Lecture 7: A Massive and Messy Marketing Survey

Marketing Overview

- Our whole economy is based on buying and selling products and services… or something like that.
- Finding out what people like and don't like can make all of the difference for a business.
- Even the best products require extensive marketing to succeed.

Marketing Versus Advertising

- Marketing (or market research) is the process of learning about the potential customers for a product or service.
- Advertising is a set of actions used to make potential customers more likely to purchase a product or service.
- Advertising makes use of marketing. Marketing can examine the impacts of advertising.

Customer Research

- The way things used to be: Small businesses, personal interactions, observations.
- As businesses grew, they needed a replacement for the in-person interactions that once informed their decisions.
- Today, businesses increasingly rely on data.
- Marketing data can be used to evaluate many decisions a business might face.

Surveys

- **Questionnaires**: typically written.
- **Questions**: Usually straightforward.
- **Answers**: Usually multiple choice.

Difficulties with Surveys

- **How you ask the question** can influence the results.
- Respondents may tell you **what they think you want to hear.**
- The setting, length, and format can have an impact on the respondent's answers.

Are Surveys Representative of the Population?

- **Probably not!**
- Nonetheless, they may be representative of important segments of the customers.
- The sampling methodology and people's willingness to participate are potentially **sources of biases**.
- Compared to medical outcomes, marketing data are typically viewed as presenting a lower standard of evidence.

Case Study: A Massive, Messy Marketing Survey

Setting: A real world marketing analytics project with a big company.

The Challenges:

- Understand everything about their surveys.
- Organize, clean, and process all of their data.
- Develop models for each state of customer engagement for each product.
- Identify the idiosyncratic factors that impact each model.
- Drill down into the important subgroups for each product and state of engagement.
- Develop **dynamic reporting tools** to summarize all of these results.
- AND do all of this in **10 weeks**. (Yikes!) We will spend the next few classes discussing all of these aspects of data science and its applications to surveys in marketing projects.

The Company

- A manufacturer in the snack food industry (but it was actually something else).
- They produce a wide variety of brands, both well-known and not so well-known.
- They want to better understand their customers' engagement with their brands and those of the competition.

The Products

- **Categories**:
    1. *Baked Goods;*
    2. *Candies;*
    3. *Salty Snacks.*
- **Pricing Categories**:
    1. *Value;*
    2. *Mainstream;*
    3. *Specialized;*
    4. *Premium;*
    5. *Ultra Premium.*
- Research directed at their own products and those of the competition.

The Surveys

- **Way too long**: 100 pages of questions, 45-60 minutes to complete.
- Administered by a research company that gives **rewards and points**.
- Taken by **semi-professional survey respondents** who like to collect rewards and points.

Red Flags

- **Survey length**: An hour long survey might not be the best means of collecting accurate information.
- The research company is more interested in **obtaining a quantity** than in selecting high quality respondents.
- The semi-professional respondents **may not be at all representative** of the market for these products. They may answer questions differently because of their vast experience taking surveys or speed through the assignment in anticipation of a quick reward.

All of these factors were potential concerns. While we could make some small recommendations about the design of the survey, the company was committed to the overall approach. They were more interested in understanding the results for the data they had than in revamping the survey.

Similar Surveys, But Not Quite Unified

We noticed a variety of differences among the surveys for the 3 product categories:

- Multiple choice **options in different orders**;
- Same Question, **different variable names** for Salty Snacks versus the others.
- An **additional survey** about shopping experiences for Baked Goods; there was extra data to incorporate for these products.

Example of Small Differences

In which kind of store are you **most likely to shop** for the product?

1. A grocery store.
2. A candy store.
3. A pharmacy.
4. A convenience store.
5. A bakery.

However, the **Salty Snacks** products are not sold in candy stores or bakeries. The same question would have different multiple choice options in each survey.

To Unify or Not to Unify?

- **Uniform questions and data collection practices** put everything in the same terms.
- It's **more streamlined** to take a uniform approach.
- However, some products have their own **idiosyncracies**:
    1. *Specialized advertising;*
    2. *Cult followings;*
    3. *Geographic bases, etc.*

Ultimately, you have to decide whether the advantages of a unified survey outweigh the advantages of collecting more specific information about certain products.

Survey Questions: Data Conversion

During the past 6 months, how often did you consume this brand of snack food?

Show 10 entries Search:

Option	Response
1	Daily
2	6 times per week
3	5 times per week
4	4 times per week
5	3 times per week
6	2 times per week
7	1 time per week
8	3 times per month
9	2 times per month
10	1 time per month

Showing 1 to 10 of 13 entries Previous 1 2 Next

These values are text entries. We'd like to convert them into a numeric form. What is the best way to proceed?

Annualizing the Rates of Consumption

We'll convert all of the values to their corresponding yearly rates of consumption.

Show 10 entries Search:

Response	Annualized.Consumption
Daily	365.25
6 times per week	313.07
5 times per week	260.89
4 times per week	208.71
3 times per week	156.54
2 times per week	104.36
1 time per week	52.18
3 times per month	36
2 times per month	24
1 time per month	12

Showing 1 to 10 of 13 entries Previous 1 2 Next

Writing a Conversion Function

```
annualize.consumption <- function(x) {
    library(plyr)
    written.rates <- c("Daily", sprintf("%d times per week",
        6:2), "1 time per week", sprintf("%d times per month",
        3:2), "1 time per month", "2 times in the past 3 months",
        "1 time in the past 3 months", "1 time in the past 6 months")
    annualized.rates <- c(365.25 * (7:1)/7, 12 * (3:1),
        4 * (2:1), 2 * (1))
    y <- mapvalues(x = x, from = written.rates, to = annualized.rates)
    return(y)
}
```

给出了所有文字的pattern

给出了转化的标准

mapvalues(x, from, to)：x为原始值（字符串+数字），函数作用于x
把from中的内容换成to中的，以把字符串变成数字
有多个的情况，可以以向量(7:1)的形式给出

Questions Requiring Text Processing

- **Hundreds of questions** across multiple surveys – it's laborious.
- Each multiple choice question needs to be formatted in a manner that makes sense to analyze.
- **Value Conversions**:
- From multiple choice to Yes/No.
- From text to Numeric Values (e.g. 0-10 scales).

Example: Measuring Advocacy

Have you recommended this brand in the past month?

```
+ 1:  I recommend this brand all of the time.
+ 2:  I recommend it when people ask about it.
+ 3:  I am neutral.
+ 4:  I recommend against trying this brand when I'm asked.
+ 5:  I recommend against trying this brand all of the time.
```

If we want to classify the respondents as advocates, we could use the following conversion:

```
recommendation.responses <- c("I recommend this brand all of the time", "I recommend it when people ask about it.")
dat[, advocates.this.brand := 1*(recommend_response %in% recommendation.responses)]
```

把多类的多项选择变成两类

This converts the text into numeric values, where 1 represents an advocate and 0 is for everyone else.

