond descriptive statistics to look for patterns in data that can be used to predict an outcome of interest,[9] including, for example, the likelihood a customer will purchase a product, stop using company services, and refer other customers to the company if offered an incentive. When there is an outcome of interest, several analytical techniques can help you understand which information is most strongly associated with it. Before considering some of the popular analytical techniques in detail, the next section makes some general, global observations about predictive modelling in a business context.

4.3 Key Assumptions in Predictive Modelling

Whether you are doing exploratory analysis or building predictive models for use in business, you should keep the following in mind:

- Predictive models assume the future will be like the past. You are assuming relationships found in past data will persist into the future but…
- Past relationships between products, customer characteristics, and behaviors can be expected to last only for as long as other things stay the same—if policies, processes, targeted customer segments, and/or competitive conditions change, relationships observed in the past may no longer be the same.
- Associations in the data do not reveal causality. For example, microfinance data often show that loans with two or more guarantors experience delinquency more than loans with only one guarantor. Intuitively, the opposite might be expected—that two or more guarantors would reduce risk of nonrepayment. But lenders typically ask for additional guarantors only when they already sense a borrower is riskier than average. The delinquency pattern in the data simply confirms the initial belief that those borrowers were riskier. The delinquency is not because the loans have two guarantors.

4.4 Recommended Practices in Predictive Modelling

Predictive models are judged on how well they predict the outcome of interest on data that were not used to build the model. This is called out-of-sample validation.

For example, if you have 10,000 rows of data, it is good practice to:

- Take a random sample of 70 percent, or 7,000 rows, of it to build a model.
- Test the model on the other 30 percent, or 3,000 rows, of data not used to build the model.

When all the data come from the same population and a model works very well only on the 70 percent sample of data used to build it, but not on the other 30 percent used to test/validate it, the model is "overfit" to the 70 percent of the data.

For example, say you are trying to predict the likelihood of purchasing an insurance product and only 200 of your 10,000 rows (2 percent of rows) have a positive response for "purchased insurance product." You will divide your data set for model building as shown in Table 15.

If you build a model on the 7,000 rows of development data using a complex and fully algorithmic modelling methodology, the methodology may assign weights to some associations with the 140 cases of "purchased insurance" outcome that are due purely to chance. This can improve the model's prediction on those 7,000 rows of data, but it can also make it worse on the

TABLE 15. **Model development and validation sample**

	Development	Validation	Total
Purchased insurance	140	60	200
Did not purchase insurance	6,860	2,940	9,800
TOTAL	7,000	3,000	10,000
Insurance Purchase Rate	2%	2%	2%

9 The term "outcome of interest" is used instead of the term "dependent variable" throughout this guide. Both mean the outcome you are seeking to predict with the other data—sometimes called independent variables.

other 3,000 rows, where those same purely chance relationships will not be found—a problem known as "overfitting."

Conditions that increase the risk of overfitting a model include:

- Too few rows of data that have the outcome of interest
- Too few rows of data overall (that limit the scope of out-of-sample testing)
- Too many variables included in the models (especially when combined with few rows of data)
- Excessively complex models (i.e., black-box models).

NOTE: For simplicity, the predictive modelling examples in this guide use the same small (200 row) set of MFI data, and out-of-sample validation is not presented.

The next sections of this guide look at exploratory regression models, classification models such as logistic regression and classification trees, and Bayesian networks. For these types of models, overfitting can be controlled by thoughtfully selecting the features that enter the model (and by thoughtfully interpreting the results). Box 6 highlights some areas of caution on using more complex machine-learning algorithms, such as neural networks, that cannot present the model relationships to humans.

4.5 Regression Analysis

Regression is perhaps the best-known method of predicting an outcome. Regressions fit a straight line to a scatter plot of data. This is easiest to illustrate in the case of two variables, where the regression equation describes the straight line that best fits the relationship of the two variables. Figure 3 shows a trendline fit to a plot of Household Size against Income from the MFI data set.

A regression equation predicts the value of the outcome of interest based on one or more other pieces of data.

Outcome of Interest = Intercept +
Estimate * Predictor Variable

Linear regression in the case of one predictor variable can be done by hand,[10] using the Analysis ToolPak add-in in Microsoft Excel,[11] or using R [A-9]. These tools will give the result:

Intercept	257.72
Predictor Variable's Coefficient	135.56

10 http://sciencefair.math.iit.edu/analysis/linereg/hand/

11 https://www.excel-easy.com/examples/regression.html

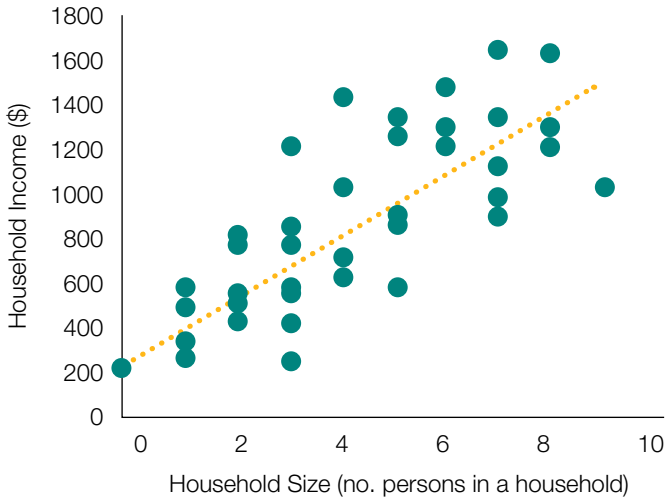
BOX 6. **Cautions on complex models**

"Complicated methods are harder to criticize than simpler ones…" *—Efron 2001*

While some customer data sets can benefit from the power of machine-learning methods more complex than those presented in this guide, caution is warranted when anyone with limited experience in data science uses models that are too complex to interpret—for example, neural networks.

If users do not know or understand what features contribute to the model's prediction, it is difficult to spot errors in the underlying data itself and spurious relationships that are unlikely to persist.

FIGURE 3. **Trendline for household income and household size**



Plugging this regression output into the general regression formula you have:

Income = 257.72 + Household Size * 135.56

If a customer's household size is three members, this model would predict that the household has an income of $664.

$664.43 = 257.72 + 3 * 135.56

Regression can be a relatively easy way to explore which variables have strong relationships with the outcome of

TABLE 16. **Pivot table, counts of "has deposit" by occupation**

Count of has_deposit	Column Labels		
Row Labels	Nontrade	Trade	Grand Total
0 (no deposit)	75	83	158
1 (has deposit)	24	18	42
Grand Total	99	101	200

TABLE 17. **Pivot table, percentage "has deposit" by occupation**

Count of has_deposit	Column Labels		
Row Labels	Nontrade	Trade	Grand Total
0 (no deposit)	76%	82%	79%
1 (has deposit)	24%	18%	21%
Grand Total	100%	100%	100%

interest. However, when real-world data in a regression are correlated, the effects of the variables begin to blend together, and the analyst can lose track of what is driving change in the outcome variable.

