

Marketing Analytics

**Technology: Statistical Analysis Software & R
Conjoint Analysis Using R**

PREREQUISITE: R BASICS

Stephan Sorger

www.stephansorger.com

Disclaimer:

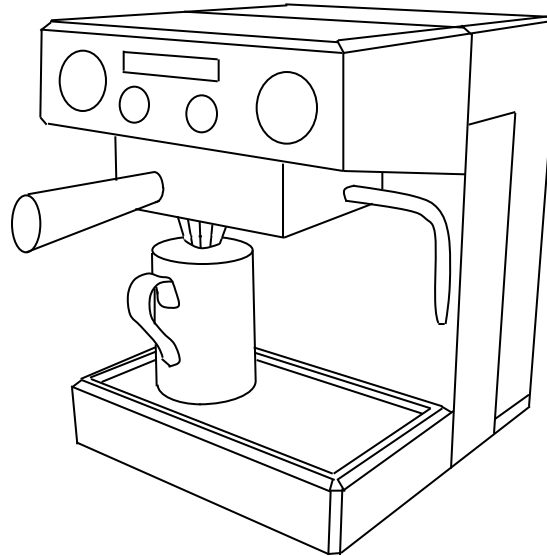
- All images such as logos, photos, etc. used in this presentation are the property of their respective copyright owners and are used here for educational purposes only
- Some material adapted from: Sorger, “Marketing Analytics: Strategic Models and Metrics”

Outline: 10 Steps to Conjoint Analysis

Topic	Discussion
1. Attribute Selection	Select relevant product characteristics
2. Attribute Level Selection	Identify levels of characteristics
3. Product Bundle Formation	Create sample products by bundling
4. Data Collection Method Selection	Decide how to collect data
5. Data Collection	Ask responders to rate products
6. Data Preparation	Get data ready for conjoint
7. Importance Weights Calculation	Assess importance of each attribute
8. Part-Worth Calculation	Assess importance of each level
9. Market Share Estimation	Predict preferences of products
10. Data Interpretation	Compare against industry averages

Conjoint Analysis Example

Topic	Discussion
Acme Espresso	You are the marketing manager for Acme Espresso Machine Small manufacturer of premium coffee makers Competitors: Breville, DeLonghi, Gaggia, Rancilio



Conjoint Analysis Example

Topic	Discussion
Objectives	Understand how consumers assess different products Identify specific features desired by consumers Predict market share for different product variants
Conjoint	Conduct conjoint analysis to address objectives Use 10-step process

Steps 1 - 5

Review

Steps 6 - 10

Execute using R

Step 1: Attribute Selection

Topic	Discussion
Attributes	Characteristics that consumers find relevant when evaluating
Research	Secondary: Product reviews, etc. Primary: Customer surveys

Speed

Brewing time, in minutes

Capacity

Number of cups machine can brew

Price

Retail price of espresso machine

Step 2: Attribute Level Selection

Topic	Discussion
Levels	Identify levels of attributes where consumers see differences
Two Levels	In this exercise, we will identify two levels for each attributes

Speed

S1: "Fast": Under 1 minute

S2: "Slow": 1 minute or more

Capacity

C1: "Single-Cup": 1 cup at a time

C2: "Multi-Cup": Multiple cups at a time

Price

P1: "Budget": Price under \$300

P2: "Premium": \$300 or more

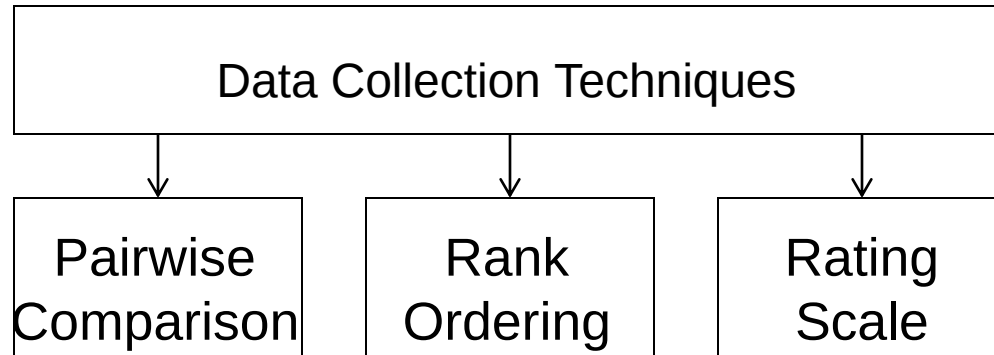
Step 3: Product Bundle Formation

Topic	Discussion
Bundles	Create candidate products by bundling together attributes Vary the levels of the attributes Bundles often called cards
Quantity	$(\text{Levels})^{\text{Attributes}}$; $2^3 = 8$