An Annotated Survey

Product	Brand Perception Variable	Values
1. Cookie Crumble	BP_1 through 250_1	1 = 0 Strongly Disagree
2. Sweet Saltines	BP_1 through 250_2	2 = 1
3. Fig Out!	BP_1 through 250_3	3 = 2
4. Brownie Pops	BP_1 through 250_4	4 = 3
5. Pretzelicious	BP_1 through 250_5	5 = 4
6. PB and Jelly Beans	BP_1 through 250_6	6 = 5
7. Caked On	BP_1 through 250_7	7 = 6
8. Gummi Beans	BP_1 through 250_8	8 = 7
9. Chip Strips	BP_1 through 250_9	8 = 8
10. Choco Loco	BP_1 through 250_10	10 = 9
11. Frostipops	BP_1 through 250_11	11 = 10 Strongly Agree
12. Lookie's Cookies	BP_1 through 250_12	**Questions**
13. Ice Cream Dough Cookies	BP_1 through 250_13	1. A brand for me.
14. Mousse Malt Magic	BP_1 through 250_14	2. Fits my budget.
15. Tiramisoup	BP_1 through 250_15	3. Tastes great.
16. Popcorn Packs	BP_1 through 250_16	4. Good to share with others.
17. Browniemint Bark	BP_1 through 250_17	5. I like the logo.
18. Cocoa Bears	BP_1 through 250_18	6. For special occasions.
19. Frozen Frogurt	BP_1 through 250_19	7. An everyday snack
20. Studel Noodles	BP_1 through 250_20	8. Healthy.
21. The Fine Tarts	BP_1 through 250_21	9. Delicious
22. Peanut Brittle Littles	BP_1 through 250_22	10. Just the right amount.
23. Chippy Cheese	BP_1 through 250_23	11. Relaxing.

Learning from the Annotated Survey

- There are seemingly 250 products mentioned in the survey.
- There are at least 23 different Brand Perception questions.
- There are quasi-numeric answers on a scale from 0 to 10 about each combination of a brand and a perception.

A Look at the Data

- The file **Simulated Marketing Data – Original Form.csv** contains information from the (fictionalized) marketing survey's data.
- This file was limited to a smaller number of products (23) and a subset of all of the traits that were measured.

```
dat <- fread(input = "Simulated Marketing Data -- Original Form.csv")
dim(dat)
```

```
[1] 100000    374
```

- That… is a lot of columns.
- In reality, the number of columns was **closer to 20 thousand**, while the whole survey may have had 20-50 thousand rows of respondents depending on the portion that was conducted.

Some Column Names

```
datatable(data = data.table(names = names(dat)))
```

Show 10 entries Search:

	names
1	id
2	age
3	gender
4	income
5	region
6	persona
7	Awareness_4
8	Awareness_17
9	Awareness_7
10	Awareness_9

Showing 1 to 10 of 374 entries Previous 1 2 3 4 5 … 38 Next

Patterns in Column Names

- Many of the names have similar first words. We can identify some patterns:

```
the.pieces <- strsplit(x = names(dat), split = "_")
first.pieces <- lapply(X = the.pieces, FUN = function(x) {
    return(x[1])
})
the.patterns <- unique(as.character(first.pieces))
print(the.patterns)
```

分成两块，然后选出第一块

```
 [1] "id"            "age"           "gender"        "income"
 [5] "region"        "persona"       "Awareness"     "BP"
 [9] "Consideration" "Consumption"   "Satisfaction"  "Advocacy"
```

- This is a more manageable number of categories to investigate.

Nonetheless, Numerous Problems to Address

- The names of the variables don't easily link to the products or traits.
- The values of the responses are not numeric – and even the description has mistakes.
- With 250 products and 50 traits, there are 12,500 columns of data for the Brand Perceptions alone. This is a peculiar structure.

We will investigate each of these issues more thoroughly.

Issue #1: Decrypting the Names

Product	Brand Perception Variable	Values
1. Cookie Crumble	BP_1 through 250_1	1 = 0 Strongly Disagree
2. Sweet Saltines	BP_1 through 250_2	2 = 1
3. Fig Out!	BP_1 through 250_3	3 = 2
4. Brownie Pops	BP_1 through 250_4	4 = 3
5. Pretzelicious	BP_1 through 250_5	5 = 4
6. PB and Jelly Beans	BP_1 through 250_6	6 = 5
7. Caked On	BP_1 through 250_7	7 = 6
8. Gummi Beans	BP_1 through 250_8	8 = 7
9. Chip Strips	BP_1 through 250_9	8 = 8
10. Choco Loco	BP_1 through 250_10	10 = 9
11. Frostipops	BP_1 through 250_11	11 = 10 Strongly Agree
12. Lookie's Cookies	BP_1 through 250_12	**Questions**
13. Ice Cream Dough Cookies	BP_1 through 250_13	1. A brand for me.
14. Mousse Malt Magic	BP_1 through 250_14	2. Fits my budget.
15. Tiramisoup	BP_1 through 250_15	3. Tastes great.
16. Popcorn Packs	BP_1 through 250_16	4. Good to share with others.
17. Browniemint Bark	BP_1 through 250_17	5. I like the logo.
18. Cocoa Bears	BP_1 through 250_18	6. For special occasions.
19. Frozen Frogurt	BP_1 through 250_19	7. An everyday snack
20. Studel Noodles	BP_1 through 250_20	8. Healthy.
21. The Fine Tarts	BP_1 through 250_21	9. Delicious
22. Peanut Brittle Littles	BP_1 through 250_22	10. Just the right amount.
23. Chippy Cheese	BP_1 through 250_23	11. Relaxing.

- The first number ranges from 1 to 250. These are the products' indices.
- The second number ranges from 1 to at least 23. The real survey had approximately 50 of these questions. These indices correspond to the question being asked about the respondent's perception to the brand.
- From the survey, we can see that **BP_14_11** is the variable for Question 11 about Product 14. On a scale from 0 to 10, how relaxing does the respondent find Mousse Malt Magic?

Lots of Columns

- 250 products
- 50 brand perception questions.
- 12500 total columns for brand perceptions across all of the products.
- Meanwhile, the simulated data was limited to 11 questions across 23 products for a total of 253 columns related to brand perceptions.