Regression works well for predicting numeric outcomes of interest—e.g., a customer's expected total spend or number of purchases. When the outcome of interest is the probability of something happening (or not), such as how likely a customer is to purchase a new product or stop using a service, it is common to use classification models. Some of the most popular and approachable classification models—Logistic Regression and Classification Trees—and the newer and more complex Bayesian Networks are explored in the next sections.

LOGISTIC REGRESSION

Logistic regression is easy to illustrate in the simplest case of one piece of input data and the outcome of interest. For example, you may want to know the probability of being a depositor based on a customer's occupation.[12] For two variables, this question is most easily answered with a cross-tabulation—such as a pivot table in Excel. Table 16 is a pivot table that counts customers with and without deposits according to their occupation—either nontrade or trade. Table 17 is the same pivot table expressed in percentage-of-column totals.

If you think of this pivot table as a model, it predicts (based on past data) that customers working outside of trade have a 24 percent chance (approximately one in every four customers) of being a depositor and customers working in trade have an 18 percent chance (approximately one in five customers) of being a depositor.

When you use statistical software and a logistic regression function to calculate the probability of being a depositor based on occupation, the formulae are more complex, but the answers are the same as the model in Table 17. (See Box 7 for more information.)

While it can be a useful tool for an experienced data analyst, you should use caution applying such methods—they are not a substitute for business knowledge or common sense.

When building a multifactor logistic regression model for a business, the following steps can help to improve transparency and understanding of the model:

1. Test each variable that may have a meaningful relationship to the outcome of interest. For data with such relationships, build single-variable models that can later be entered into a multivariable model.
2. Check for correlation of candidate variables. If one or more variables are highly correlated (i.e., correlation coefficient of > 0.80), select only one of them for the multivariable model.
3. Build the multivariable model one variable at a time, observing how all the model estimates change as you add each variable.

In contrast to logistic regression, classification trees do not require these screening and preparation steps. You can use them to explore the data for interesting and unexpected relationships that increase the probability of the outcome of interest. They can also be used to produce simple, transparent models that front-line users can understand.

12 These are the fields "has_deposit" and "occupation_type" in the MFI data set.

BOX 7. **Math club party tricks: Logistic regression by hand**

In the logistic regression example depicted in Table 17, the probability of being a depositor based on a customer's occupation, for a logistic regression model with only the outcome of interest and one input variable, the counts in Table 16 are transformed into log odds (model estimates) and predicted probability, as shown in Table B7-1.

A logistic model "output" in statistical software such as R [A-10], shows estimates in the format of log odds—along with some other output statistics not discussed in this guide.[a]

COEFFICIENTS

(Intercept)

Estimate Std. Error	-1.1394	0.2345
z value Pr(>|z|)	-4.859	1.18e-06 ***

occupation_typetrade

Estimate Std. Error	-0.3890	0.3501
z value Pr(>|z|)	-1.111	0.267

Illustrating where logistic regression coefficients come from in this guide highlights that a predictive model's power comes primarily from the power of information in the data itself. In Table 17, it is clear where the predicted probabilities come from. In the output of a logistic regression model, it may be less clear—yet both are different expressions of the same information. In practice, the simplest and most complex methods often (but not always) produce remarkably similar results for a given data set—and simple methods have the benefit of generally being easier to understand.

a. See documentation for the glm package in R or any other text on logistic regression that discusses standard errors, z-values and p-values.

TABLE B7-1. **Pivot table, counts of "has deposit" by occupation**

	has_deposit	Column Labels	
	Row Labels	Nontrade	Trade
A	0 (no deposit)	75	83
B	1 (has deposit)	24	18
C	Grand Total	99	101
D	Odds = row B/A	24 / 75 = 0.32	18 / 83 = 0.217
E	Odds Ratio =	0.217/0.32 = 0.678	
F	Log Odds	LN (0.32) = -1.139	LN (0.678) = -0.389
G	Logit Formula Y = Intercept + X * Estimate	-1.139 = -1.139 + 0 *0	-1.528 = 1.139 + 1 * -0.389
H	Probability	EXP (-1.139)/ (1+ EXP (-1.139)) = 0.24%	EXP (-1.528) / (1+ EXP (-1.528)) = 0.18%

Note:
Odds (Row D): The number of "has deposit" divided by the number of "no deposit".
Odds Ratio (Row E): The odds divided by the category with the highest odds (here nontrade).
Log Odds (Row F): The natural log of the odds ratio (for the "reference" category, nontrade, the natural log of its odds is the model "intercept" term).
Logit Formula (Row G): Outcome of Interest = Intercept + Predictor Variable * Estimate.
Probability (Row H): Conversion of the log-odds estimates to probability.

Use highly automated techniques with caution

Procedures such as stepwise regression make it possible to pour in all the data and keep only the features that improve prediction accuracy. However, using intense computation to data mine potential model features is generally not a good substitute for subject area expertise.[13] So, while it can be a useful tool for an experienced data analyst, use caution when applying these methods because they are not a substitute for business knowledge or common sense.

Credit scoring models

Credit scoring models often use logistic regression to predict the probability that a loan will be bad. The estimates of logistic regression models are traditionally converted into point scores where higher credit scores are also better, indicating a lower probability of being bad.[14]

CLASSIFICATION TREES

Classification trees repeatedly split data into contrasting groups to find groups with a larger share of the outcome of interest. Because tree-based algorithms can handle data of any type, including missing values, they can be a relatively easy way to put in all the data at once and see what factors and combinations of factors have been associated with the outcome of interest. They are especially good at identifying segments that may be very small yet have very clear characteristics that can then be identified and addressed via business rules.

Figure 4 shows a classification tree built with the MFI data to predict the probability of being a depositor [A-11]. Each node (shaded box) is labelled with:

- The predicted class (No Deposit or Has Deposit). In this example, if the probability of having a deposit is > 0.50,15 the predicted class is labelled "Has Deposit," otherwise it is labelled "No Deposit."
- The predicted probability of Has Deposit.
- The percentage of observations in the node (Milborrow 2019).

The nodes at the bottom of the tree—the terminal nodes—are the model's predictions. Where the average probability of being a depositor is 21 percent (from Table 17), the two green nodes identify 10 percent of customers with much higher than average probability of being depositors and 10 percent of customers with more than twice the average probability:

- 4 percent of customers have an 86 percent chance of being depositors.
- 6 percent of customers have a 64 percent chance of being depositors.
- 10 percent have a 47 percent chance of being depositors—more than twice the average chance of being depositors.