<u>Card/Bundle</u>	<u>Speed (S)</u>	<u>Capacity (C)</u>	<u>Price (P)</u>
1	1	1	1
2	1	1	2
3	1	2	1
4	1	2	2
5	2	1	1
6	2	1	2
7	2	2	1
8	2	2	2

Step 4: Data Collection Method Selection

Topic	Discussion
Pairwise Comparison	Respondents compare pairs of options
Rank Ordering	Respondents place options in rank order: 1 – 100
Rating Scale*	Respondents rate each option independently



Step 5: Data Collection

Topic	Discussion
Rating Scale	Ask respondents to rate each card on a scale from 1 to 5 1 = poor; 5 = outstanding

<u>Card/Bundle</u>	<u>Speed (S)</u>	<u>Capacity (C)</u>	<u>Price (P)</u>	<u>Preference</u>
1	1	1	1	4
2	1	1	2	3
3	1	2	1	5
4	1	2	2	4
5	2	1	1	3
6	2	1	2	1
7	2	2	1	3
8	2	2	2	2

Step 6: Data Preparation

Topic	Discussion
Data	Prepare the data for use with R Same data we used for Microsoft Excel approach

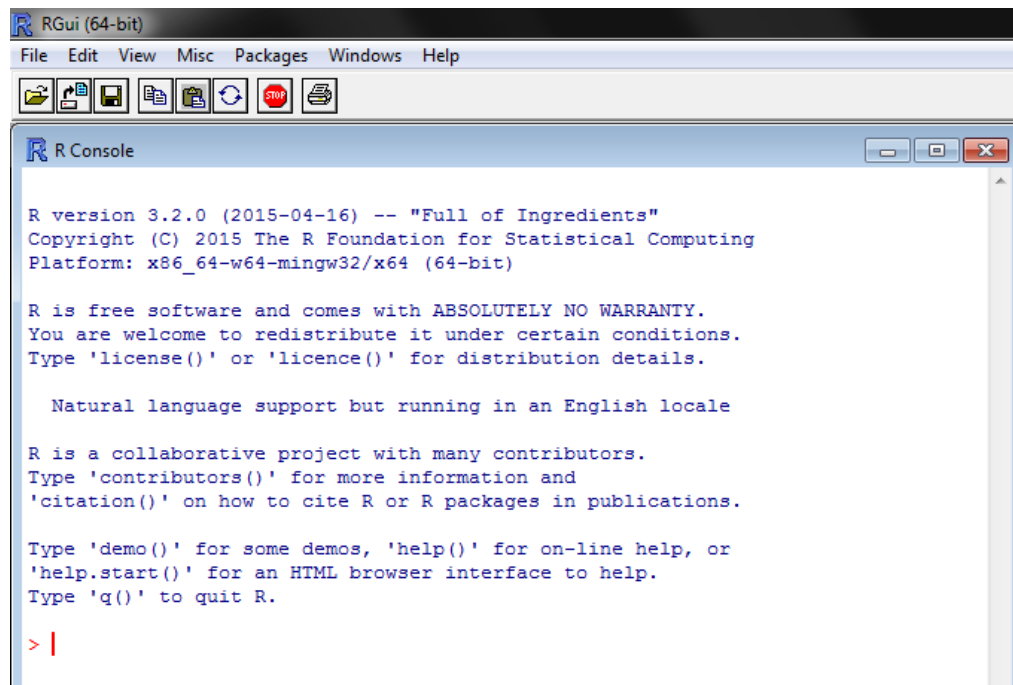
<u>Step</u>	<u>Data Preparation</u>
A	Download R
B	Launch R
C	Install the R conjoint package
D	Load the conjoint library
E	Convert data set from Excel-friendly format to R-friendly format
F	Load the data into R