Improving the Naming Conventions

Using numbers in place of the names of the products and the traits can create confusion and errors.

- Off by 1 errors caused by adding products to or removing them from the list.
- Accidentally switching the numbers for the product and the trait.
- Perhaps even confusing the numeric data (on a 0-10 scale) for the product or trait.

It is better to maintain an unambiguous link between the products, traits, and resulting data.

A better variable name: BP_14_11 becomes **Mousse_Malt_Magic_Relaxing_0_10**.

Creating a Products File

The products and their indices were listed in the annotated survey, but this information was nowhere to be found in the actual data set.

I created my own short file called **products.csv**.

```
products <- fread(input = "products.csv")
datatable(data = products, rownames = FALSE)
```

Show 10 entries Search:

Number	Name
1	Cookie Crumble
2	Sweet Saltines
3	Fig Out!
4	Brownie Pops
5	Pretzelicious
6	PB and Jelly Beans
7	Caked On
8	Gummi Beans
9	Chip Strips
10	Choco Loco

Showing 1 to 10 of 23 entries Previous 1 2 3 Next

Creating a Traits File

I created a file called **brand perception traits.csv**.

```
bp.traits <- fread(input = "brand perception traits.csv")
datatable(data = bp.traits, rownames = FALSE)
```

Show 10 entries Search:

Number	Name
1	A brand for me.
2	Fits my budget.
3	Tastes great.
4	Good to share with others.
5	I like the logo.
6	For special occasions.
7	An everyday snack.
8	Healthy.
9	Delicious.
10	Just the right amount.

Showing 1 to 10 of 11 entries Previous 1 2 Next

Converting the Variables' Names

```
change.bp.variable.name <- function(the.names, products,
    bp.traits, prefix = "BP_") {
    require(plyr)
    require(data.table)

    short.names <- gsub(pattern = prefix, replacement = "",
        x = the.names)
    new.names <- character(length(short.names))
    the.pieces <- as.data.table(t(as.data.table(strsplit(x = short.names,
        split = "_", fixed = TRUE))))
    setnames(x = the.pieces, old = names(the.pieces), new = c("Trait",
        "Product"))

    the.trait <- mapvalues(x = the.pieces[, Trait], from = bp.traits[,
        Number], to = bp.traits[, Name], warn_missing = FALSE)

    the.product <- mapvalues(x = the.pieces[, Product],
        from = products[, Number], to = products[, Name],
        warn_missing = FALSE)

    new.names <- sprintf("%s_%s_0_10", the.product, the.trait)

    return(new.names)
}
```

fixed
logical. If TRUE match split exactly, otherwise use regular expressions

Displaying the New Variable Names

```
bp.variables <- names(dat)[grep(pattern = "BP_", x = names(dat))]
new.bp.names <- change.bp.variable.name(the.names = bp.variables,
    products = products, bp.traits = bp.traits)
print(new.bp.names[1:12])
```

可以在list的方框中直接加正则表达式

```
 [1] "Brownie Pops_A brand for me._0_10"
 [2] "Browniemint Bark_A brand for me._0_10"
 [3] "Caked On_A brand for me._0_10"
 [4] "Chip Strips_A brand for me._0_10"
 [5] "Chippy Cheese_A brand for me._0_10"
 [6] "Choco Loco_A brand for me._0_10"
 [7] "Cocoa Bears_A brand for me._0_10"
 [8] "Cookie Crumble_A brand for me._0_10"
 [9] "Fig Out!_A brand for me._0_10"
[10] "Frostipops_A brand for me._0_10"
[11] "Frozen Frogurt_A brand for me._0_10"
[12] "Gummi Beans_A brand for me._0_10"
```

Cleaning Up the New Variable Names

- We will remove special characters: ! . , etc.
- Spaces will be converted to underscores.

```
variable.name.cleanup <- function(x, removal.characters = c(",",
    ".", "!")) {
    for (i in 1:length(x)) {
        x[i] <- gsub(pattern = " ", replacement = "_", x = x[i],
            fixed = TRUE)
        for (j in 1:length(removal.characters)) {
            x[i] <- gsub(pattern = removal.characters[j],
                replacement = "", x = x[i], fixed = TRUE)
        }
    }
    return(x)
}
new.bp.names <- variable.name.cleanup(x = new.bp.names)
```

The Cleaned Up Names

```
print(new.bp.names[1:12])
```

```
 [1] "Brownie_Pops_A_brand_for_me_0_10"
 [2] "Browniemint_Bark_A_brand_for_me_0_10"
 [3] "Caked_On_A_brand_for_me_0_10"
 [4] "Chip_Strips_A_brand_for_me_0_10"
 [5] "Chippy_Cheese_A_brand_for_me_0_10"
 [6] "Choco_Loco_A_brand_for_me_0_10"
 [7] "Cocoa_Bears_A_brand_for_me_0_10"
 [8] "Cookie_Crumble_A_brand_for_me_0_10"
 [9] "Fig_Out_A_brand_for_me_0_10"
[10] "Frostipops_A_brand_for_me_0_10"
[11] "Frozen_Frogurt_A_brand_for_me_0_10"
[12] "Gummi_Beans_A_brand_for_me_0_10"
```

Changing the Names of the Variables in the data.table

```
setnames(x = dat, old = bp.variables, new = new.bp.names)
datatable(data = dat[1:5, .SD, .SDcols = grep(pattern = "Cookie_Crumble_",
    x = names(dat))], rownames = FALSE)
```

Show 10 entries Search:

Cookie_Crumble_A_brand_for_me_0_10	Cookie_Crumble_Fits_my_budget_0_10	Cookie_Crumble_Tastes_gr
6	5	0: Strongly Disagree
3	5	1
1	3	5
2	1	0: Strongly Disagree
2	1	6

Showing 1 to 5 of 5 entries Previous 1 Next

Examining A Brand Perception Variable

```
tab <- dat[, .N, keyby = "Cookie_Crumble_Tastes_great_0_10"]
datatable(data = tab, rownames = FALSE)
```

Show 10 entries Search:

Cookie_Crumble_Tastes_great_0_10	N
0: Strongly Disagree	31540
1	19144
10: Strongly Agree	2
2	19262
3	14728
4	8865
5	4265
6	1585
7	486
8	100

Showing 1 to 10 of 11 entries Previous 1 2 Next

Issue #2: Cleaning the Values of the Variables

The values of the Brand Perceptions of Cookie Crumble appear to be numeric. However, there are character values for:

- **0: Strongly Disagree**
- **10: Strongly Agree**

We want to place these values on a numeric scale to make them more amenable to analyses:

- Averaging
- As numeric variables in regression models

Numeric Conversions

```
convert.bp.to.numeric <- function(x){
  x[x == "0: Strongly Disagree"] <- "0"
  x[x == "10: Strongly Agree"] <- "10"

  return(as.numeric(x))
}
dat <- dat[, (new.bp.names) := lapply(X = .SD, FUN = "convert.bp.to.numeric"), .SDcols = new.bp.names]
```

Checking the Conversion

```
new.tab <- dat[, .N, keyby = "Cookie_Crumble_Tastes_great_0_10"]
datatable(data = new.tab, rownames = FALSE)
```

Show 10 entries Search:

Cookie_Crumble_Tastes_great_0_10	N
0	31540
1	19144
2	19262
3	14728
4	8865
5	4265
6	1585
7	486
8	100
9	23

Showing 1 to 10 of 11 entries Previous 1 2 Next

Issue #3: The Data's Peculiar Structure

The data covers many products and variables. There are separate columns for **each pair of a product and a product-specific trait**.