A rule is written at each split, or branch, of the tree. Assuming that you would like to target the 20 percent of customers most likely to be depositors, you will communicate to the following customers who are not already depositors:

Dark Green	Satisfaction survey score of >= 40 Customer for less than 4.3 years Maximum loan of less than $1,975 Assets less than $12,000 Nontrade occupation
Light Green	Satisfaction survey score of >= 30 Customer for less than 4.3 years Maximum loan of at least $1,975
Light Blue	Same as Dark Green, but Satisfaction Survey score >= 30

Trees have the following advantages:

- They are relatively easy to generate using statistical software.
- Visual presentation aids understanding.
- They can be implemented as a series of IF-THEN statements.

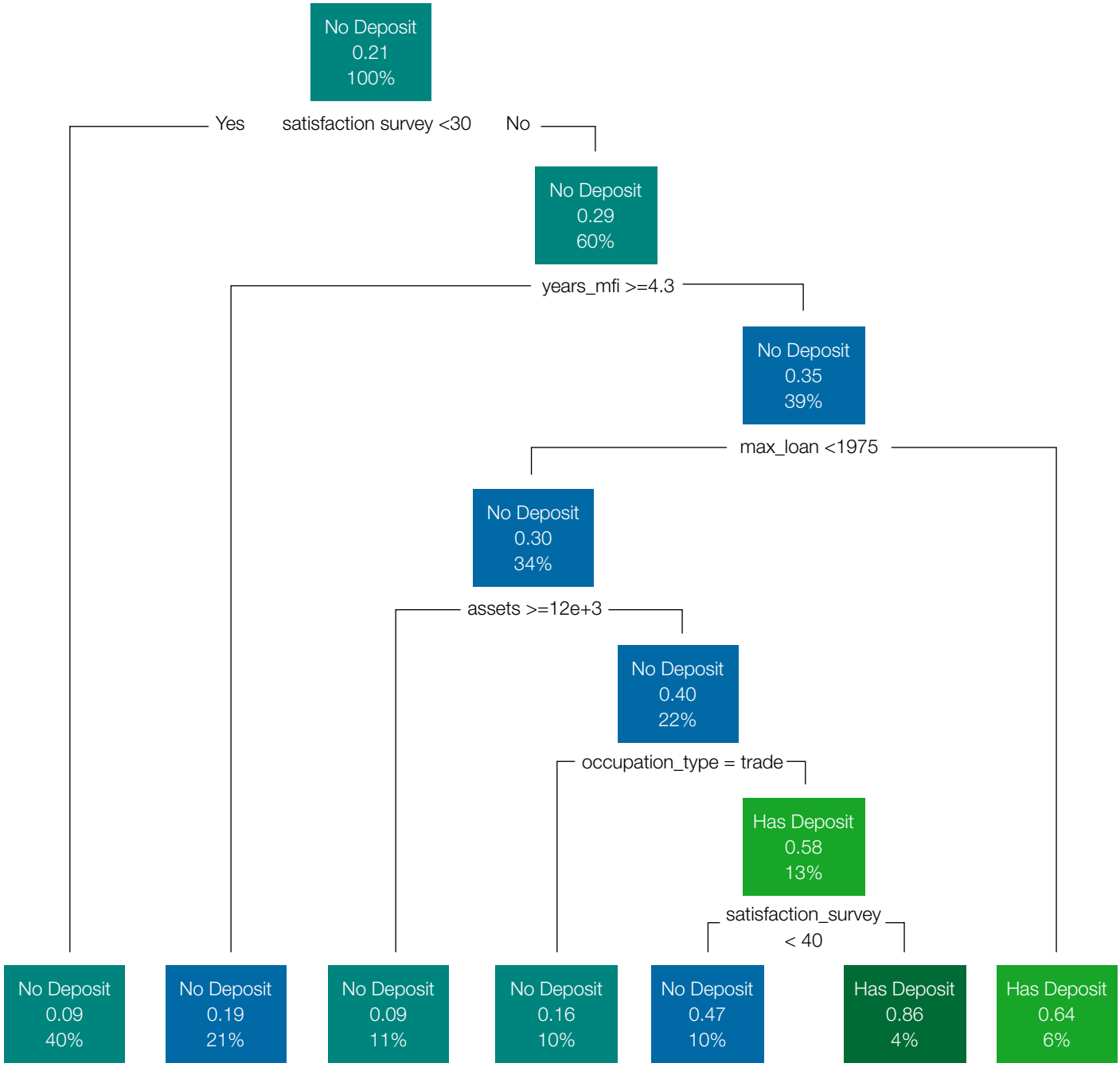
Note that the example MFI data set is very small, with only 200 rows, and the example tree presented did not involve out-of-sample testing. Data can be too small to test out-of-sample (with 200 rows being a good example). When there are too few data, patterns may be very sensitive to certain rows included (or not) in a sample. Be cautious of conclusions drawn from very few data.

13 https://en.wikipedia.org/wiki/Stepwise_regression

14 See the Vidal and Barbon (2019) for a step-by-step guide to implementing a credit scoring project.

15 By default, statistical software for classification models set the threshold for class membership at 50 percent or greater probability; this is not necessarily appropriate for some use cases, and the analyst should adjust them accordingly.

FIGURE 4. **Classification tree—Probability of having deposit**



- They can identify combinations that lead to a higher likelihood of the target outcome.

Limitations of classification trees include the following:

- They may require expertise to get results that best fit tactical/strategic needs.
- The selection of variables and cut points are sensitive to the math used to grow trees, and there are many methods available. Beginners should try the default settings—meant to be most applicable to the general case—and test results against a few competing methods.
- No relationships will be revealed for certain data points that don't meet the significance criteria for making splits in the tree. In this case, other methods (such as single-variable regression or cross-tables) can be used to investigate other variables of interest.

BAYESIAN NETWORKS

An increasingly popular way to predict consumer choice is a set of machine-learning methods called Bayesian networks.[16, 17] These networks can sometimes find patterns that other methods and domain experts miss.

A key difference, and potential advantage, of networks is that, by design, their measures of importance change for all variables when any one variable changes. On the other hand, a regression assumes other variables remain constant when one is changed.[18]

A network's ability to uncover relationships comes with the challenge of interpreting the model results. Networks are graphical analytical methods and should be understood by looking at diagrams rather than at resulting equations. The key relationships in network diagrams are:

- A parent is at the start of an arrow.
- A child of the parent is at the end of an arrow.
- Parents can have several children, and children can have several parents.

Variables directly depend on one another only if they are directly connected. This concept is important when screening variables to include in the model. One technique is to use only variables that depend on the outcome of interest.[19]

If you generate a network for the MFI data set (using the data as is and without additional preparatory work), you may get something like the diagram in Figure 5 [A-12].