Step 6: Data Preparation; A: Download R

Platform	Link
Windows	http://cran.r-project.org/bin/windows/base/
Mac	http://cran.r-project.org/bin/macosx/

Step 6: Data Preparation; B: Launch R

Topic	Discussion
Prompt	You will see a “>” prompt in the “R Console”



```
RGui (64-bit)
File Edit View Misc Packages Windows Help

R Console

R version 3.2.0 (2015-04-16) -- "Full of Ingredients"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

You will be typing commands at the prompt: “>”

Step 6: Data Preparation; C: Install R Conjoint Package

Topic	Discussion
Packages	Need special functionality to execute our conjoint analysis R extends it functionality through functions called packages (1)
Conjoint	Package called “conjoint” for conjoint analysis (2)

Syntax: `install.packages(“packagename”)`

Our case: `install.packages(“conjoint”)`

(See next few slides for screenshots)

(1) R Documentation, “Install Packages from Repositories

<http://stat.ethz.ch/R-manual/R-devel/library/utils/html/install.packages.html>

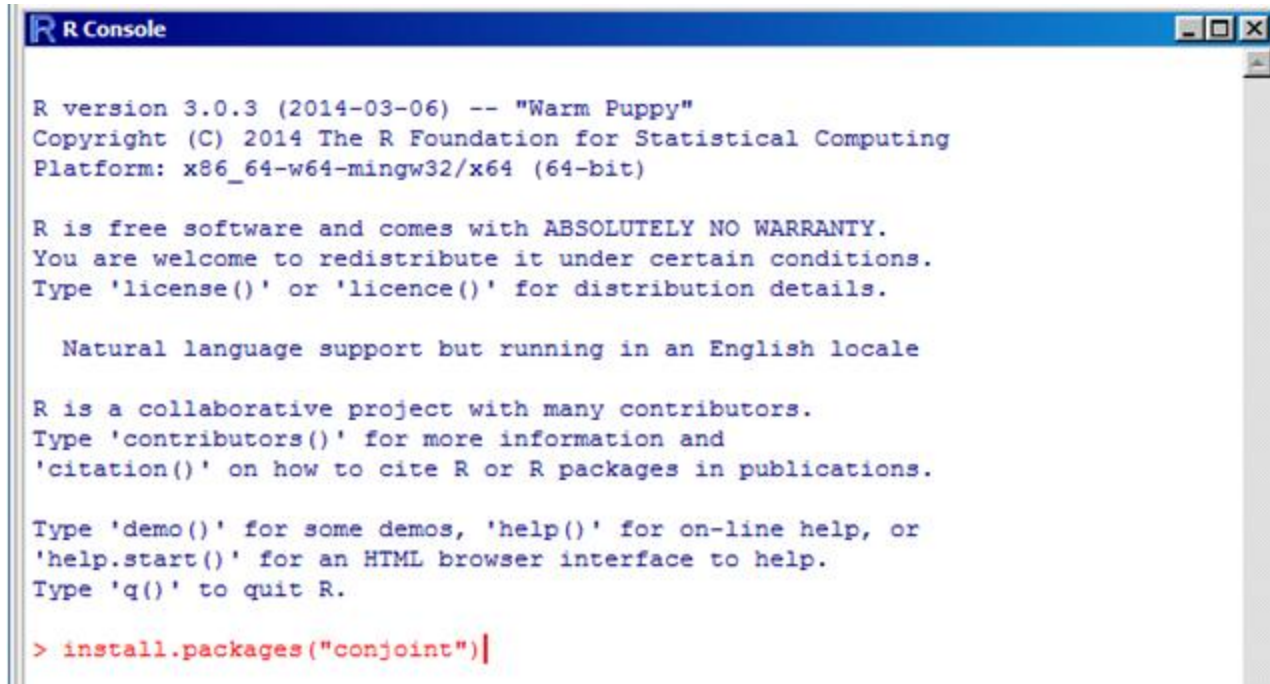
(2) R Documentation, Package “Conjoint”. August 29, 2013. Bak, Andrej and Bartlomowicz

<http://cran.r-project.org/web/packages/conjoint/conjoint.pdf>

Step 6: Data Preparation; C: Install R Conjoint Package

Topic	Discussion
-------	------------

Screenshot	<code>install.packages("conjoint")</code>
------------	---