- **Awareness_1** is the awareness status for Product #1.
- **Consideration_2** is the consideration status for Product #2.
- **BP_1_4** is the fourth trait for the first product.

The Full Extent of the Problem.

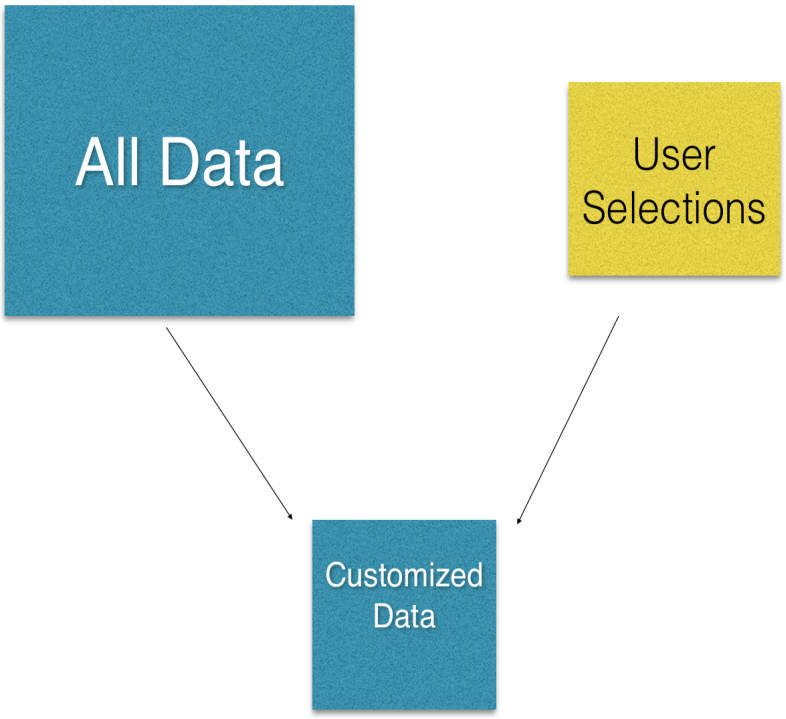
- Approximately 250 products and 500 measured variables per product.
- Approximately 20,000 rows and 50,000 columns of data. That's approximately 2.5 billion values!
- Worse yet, every variable had its own messes to clean up.

The ultimate goal of the project was to create analyses and models for each product, category, and subgroup, and to display everything in a **dynamic interface** that would allow the marketing team to access information easily and interactively. But, with this kind of data, how could we pull out the information they needed in a reasonable way?

What To Do?

- **Option 1**: Dynamic extraction.
- **Option 2**: Separate data sets for each product.
- **Option 3**: Restructuring the Data.

Option 1: Dynamic Extraction



Dynamic Extraction's Mechanics

```
datatable(data = dat[1:5], rownames = FALSE)
```

Show 10 entries Search:

id	age	gender	income	region	persona	Awareness_4	Awareness_17	Awareness_7	Awar
1	49	Male	57000	West	Millenial Muncher	1	0	1	
2	65	Male	133000	West	Righteous Reviewer	1	0	1	
3	18	Male	31000	West	Mainstream Maynard	1	0	0	
4	54	Female	85000	West	Mainstream Maynard	1	1	1	
5	33	Male	133000	West	Millenial Muncher	1	0	1	

Showing 1 to 5 of 5 entries Previous 1 Next

1. Select a brand.
2. Map the brand to its corresponding number.
3. Extract the appropriate columns of every brand-specific variable into a new data set.

Example of Dynamic Extraction

```
extract.brand.data <- function(dat, the.brand, products) {
    the.number <- products[Name == the.brand, Number]
    brand.dat <- dat[, .(id, age, Awareness = get(sprintf("Awareness_%d",
        the.number)), Consideration = get(sprintf("Consideration_%d",
        the.number)))]
    return(brand.dat)
}
cookie.crumble.dat <- extract.brand.data(dat = dat, the.brand = "Cookie Crumble",
    products = products)
datatable(data = cookie.crumble.dat[1:5, ], rownames = FALSE)
```

products是原始数据集，
the.brand是我所关心的
产品

Show 10 entries Search:

id	age	Awareness	Consideration
1	49	0	
2	65	0	
3	18	0	
4	54	1	0
5	33	0	

Showing 1 to 5 of 5 entries Previous 1 Next

Advantages of Dynamic Extraction

- One single data file.
- Requires the least pre-processing.
- Maintains one respondent in each row.

Disadvantages of Dynamic Extraction

There are a number of drawbacks to this approach:

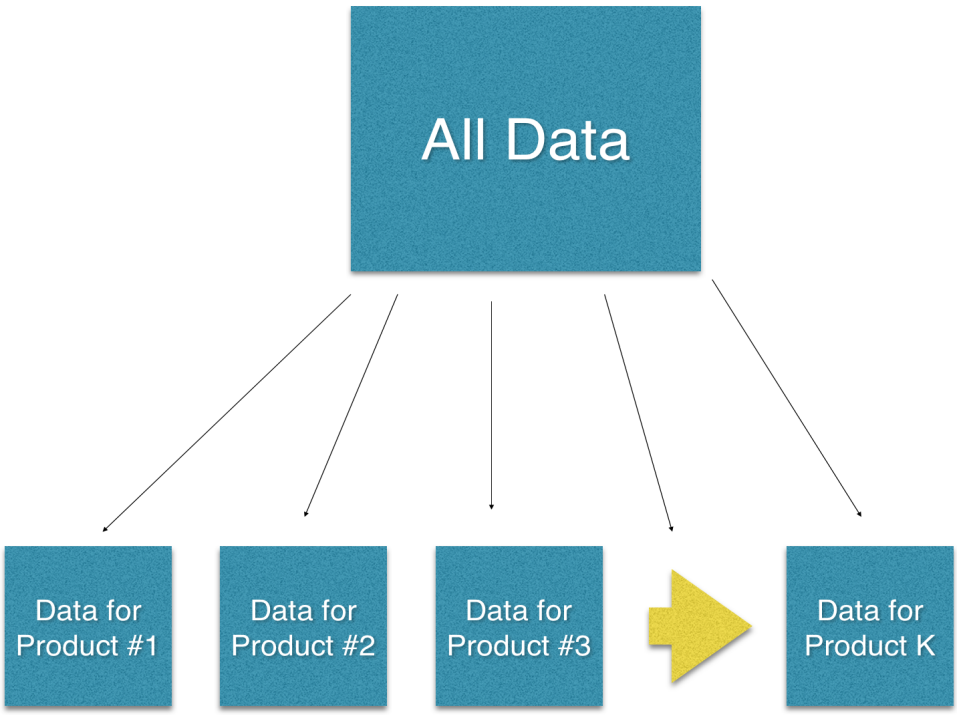
- All of the data **must be loaded each time**.
- Extraction is performed one column or group of columns at a time.
- May involve lots of processing to transform text variables dynamically.
- Not all products may have exactly the same variables.
- Aggregation across categories requires extractions and binding for each product.

For example, suppose you would like to aggregated all of the **Premium products**. This would involve dynamically extracting the data for **each individual product** in the Premium category and then binding all of these separate data sets into one larger one. That's a lot of processing every time you want to start an analysis!

That's Exactly What I Tried To Do

- The required processing was extremely time consuming.
- My reporting engine would take nearly a minute to calculate a simple graph or model.
- Any kind of aggregation, subsetting, or further customization was maximally difficult.
- Worse than that, all of these slowdowns meant that the numerous other problems with the data – cleaning issues, missing data, integrations from multiple sources, etc. – were also extremely challenging. Committing to this design made every problem more difficult to solve.

Option 2: Separate Data for Each Product



The Mechanics of Separating the Data

```
separate.product.data <- function(dat, products) {
    library(data.table)
    for (i in 1:products[, .N]) {
        product.dat <- extract.brand.data(data = dat, the.brand = products[i,
            Name], products = products)
        fwrite(x = product.dat, file = sprintf("%s Data.csv",
            products[i, Name]))
    }
}
```

Now we have the data for each individual product stored in separate CSV files.

Fundamentally, this approach requires **pre-processing** of the data. All of the clean-up, transformations, and extractions are done at an earlier stage. Then, for the modeling and reporting, only the data from the selected products would be loaded.

Advantages of Separate Data Sets

- Shifts the burden of dynamic processing to pre-processing.
- Smaller files to load – only the brand you need.
- More amenable to in-depth exploration of a single brand.
- Greater capability for customized models on each product.

Disadvantages of Separate Data Sets

Creating separate files can have a number of drawbacks:

- A lot of labor to separate all of the files.
- More difficult to generalize the approach across many products
- Aggregation might require loading many different sources of data.
- Small differences in which variables are included might generate unexpected errors.
- Much greater maintenance may be required – quarterly updates would have to be re-processed all over again, the engineering team would have many more quality checks, documentation, etc.

Option 3: Restructuring the Data

Mechanics of Restructuring

Starting with a data set that has **n** survey respondents and **k** products:

1. Classify your variables as **person-specific** or **brand-specific**.
2. Create a new data structure with $n * k$ rows.
3. Fill in the **person-specific** variables.
4. Aggregate the **brand-specific** columns into a single variable.

Fundamentally, the restructured form would change the **unit of observation** from a single person's responses for all of the products to that of a **single person's response to a single product**.

Step 1: Classifying the Variables

Identify all of the variables that are **person-specific**:

- Age
- Geography
- Income, Gender, Persona, etc.

Identify all of the variables that are **brand-specific**:

- Awareness of Brands 1, 2, 3, …
- Loyalty to Brands 1, 2, 3, …
- Brand Perception: Relaxing of Brands 1, 2, 3, …

Step 2: Create a New Data Structure

Create a new data structure with $n * k$ rows:

- Each row corresponds to the information **for a respondent and about a single brand**.
- Each column corresponds to a **single variable** measured across all pairs of people and brands. Instead of separate columns for Awareness in each product, there is only **one Awareness column** in this design.
- Each person ends up appearing in **k** different rows, while each brand appears in **n** different rows.

Before filling in the values of this new structure, it might look something like this:

Show 10 entries Search:

id	age	Product	Awareness	Consideration	Age
1	49				
2	65				
3	18				
1	49				
2	65				
3	18				

Showing 1 to 6 of 6 entries Previous 1 Next

Step 3: Fill in the Person-Specific Variables

- Each person's information is repeated for each product.
- The first three people's responses for the first two products would look like this:

Show 10 entries Search:

id	age	Product	Awareness	Consideration
1	49	Cookie Crumble		
2	65	Cookie Crumble		
3	18	Cookie Crumble		
1	49	Sweet Saltines		
2	65	Sweet Saltines		
3	18	Sweet Saltines		

Showing 1 to 6 of 6 entries Previous 1 Next

Step 4: Aggregate the Brand-Specific Variables

- Each fundamental trait (e.g. Awareness) will be organized into a single variable.
- The brand-specific variables for that trait will be combined into one vector:

Show 10 entries Search:

id	age	Product	Awareness	Consideration
1	49	Cookie Crumble	0	
2	65	Cookie Crumble	0	
3	18	Cookie Crumble	0	
1	49	Sweet Saltines	1	1
2	65	Sweet Saltines	1	1
3	18	Sweet Saltines	1	0

Showing 1 to 6 of 6 entries Previous 1 Next

Some Vocabulary for Restructuring Data Sets

- **Melting** is the process of turning wide data into long data.
- **Casting** is the process of turning long data into wide data.
- **Reshaping** is a term that goes both ways, either from wide to long or long to wide.