Only one feature, satisfaction survey, depends on the outcome of interest variable "has_deposit." On one hand, this doesn't tell you much about the importance of the rest of the variables to being a depositor (although you can also examine their interrelations on the figure). On the other hand, this satisfaction survey factor was also the most important determinant of being a depositor using trees (or regression), so that information is clearly picked up by all methods.[20]

To directly view an output of predicted probabilities for the Bayesian network, you need to create grouped variables. You can do this by dividing each of the numeric variables into two groups (above and below certain thresholds—i.e., above and below the mean or median value for that column) and non-numeric variables in two to three groups [A-13]. The results of a network run on a discretized data set is shown in Figure 6.

This network shows two dependent nodes for "has_deposit":

- Deposits depend on assets.
- Deposits depend on satisfaction survey.

In addition, health insurance depends on assets.

FIGURE 5. **Bayesian network for MFI data set**

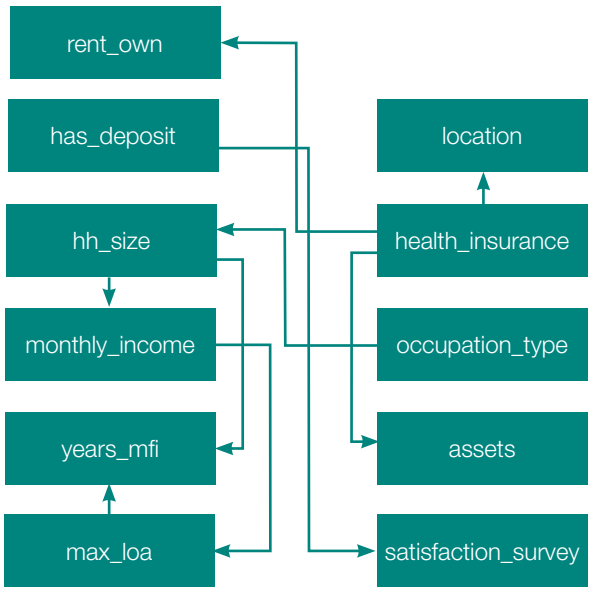
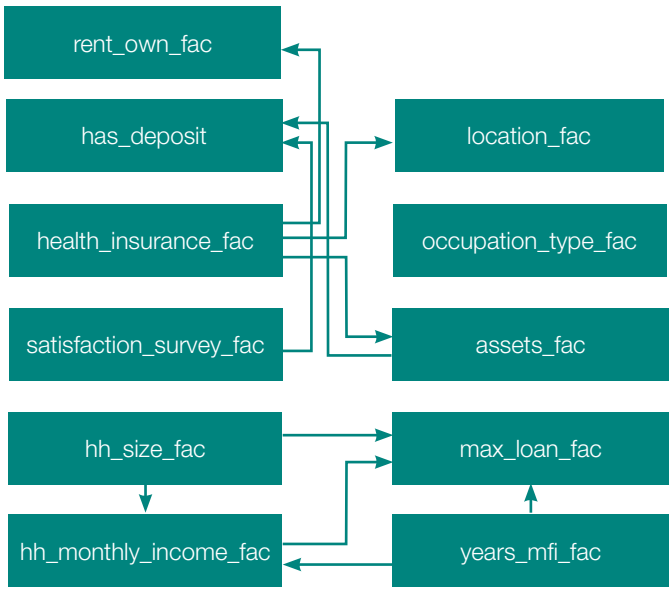


FIGURE 6. **Network with discretized (two-state) variables**



16 For more marketing-oriented examples of using Bayesian networks, see Struhl (2017, Chapter 7). For details of the method and its math, see Barber (2016).

17 Named for English statistician Thomas Bayes.

18 See Struhl (2017, Chapter 7).

19 See Struhl (2017, Chapter 7).

20 Although an exploratory regression for all variables was not shown in the guide, this can be generated using the R-code in the appendix. Satisfaction Survey also has the largest effect on "has_deposit" in a regression.

The probability of being a depositor based on the combined state of its two parent variables is shown in Table 18.

A possible strategy could be to market deposit accounts (or a new deposit product) to new customers who score more than 30 on a satisfaction survey and have assets of less than $11,000 and expect that about 1/3 of them (34 percent) may take up the product.

TABLE 18. **Probability of "Depositor" based on combined state of two-variables**

Satisfaction Survey	Assets	Probability is Depositor	Share of Customers
<=30	<=11,000	11%	25%
>30	**<=11,000**	**34%**	**34%**
<=30	>11,000	3.7%	14%
>30	>11,000	22%	27%

4.6 Conclusion on Predictive Modelling

There are many other predictive modelling methods in addition to regression, trees, and Bayesian networks. This guide has provided only simple examples as a starting point to help you get acquainted with the methods. Remember that insights, for the most part, will come from the data itself, not the specific method or algorithm used. Also remember that data are a representation of reality, but they are not reality itself. Different prediction models may have different results, and a given method may work better in one situation than in another, but the most powerful relationships in the data generally will show up regardless of the method used. As with all analytical methods, it is important to have a good understanding of the business to ensure that the variables used can result in actionable outcomes.

SECTION **5**

MAIN CHALLENGES AND KEY LEARNINGS

The following are some key messages from this guide:

- Significant effort is needed to prepare the data and verify their consistency and accuracy. Insights learned from flawed data will also be flawed.
- The veracity of your analysis using these methods will depend on:
  - Data quality and quantity
  - Degree of variation in the customer data—the fewer products and/or types or groups of customers, the fewer useful insights data-driven analytical approaches are likely to yield
- Associations found in the data do not indicate the impact of future changes. For example, changes in customer targeting, product offers, or credit policies should be expected to lead to new and different results, not only to outcomes observed in the past.
- Use several analytical methods and extract possible insights into the data from each. The best model to use will depend on the use case.

This guide provides an entry point to data-driven analytics. In practice, many more questions will arise than have been answered here. The more complex the analytical problem and the larger and richer the data set, the more specialist skills (i.e., the input of a trained data scientist) may be needed to ensure the analysis is performed correctly. Beginners and experts alike should keep in mind the following:

- Models and analyses should be consistent over different samples of the data.
- If a use case makes sense when it is tested with data, it should make sense in real life.
- For many business applications, if a model looks too good to be true, there is probably a mistake somewhere.