```
R Console

R version 3.0.3 (2014-03-06) -- "Warm Puppy"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

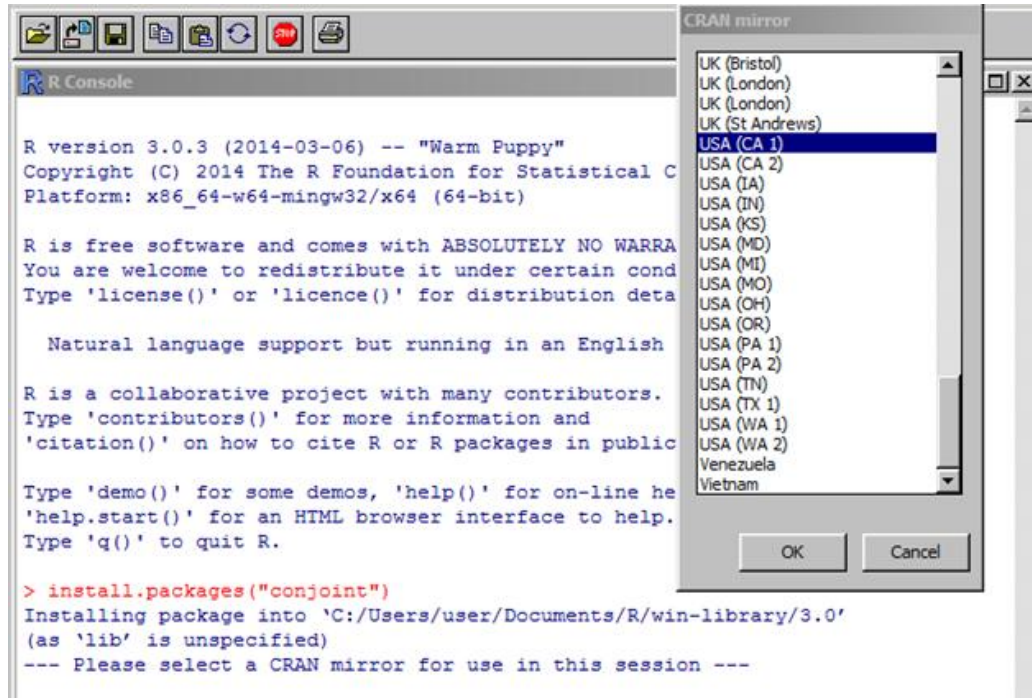
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> install.packages("conjoint")
```

Step 6: Data Preparation; C: Install R Conjoint Package

Topic	Discussion
-------	------------

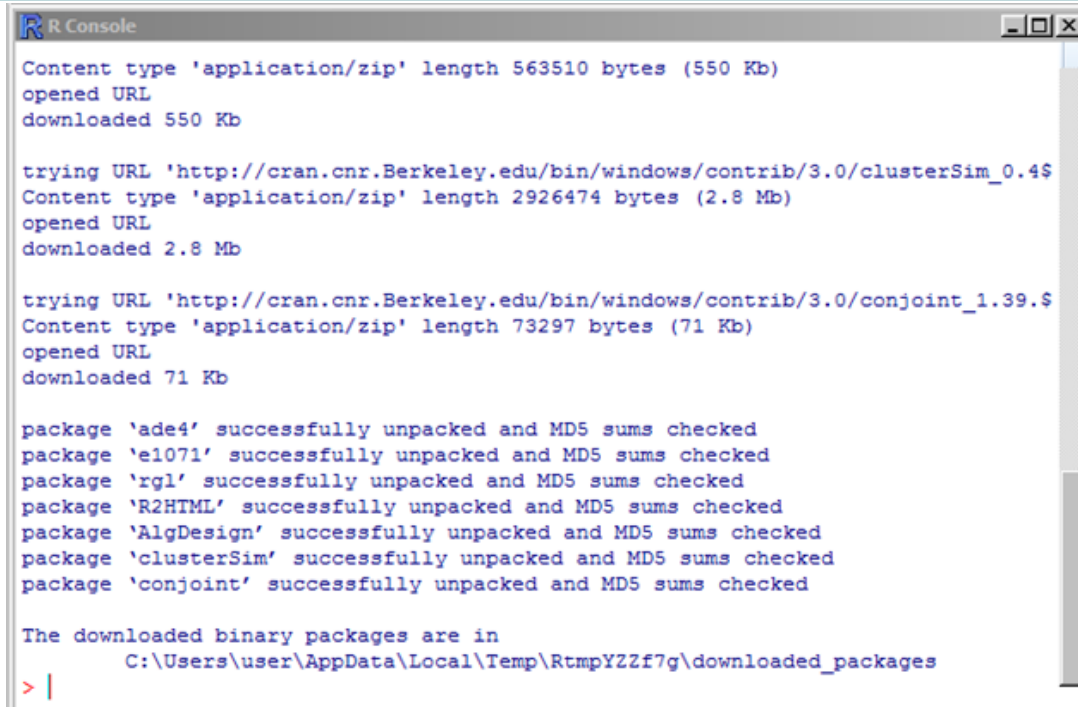
Screenshot	Downloading multiple files in package
------------	---------------------------------------



Step 6: Data Preparation; C: Install R Conjoint Package

Topic	Discussion
-------	------------

Screenshot	Package elements unpacked successfully
------------	--

A screenshot of an R Console window. The window title is 'R Console'. The text inside shows the process of downloading and installing several R packages from CRAN. It lists the content type and size for each package, followed by the URL being tried, the content type and size again, and then the download status. Finally, it lists the packages that were successfully unpacked and had their MD5 sums checked. The packages are: 'ade4', 'e1071', 'rgl', 'R2HTML', 'AlgDesign', 'clusterSim', and 'conjoint'. The console also shows the path where the downloaded binary packages are stored: 'C:\Users\user\AppData\Local\Temp\RtmpYZzf7g\downloaded_packages'. The prompt '> |' is visible at the bottom.