Melting Data – the Laborious Way

As an example, we will create new columns for Awareness and Consideration that combine the multiple columns across the first two products.

```
n <- dat[, .N]
k <- products[1:2, .N]
mdat <- data.table(id = rep(x = dat[, id], times = k), age = rep(x = dat[, age], times = k))
mdat[, Product := ""]
mdat[, Awareness := numeric(n)]
mdat[, Consideration := numeric(n)]
for(i in 1:k){
  mdat[(i-1)*n + 1:n, Product := rep(products[i, Name])]
  mdat[(i-1)*n + 1:n, Awareness := dat[, get(sprintf("Awareness_%d", products[i, Number]))]]
  mdat[(i-1)*n + 1:n, Consideration := dat[, get(sprintf("Consideration_%d", products[i, Number]))]]
}
```

创建三个空的新列

原来的一个值现在变成N个值

Melting Data – The Efficient Way

```r
id.vars <- c("id", "age")
measure.vars <- list(Awareness = sprintf("Awareness_%d", 1:k), Consideration = sprintf("Consideration_%d", 1:k))
mdat <- melt(data = dat, id.vars = id.vars, measure.vars = measure.vars, variable.name = "Product", value.name = c("Awareness",
  "Consideration"))
mdat[, Product := mapvalues(x = Product, from = products[, Number], to = products[, Name], warn_missing = FALSE)]
datatable(data = mdat[1:100], rownames = FALSE)
```

measure.vars为想转换的列

Show 10 entries Search:

id	age	Product	Awareness	Consideration
1	49	Cookie Crumble	0	
2	65	Cookie Crumble	0	
3	18	Cookie Crumble	0	
4	54	Cookie Crumble	1	0
5	33	Cookie Crumble	0	
6	64	Cookie Crumble	1	1
7	49	Cookie Crumble	0	
8	66	Cookie Crumble	0	
9	31	Cookie Crumble	1	1
10	81	Cookie Crumble	1	0

Showing 1 to 10 of 100 entries Previous 1 2 3 4 5 … 10 Next

Advantages of Melting a Wide Data Set

- The data are now in a version of **standard format** with the unit of observation set as the person's response about a specific product.
- **One overall column** for each trait.
- **Extraction and Aggregation** are now simple subsetting operations on the rows of the data set.
- Analyses are now straightforward, just like working with a data set in standard format.

Disadvantages of Melting

- The data set **may be even larger**; the number of additional rows may outpace the reduction of the columns.
- This still requires **loading all of the data**. However, the data can still be separated if needed, and this process is now easier because it's subsetting on the rows instead of extracting the columns.
- **Pre-processing is required**.

Example: Product Awareness

What percentage of respondents are aware of Cookie Crumble?

```
mdat[Product == "Cookie Crumble", sprintf("%.2f%%", 100 *
    mean(Awareness))]
```

```
[1] "75.68%"
```

This is considerably easier than mapping from a product name to its number, finding the corresponding column for Awareness, and then performing the computation. The melted structure of the data is more amenable to analysis.

Example: Product Consideration

What percentage of respondents have considered Cookie Crumble and Sweet Saltines in age groups of 18-34, 35-49, 50-65, and 65+?

Function like cut but left endpoints are inclusive and labels are of the form [lower, upper), except that last interval is [lower,upper]

```
library(Hmisc)
mdat[, `:=`(age_bracket, cut2(x = age, cuts = c(18, 35,
    50, 65, 120)))]
tab = mdat[, .(Consideration = round(x = 100 * mean(Consideration,
    na.rm = TRUE), digits = 2)), keyby = c("Product", "age_bracket")]
datatable(data = tab, rownames = FALSE)
```

Show 10 entries Search:

Product	age_bracket	Consideration
Cookie Crumble	[18, 35)	44.1
Cookie Crumble	[35, 50)	51.4
Cookie Crumble	[50, 65)	48.45
Cookie Crumble	[65,120]	57.79
Sweet Saltines	[18, 35)	67.2
Sweet Saltines	[35, 50)	61.74
Sweet Saltines	[50, 65)	68.73
Sweet Saltines	[65,120]	69.32

Showing 1 to 8 of 8 entries Previous 1 Next

Guidelines for Wide, Messy Data

For a data set with many products and separate columns for each product, melting the data can be advantageous. Here are some general recommendations:

- **Use the melt function** to reshape the data into long format. Other reshaping functions (e.g. reshape, gather) will not be as efficient.
- Use pre-processing to clean up the data as needed. Do this once per version of the data set.
- Then, for more dynamic applications (e.g. a web interface), directly load the pre-processed, melted data. Make any kind of application that will repeatedly use the data **as streamlined as possible**.

Customer Engagement

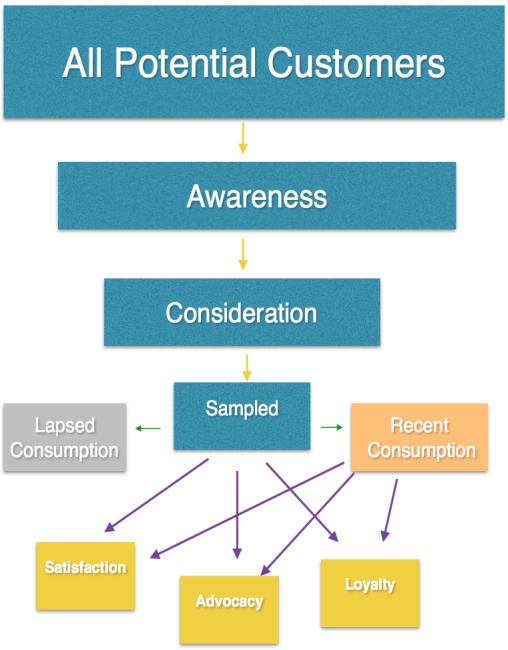
- The idea of the survey is to better understand the customers' engagement with the brands.
- We want to start with explorations of the data.
- Eventually, we want to understand the relationship between engagement and the other factors.

Hidden Biases

- Anyone who would take a 30 minute survey is probably not representative of the broader population.
- It is probably not reasonable to ask about over a hundred products at once.
- The responses are self-reported figures. These might be based on faulty memories about brands that the respondents sampled a long time ago.

If you are willing to accept the assumptions in your sampling methodology, then you can analyze the data to answer questions about your customers.

States of Engagement



A Progression

- Customers may or may not be aware of the product.
- Those who are not aware do not take the rest of the survey for that product.
- Those who are aware but have not considered the product do not proceed further.
- Those who have at least sampled the product are then asked all of the questions about the deeper states of engagement.

Awareness

“Have you ever heard of this product?”

- A respondent is aware of the brand if they have some prior knowledge of it.
- Awareness is the most basic state of engagement.
- Without awareness, no further degree of engagement is possible.

Consideration

“Would you consider using this product?”

- Some people who are aware of a brand have no desire to ever try it.
- Everyone else is at least willing to consider the brand.
- This leaves consideration in between an awareness of and the use of the product.

Consumption

This is a question of whether a respondent has used the product. Consumption is considered so important that the types of use are often categorized.