REFERENCES

Breiman, Leo. 2001. *Statistical Modeling: The Two Cultures*. Institute of Mathematical Statistics. https://www.jstor.org/stable/2676681

Caire, Dean Eduard, Leonardo Camiciotti, Soren Heitmann, Susie Lonie, Christian Racca, Minakshi Ramji, and Qiuyan Xu. 2017. "Data Ana¬lytics and Digital Financial Services: Handbook." Washington, D.C.: World Bank Group. http://documents.worldbank.org/curated/en/142171531294752716/Data-analytics-and-digital-financial-services-handbook

CGAP. 2016. "Customer Segmentation Toolkit." Washington, D.C.: CGAP. http://www.cgap.org/publications/customer-segmentation-toolkit

Efron, Bradley. 2001. *"Statistical Modeling: The Two Cultures." Comment.* Institute of Mathematical Statistics. https://www.jstor.org/stable/2676683

Gavett, Gretchen. 2014. "What You Need to Know About Segmentation." *Harvard Business Review.* https://hbr.org/2014/07/what-you-need-to-know-about-segmentation

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. 2013. *An Introduction to Statistical Learning: with Applications in R*. Springer. https://www.springer.com/us/book/9781461471370

Jolliffe, Ian. 2002. Principal Component Analysis. Springer. https://www.springer.com/us/book/9780387954424

Schreiner, Mark. 2018. "Simple Poverty Scorecard Tool Mali." Scorocs. http://www.simplepovertyscorecard.com/MLI_2016_ENG.pdf

Struhl, Steven. 2017. "Artificial Intelligence Marketing and Predicting Consumer Choice: An Overview of Tools and Techniques." KoganPage. https://www.koganpage.com/product/artificial-intelligence-marketing-and-predict-ing-consumer-choice-9780749479558

Vidal, Maria Fernandez, and Fernando Barbon. 2019. "Credit Scoring in Financial Inclusion." Washington, D.C.: CGAP.

APPENDIX

BY DEAN CAIRE, WITH RAGUL RAM

Using R Software to Replicate Guide Examples

To practice the statistical techniques presented in this guide:

1. Download and install the R software and necessary libraries. The software and libraries are available at http://www.r-project.org/.
2. Install the recommended packages.
3. Download the data file used in these examples.

This appendix walks you through the R set-up and reading in the data. It then points to the appendix references, which are set in red type throughout this guide. The references point to code examples needed for each method. The code used in the appendix is available as a text file and as an R file.

INSTALLING R

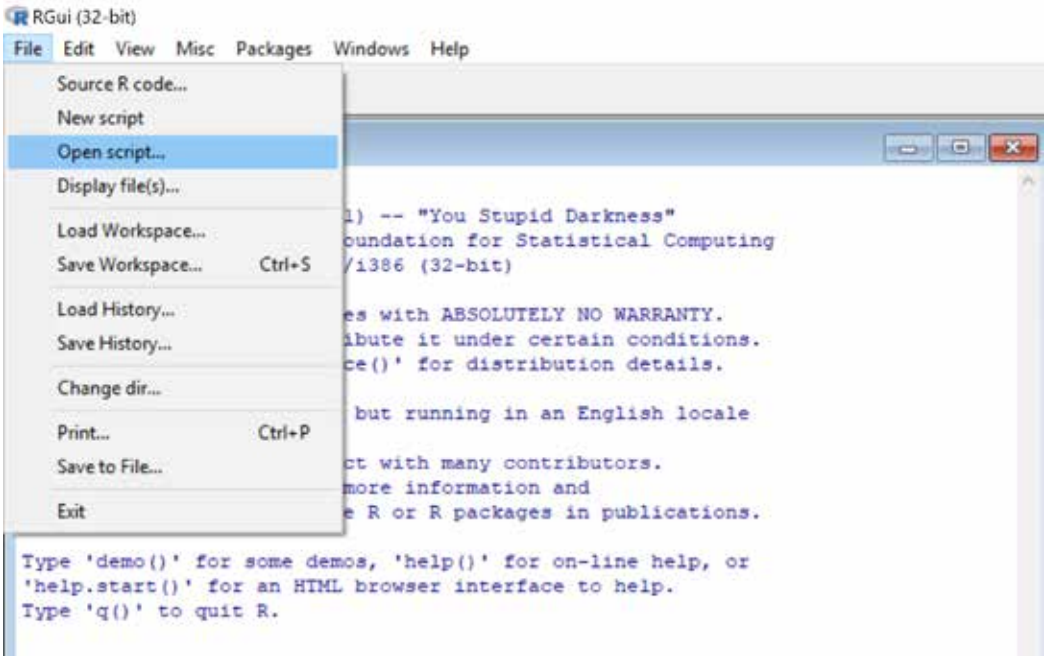
The latest version of the R software can be downloaded free of charge. The web page provides installation instructions.

You may want to read some basic introductory materials and/or watch one of the many introductory tutorial videos available online, for example, https://www.datacamp.com/courses/free-introduction-to-r and https://www.youtube.com/watch?v=32o0DnuRjfg.

Once you have installed R, open it by double-clicking on the desktop R icon and follow these steps:

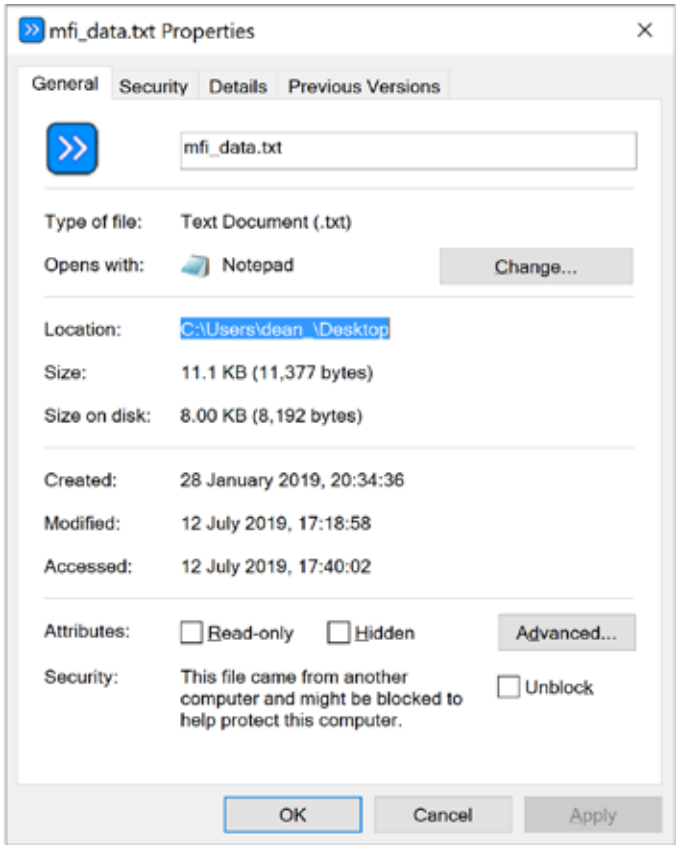
IMPORTING DATA AND INSTALLING PACKAGES

1. **Open a new script in R (File -> New script)**

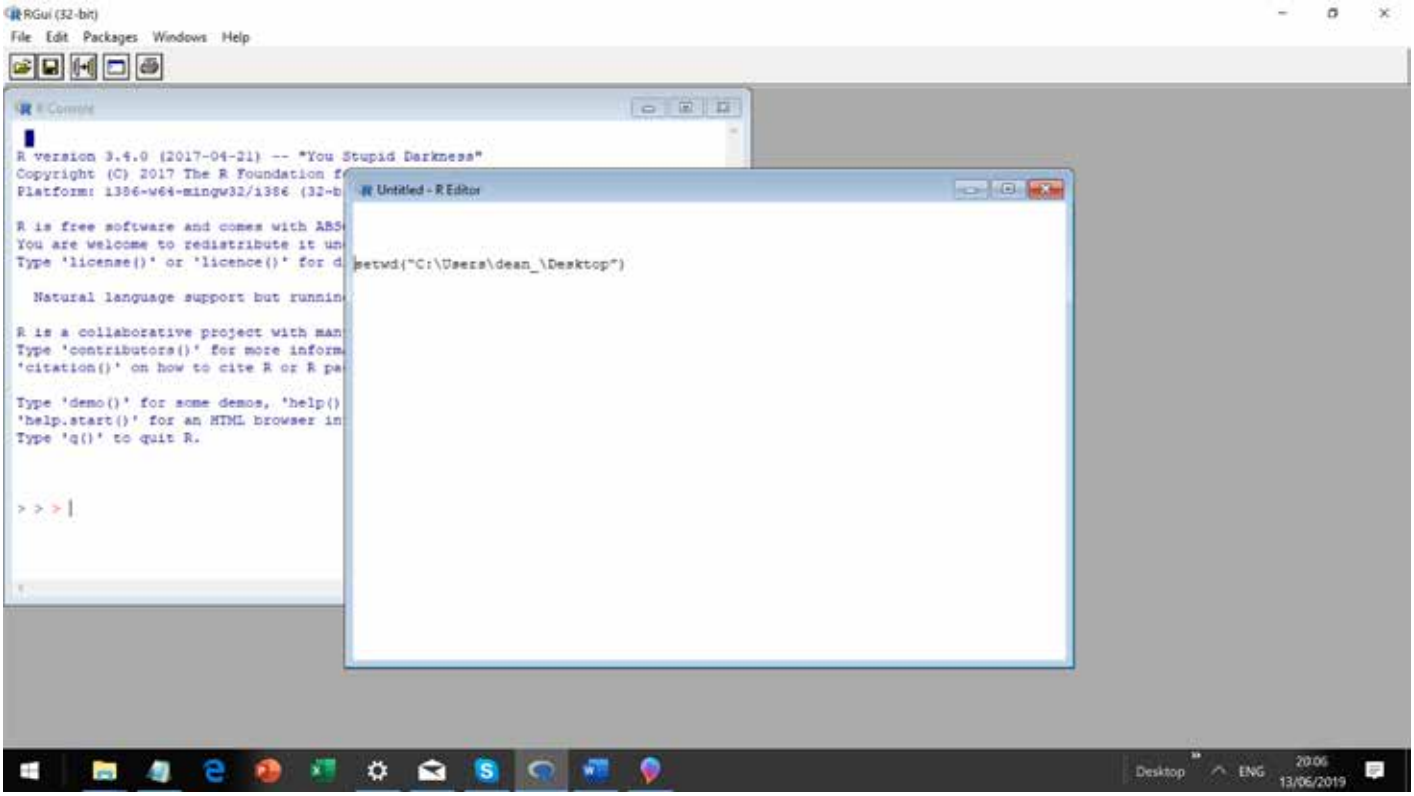


2. **Set a path to the location on your computer where you have put the data file you will analyze.**

An easy way to do this is to navigate to the file and right click on it, which launches a properties window:



Copy the path to the file location (outlined in blued) and paste it into the script editor as follows:



Double the forward slashes in the path and add the text:

setwd("C:\\Users\\dean_\\Desktop")

3. Import the data

The sample data file is a tab-delimited text file:

- The first row contains column headings.
- All subsequent rows contain data records; there are no extra cells such as lines with totals or other single cells not relating to columns or rows of the table.
- Data are formatted correctly.
- There are no missing values.

The script to read in the data file is:

```
mfi <-read.delim("mfi_data.txt")
```

To run this code (and read in the data file), highlight this text in the R script editor and either click the button (which is circled in the figure below) or enter the key combination "ctrl + R".

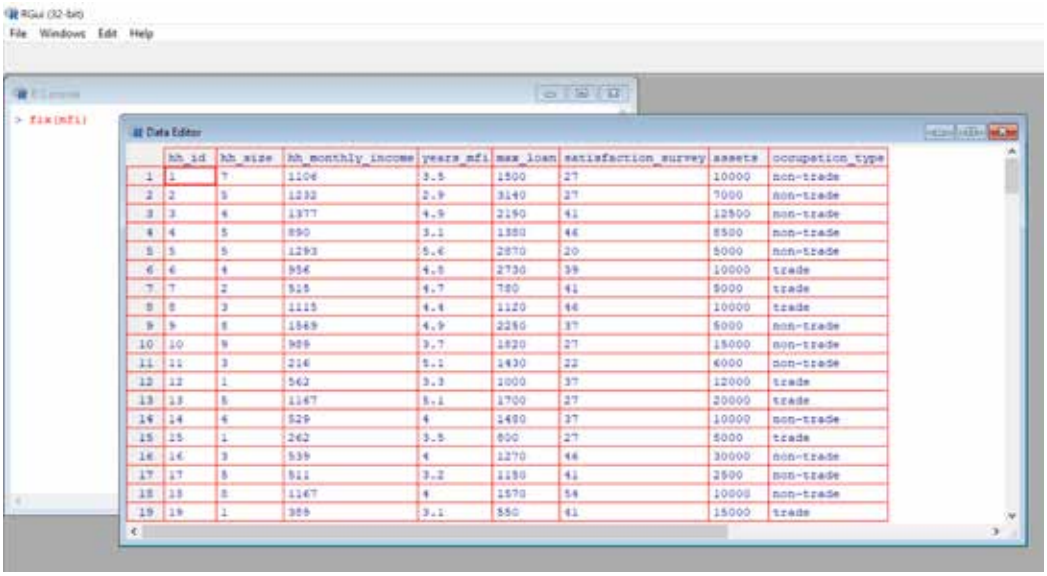
In the above code, create an object arbitrarily named "mfi" to which you assign (using "<-") a data table that contains the data in the .txt file.

View your imported data table in the R data editor using the "fix" command, which brings up a spreadsheet view of the data in R as shown below

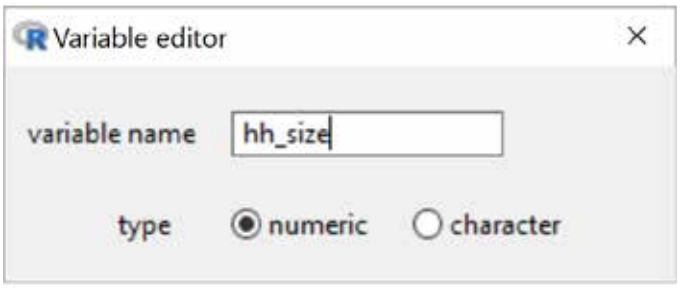
fix (flat)

Again, highlight that code and click the button or use "ctrl + R".

Data Editor in R

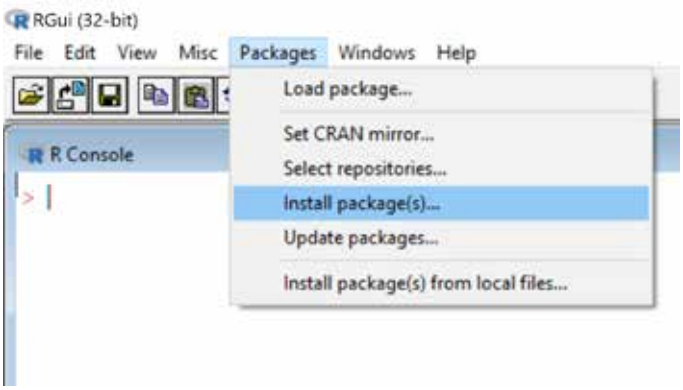


You can visually check your data by double-checking the number of rows in the table, checking that text (string) values are visible (especially if you are working with various languages), checking that numeric data are formatted as numeric data, etc. The variable name and data type can be viewed by clicking on the column name in the data editor. This calls up a message box of the type:



4. Install the packages in R

Navigate to Packages -> Install package(s)…



Select any of the online repositories (such as the first: 0-Cloud[https] and click OK.

Find the following packages, and, one at a time, highlight them and click OK.

cluster
factoextra
magrittr
Hmisc
dplyr
mfi

This downloads the function libraries to the R directory on your computer. These need to be installed only once; they will be available for future use by running code to load the library into memory. Here are the codes to launch the libraries for this guide:

```
library(cluster)
library(factoextra)
library(magrittr)
library (Hmisc)
library (dplyr)
```

STATISTICAL TECHNIQUES USED IN THE GUIDE

[A-1] Descriptive Statistics

```
str(mfi)

###looks like this

library (Hmisc)

describe(mfi)

###looks like this:
```

[A-2] Inter-quartile Range

Note: This code was not used on the example data set for the examples shown in this guide.

```
dsBase <- mfi
# I wanted to keep a copy of the original data set
dsBase.iqr <- dsBase

# Create a variable/vector/collection of the column names
you want to remove outliers on.
vars <- c("hh_size", "assets")

# Create a variable to store the row id's to be removed
Outliers <- c()
 # Loop through the list of columns you specified
for(i in vars){

 # Get the Min/Max values
 max <- quantile(dsBase.iqr[,i],0.75, na.rm=TRUE) +
(IQR(dsBase.iqr[,i], na.rm=TRUE) * 3 )
 min <- quantile(dsBase.iqr[,i],0.25, na.rm=TRUE) -
(IQR(dsBase.iqr[,i], na.rm=TRUE) * 3 )

 # Get the id's using which
 idx <- which(dsBase.iqr[,i] < min | dsBase.iqr[,i] > max)

 # Output the number of outliers in each variable
 print(paste(i, length(idx), sep=")")

 # Append the outliers list
 Outliers <- c(Outliers, idx)
}

# Sort, I think it's always good to do this
Outliers <- sort(Outliers)

# Remove the outliers
dsBase.iqr <- dsBase.iqr[-Outliers,]
fix(dsBase.iqr)
####source: http://stamfordresearch.com/
outlier-removal-in-r-using-iqr-rule/
```

[A-3] Scale Numeric Data

```
##Create a data frame with only the numeric variables
to be analyzed

mfi_num <- mfi[,c("hh_size",

"hh_monthly_income",
"years_mfi",
"max_loan",
"satisfaction_survey",
"assets")]

###scale the variables and save in a data frame named
'mfi_features'

mfi_features <- scale(mfi_num)
```

[A-4] Correlation Matrix

```
cor(mfi_num)
```

[A-5] Principal Component Analysis

```
###Principal component analysis

pr.out =prcomp (my_data , scale =TRUE)

pr.out$rotation

biplot (pr.out , scale =0)

pr.var =pr.out$sdev ^2

pve=pr.var/sum(pr.var )

pve

plot(cumsum (pve ), xlab="Principal Component",
ylab = "Cumulative Proportion of Variance
Explained", ylim=c(0,1) ,type= "b")
```

[A-6] K-means clustering

```
set.seed(123)
km.res <- kmeans(my_data, 3, nstart = 25)
# Visualize
library("factoextra")
fviz_cluster(km.res, data = my_data,
ellipse.type = "convex",
```

```
palette = "jco",
ggtheme = theme_minimal())
```

[A-7] Clustering for mixed data types

```
mfi_cat <- mfi[,c("occupation_type",

"health_insurance",
"location",
"rent_own")]

library(cluster)
library(clustMixType)
library(Rtsne)

all_mfi <- cbind(mfi_features,mfi_cat)


k.pres_2 <- as.matrix(daisy(all_mfi, metric = "gower"))
k.pres_2

library(magrittr)
library (dplyr)

#fix(all_mfi)
pam_fit <- pam(k.pres_2, diss = TRUE, k = 3)

pam_results <- mfi %>%
 dplyr::select(-hh_id) %>%
 mutate(cluster = pam_fit$clustering) %>%
 group_by(cluster) %>%
 do(the_summary = summary(.))

pam_results$the_summary

tsne_obj <- Rtsne(k.pres_2, is_distance = TRUE)

tsne_data <- tsne_obj$Y %>%
 data.frame() %>%
 setNames(c("X", "Y")) %>%
 mutate(cluster = factor(pam_fit$clustering),
 name = mfi$hh_id)

ggplot(aes(x = X, y = Y), data = tsne_data) +
 geom_point(aes(color = cluster))

##This is the output of the mixed clustering 3-cluster
solution with the MFI data
```