- **Ever Consumed** – Have you ever tried the product?
- **Lapsed Consumption** – Have you stopped using the product? Has it been more than 30 days since your last use of it?
- **Recent Consumption** – Have you used this product in the last 30 days?
- **Ongoing Consumption** – How often/regularly do you use this product?

Satisfaction

- A measure of whether or how much the respondent enjoys the product.
- "On a scale of 0-10, how satisfied are you with the product?"

This could also be a binary measure:

- **6-10**: Satisfied;
- **0-5**: Not Satisfied.

Loyalty

- A measure of whether the respondent continues to choose this product over the competition.
- A loyal customer is the surest sign of a good customer.
- Could also be measured in binary terms or on a continuous scale.

Advocacy

A measure of whether the respondent recommends the product to others – and how strongly.

I recommend the product:

- recommend it spontaneously, without being asked.
- recommend it, but only when I'm asked.
- neither recommend for or against this product.
- recommend **against** using this product, but only when I'm asked.
- recommend **against** using this product, without being asked.

Different combinations of selections could be used to measure:

- Weak Advocates;
- Strong Advocates;
- The degree of negative publicity around this product.

Businesses Seek Out Advocates

- Advocates provide word-of-mouth advertising, which is often the most effective.
- Advocates create **multiplicative** effects on sales.
- They do the work of a business… for free!

Subgroup Analyses

- Marketing teams want to know who is and is not a good customer.
- Combinations of age groups, geographic regions, income levels, and other high level factors are worth investigating.
- Goals: Confirm intuitions and discover new information.

Creating Personas for Customers

- An exercise in creating a shorthand for different types of customers.
- Illustrative, alliterative names are often used.
- The idea is to convey an image of a full personality with a single phrase.

Clustering the Customers

- Some survey questions were used to evaluate the respondents' willingness to try new things and use of technology.
- The responses to these questions were scored and compared to each other.
- Clustering was used to place the respondents into groups.

Assigning Personas to Respondents

The Selected Personas for Snack Food Customers

- **Millenial Munchers**: young people who are addicted to their snacks as much as their phones.
- **Righteous Reviewers**: Very adamant about their likes and dislikes, and they love to tell others about it online.
- **Old School Olivers**: They like what they like, and they're not going to change.
- **Savvy Samanthas**: They're always ahead of the curve, looking for the best new products and deals.
- **Easygoing Ediths**: Social and relaxed, they like just about everything they try.
- **Mainstream Maynards**: They tend to like what they see on TV and what others around them recommend.

Customer Engagement by Persona

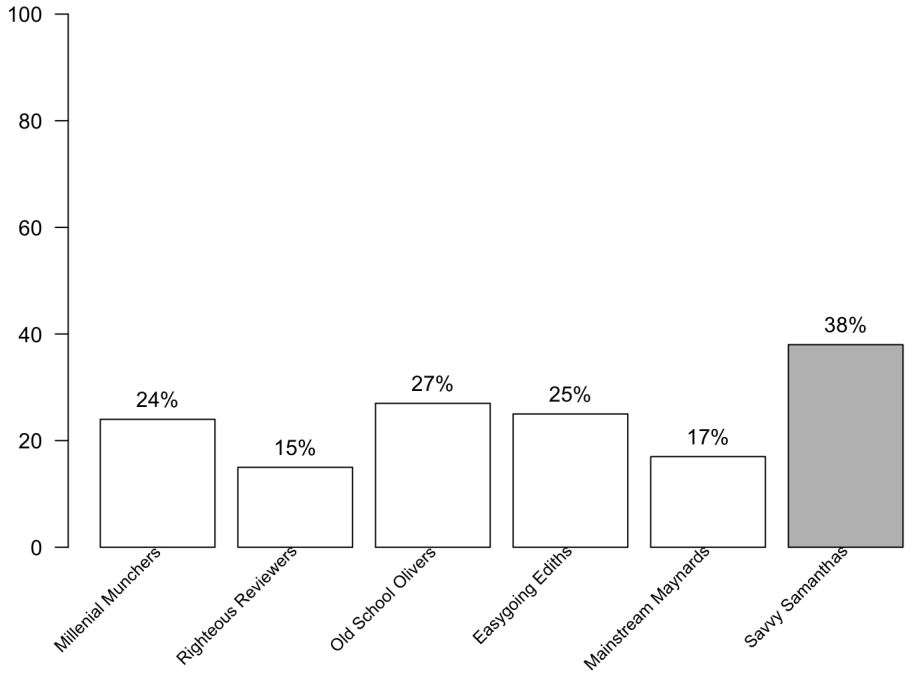
- For any product, we can estimate the rates of engagement using the survey data.
- For each product and state of engagement, we can compare the results for different personas.
- Through exploration, we can seek out new insights.

The Story of Cookie Crumble

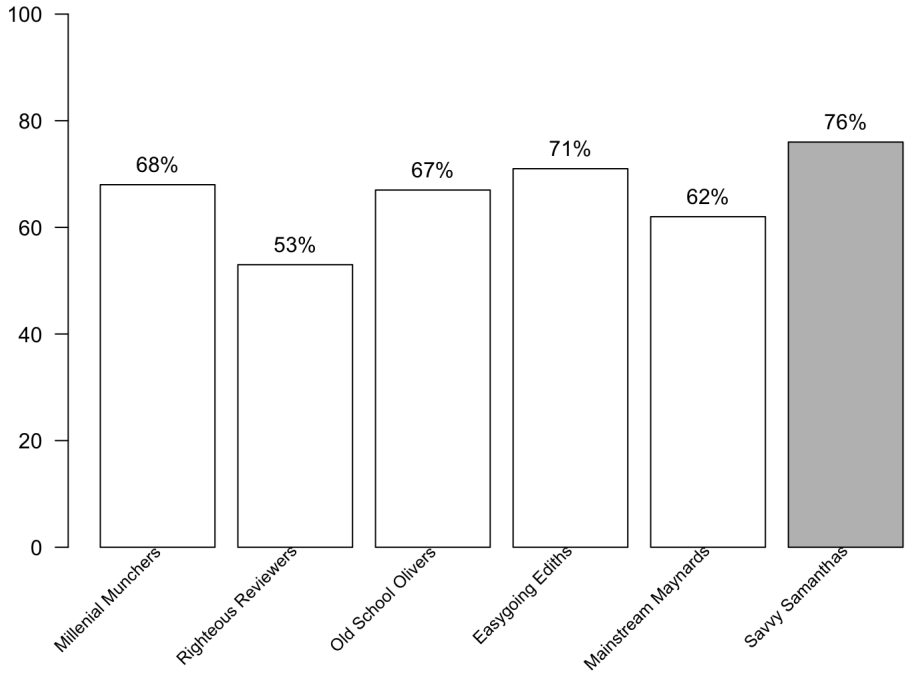
The following story is **based on the real data** that I saw – rather than what was simulated for the file used in this lecture.

- Cookie Crumble is the client's **flagship product**.
- The product is **well-known and a bestseller** in the industry.
- The product is also the most highly studied within the company. **Any new insights would be surprising** – and quickly prove the value of using data science to the marketing team. Here is what we found:

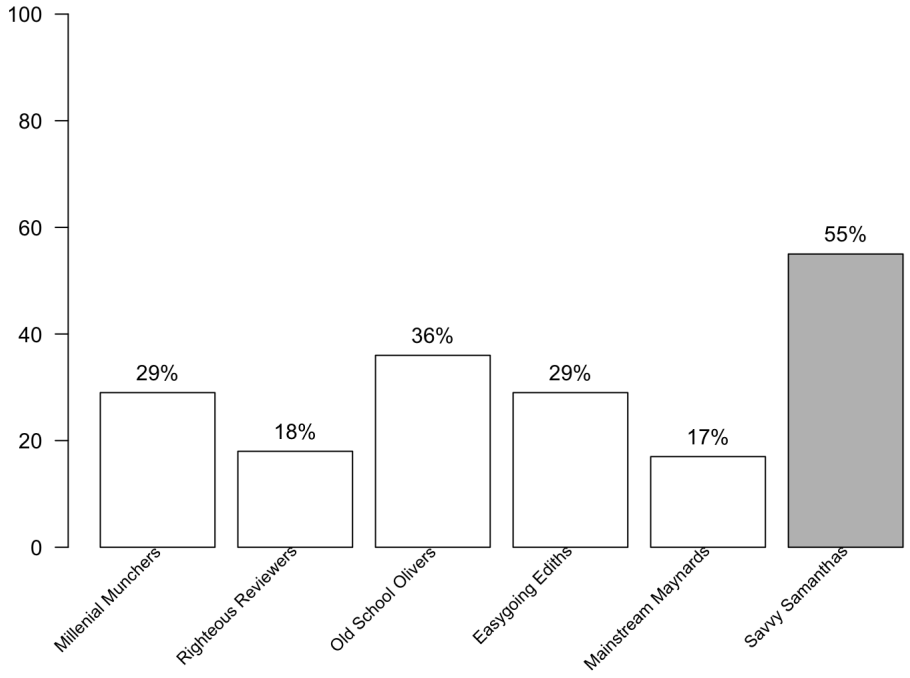
Recent Consumption by Persona



Satisfaction by Persona



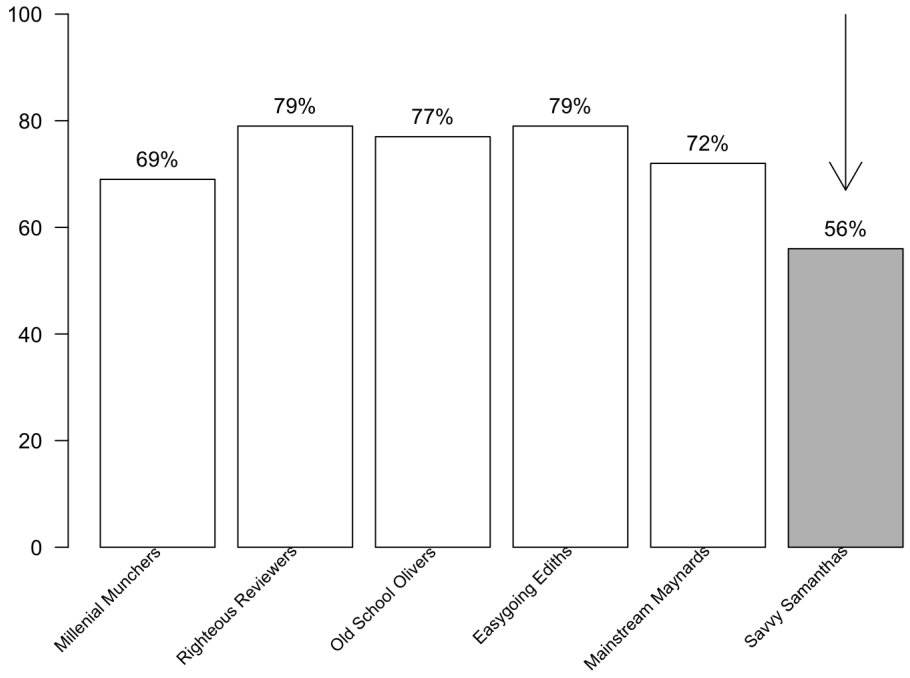
Advocacy by Persona



The Best Customers

- Savvy Samanthas have considerably higher rates in these categories and are competitive in many others.
- Given their persona, these results match up with our intuitions. Savvy Samanthas do lots of research and are proud of consuming the most hip, interesting products.
- So imagine our surprise when we saw the next graph…

Awareness by Persona



A Major Finding

- Savvy Samanthas have by far the worst rate of awareness of Cookie Crumble of any group.
- All other evidence suggests that Savvy Samanthas are the best customers among those who are aware of the product.
- **An obvious recommendation:** the client had the most to gain by making Savvy Samanthas aware of the product. So spend more of the advertising budget on reaching them!

Or So We Thought

- Interestingly enough, the marketing team we worked with had almost no reaction to this new information.
- The marketing team was much more concerned with its primary goals of building dynamic tools that would enable explorations across a much wider range of factors.
- Due to the constraints of time on a project with a limited scope and budget, the team never followed up on this information.
- In this way, perhaps the most interesting results that came out of the project – insights on a high-volume brand with real implications for strategy and sales – **were completely ignored.**

Checking Other Products

- The same pattern seemed to appear in many other products we checked.
- The Savvy Samanthas may be sophisticated about the products they know, but this doesn't mean that they have greater awareness of what's out there.
- However, this does raise some questions:

How Can We Resolve This Inconsistency?

- Perhaps our metrics that informed the Personas (trying new things and technical savvy) are not so predictive after all.
- Perhaps the Personas are unreasonable caricatures of human behavior.
- Perhaps the Personas are reasonable, but Savvy Samanthas concentrate their energy on the products they like at the expense of trying out new things.
- … Or perhaps this story illustrates the limitations of working with surveys.

More to Come

With hundreds of products and thousands of variables, we have only begun to dig into the kinds of insights we can produce by examining surveyed data.