[A-8] Methods for finding mathematically optimal number of clusters

Note: this code was not used on the example data set for the examples shown in the guide and will require that the data set be named 'data_set' to use the code directly.

```
data_set <- mfi_num
 #Step 1
 # Sample for Clustering #

 clust_sample<-data_set

 #Step 2
 #Clculating the dissimilarity matrix #
 gower<-daisy(clust_sample, metric="gower")

 #sample for k means
 data.scaled<-clust_sample
 scaled_final<-scale(data.scaled)
 #Calculating The number of Clusters

 #Step 3
 # Calculate optimum numbe of clusters #
 sil_width<-c(NA)

 for(i in 2:10){
 pam_fit<-pam(gower, diss= TRUE, k=i)
 sil_width[i]<-pam_fit$silinfo$avg.width
 }

 #Additional methods to calculate Optimum number of
clusters
 #gower_mat <- as.matrix(gower)

 #fviz_nbclust(gower_mat, pam, method = "wss") +
 # geom_vline(xintercept = 3, linetype = 2)
 # Plot silhoutte width

 # nb <- NbClust(scaled_final, distance = "Euclidean",
min.nc = 2,
 # max.nc = 10, method = "complete", index ="all")

 plot(1:10, sil_width,
 xlab = "Number of clusters",
 ylab = "Silhouette Width")
 lines(1:10, sil_width)
```

[A-9] Simple Linear Regression

```
fit <- lm(mfi$hh_monthly_income ~ mfi$hh_size,
data=mfi)
summary(fit) # show results
```

[A-10] Logistic Regression

```
library(caTools)

mfi$has_deposit[mfi$deposit_balance > 0]      <- 1
mfi$has_deposit[mfi$deposit_balance <= 0]     <- 0

logit_model <- glm (mfi$has_deposit ~

 mfi$occupation_type          +
  1
 , binomial())
summary(logit_model)

mfi$logit_model =predict(logit_model,type =
"response")

colAUC(mfi$logit_model,mfi$has_deposit,
plotROC=TRUE)
```

[A-11] Classification Trees

```
library(rpart)
library(party)
library(rpart.plot)

mfi$has_deposit_num = factor(mfi$has_deposit,la-
bels=c("No Deposit","Has Deposit"))

binary.model <- rpart(has_deposit_num ~ hh_size +
hh_monthly_income  +
years_mfi          +
max_loan           +
satisfaction_survey    +
assets             +
occupation_type    +
health_insurance   +
location           +
rent_own,
      method="class", data=mfi, cp =.01)

summary(binary.model) # detailed summary of splits
rpart.plot(binary.model,tweak=1.3)
```

[A-12] Bayesian Networks

```
mfi[12] <- lapply(mfi[12:12], as.factor)
mfi[2:7] <- lapply(mfi[2:7], as.numeric)

bayes_variables_multinomial <- mfi[,c("has_deposit",
"hh_size",
"hh_monthly_income",
"years_mfi",
"max_loan",
"satisfaction_survey",
"assets",
"occupation_type",
"health_insurance",
"location",
"rent_own")]

library(bnlearn)
res <- hc(bayes_variables_multinomial)
plot(res)

fittedbn <- bn.fit(res, data =
bayes_variables_multinomial)
fittedbn
```

[A-13] Bayesian Networks

```
#factor_satisfaction
mfi$satisfaction_survey_bin[mfi$satisfaction_survey >
30]  <- 1
mfi$satisfaction_survey_bin[mfi$satisfaction_survey <=
30]  <- 0
mfi$satisfaction_survey_fac = factor(mfi$satisfaction_
survey_bin,labels=c("<=30",">30"))

#hh_size_fac
mfi$hh_size_bin[mfi$hh_size > 3]      <- 1
mfi$hh_size_bin[mfi$hh_size <= 3]     <- 0
mfi$hh_size_fac =
factor(mfi$hh_size_bin,labels=c("<=3",">3"))

#hh_monthly_income
mfi$hh_monthly_income_bin[mfi$hh_monthly_income
> 800]          <- 1
mfi$hh_monthly_income_bin[mfi$hh_monthly_income
<= 800]         <- 0
mfi$hh_monthly_income_fac = factor(mfi$hh_
monthly_income_bin,labels=c("<=800",">800"))

#factor_satisfaction
```

```
mfi$years_mfi_bin[mfi$years_mfi > 4]            <- 1
mfi$years_mfi_bin[mfi$years_mfi <= 4]
<- 0
mfi$years_mfi_fac =
factor(mfi$years_mfi_bin,labels=c("<=4",">4"))

#max_loan
mfi$max_loan_bin[mfi$max_loan > 1500]
<- 1
mfi$max_loan_bin[mfi$max_loan <= 1500]
<- 0
mfi$max_loan_fac =
factor(mfi$max_loan_bin,labels=c("<=1500",">1500"))

#assets
mfi$assets_bin[mfi$assets > 11000]              <- 1
mfi$assets_bin[mfi$assets <= 11000]             <- 0
mfi$assets_fac =
factor(mfi$assets_bin,labels=c("<=11,000",">11,000"))
mfi$occupation_type_fac = factor(mfi$occupation_
type,labels=c("non-trade","trade"))

#factor health insurane
mfi$health_insurance[mfi$health_insurance ==
"Missing"] <- "No"
mfi$health_insurance_fac =
factor(mfi$health_insurance,labels=c("No","Yes"))

#"location",
mfi$location_bin_1[mfi$location == "Rural"]    <- 0
mfi$location_bin_1[mfi$location == "Countryside"] <- 0
mfi$location_bin_1[mfi$location == "No"]       <- 0
mfi$location_bin_1[mfi$location == "Semi-urban"] <- 1
mfi$location_bin_1[mfi$location == "Urban"]    <- 2

mfi$location_fac = factor(mfi$location_bin_1,la-
bels=c("Rural","Semi-urban","Urban"))

mfi$rent_own_bin_1[mfi$rent_own == "No"]   <- 0
mfi$rent_own_bin_1[mfi$rent_own == "Rent"] <- 0
mfi$rent_own_bin_1[mfi$rent_own == "Owned"] <- 1

mfi$rent_own_fac =
factor(mfi$rent_own_bin_1,labels=c("Rent","Owned"))
####BAYES NETWORK Expiriment with multino
mial (all factored)")]

bayes_discrete <- mfi[,c("has_deposit",
"health_insurance_fac",
"satisfaction_survey_fac",
"hh_size_fac",
"hh_monthly_income_fac",
"years_mfi_fac",
"max_loan_fac",
"assets_fac",
"occupation_type_fac",
"location_fac",
"rent_own_fac")]

bayes_variables_predictors
<- mfi[,c("health_insurance_fac",
"satisfaction_survey_fac",
"hh_size_fac",
"hh_monthly_income_fac",
"years_mfi_fac",
"max_loan_fac",
"assets_fac",
"occupation_type_fac",
"location_fac",
"rent_own_fac")]
library(bnlearn)
res <- hc(bayes_discrete)
plot(res)

fittedbn <- bn.fit(res, data = bayes_discrete)
fittedbn

predicted = predict(fittedbn, node = "has_deposit", data
= bayes_variables_predictors)
predicted
```

CGAP
cgap.org