```
R Console

Content type 'application/zip' length 563510 bytes (550 Kb)
opened URL
downloaded 550 Kb

trying URL 'http://cran.cnr.Berkeley.edu/bin/windows/contrib/3.0/clusterSim_0.4$
Content type 'application/zip' length 2926474 bytes (2.8 Mb)
opened URL
downloaded 2.8 Mb

trying URL 'http://cran.cnr.Berkeley.edu/bin/windows/contrib/3.0/conjoint_1.39.$
Content type 'application/zip' length 73297 bytes (71 Kb)
opened URL
downloaded 71 Kb

package 'ade4' successfully unpacked and MD5 sums checked
package 'e1071' successfully unpacked and MD5 sums checked
package 'rgl' successfully unpacked and MD5 sums checked
package 'R2HTML' successfully unpacked and MD5 sums checked
package 'AlgDesign' successfully unpacked and MD5 sums checked
package 'clusterSim' successfully unpacked and MD5 sums checked
package 'conjoint' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
  C:\Users\user\AppData\Local\Temp\RtmpYZzf7g\downloaded_packages
> |
```

Step 6: Data Preparation; D: Load Conjoint Library

Topic	Discussion
Load files	library(conjoint)

Step 6: Data Preparation; E: Convert Data Set to R Format

Topic	Discussion
-------	------------

Preferences	Matrix dedicated to preferences stated by respondents
-------------	---

Card/Bundle	Speed (S)	Capacity (C)	Price (P)	Preference
1	1	1	1	4
2	1	1	2	3
3	1	2	1	5
4	1	2	2	4
5	2	1	1	3
6	2	1	2	1
7	2	2	1	3
8	2	2	2	2



	A	B	C	D	E	F	G	H
1	card1	card2	card3	card4	card5	card6	card7	card8
2	4	3	5	4	3	1	3	2

Step 6: Data Preparation; E: Convert Data Set to R Format

Topic	Discussion
-------	------------

Profiles	Matrix dedicated to description of data structure
----------	---

<u>Card/Bundle</u>	<u>Speed (S)</u>	<u>Capacity (C)</u>	<u>Price (P)</u>
1	1	1	1
2	1	1	2
3	1	2	1
4	1	2	2
5	2	1	1
6	2	1	2
7	2	2	1
8	2	2	2

Step 6: Data Preparation; E: Convert Data Set to R Format

Topic	Discussion
-------	------------

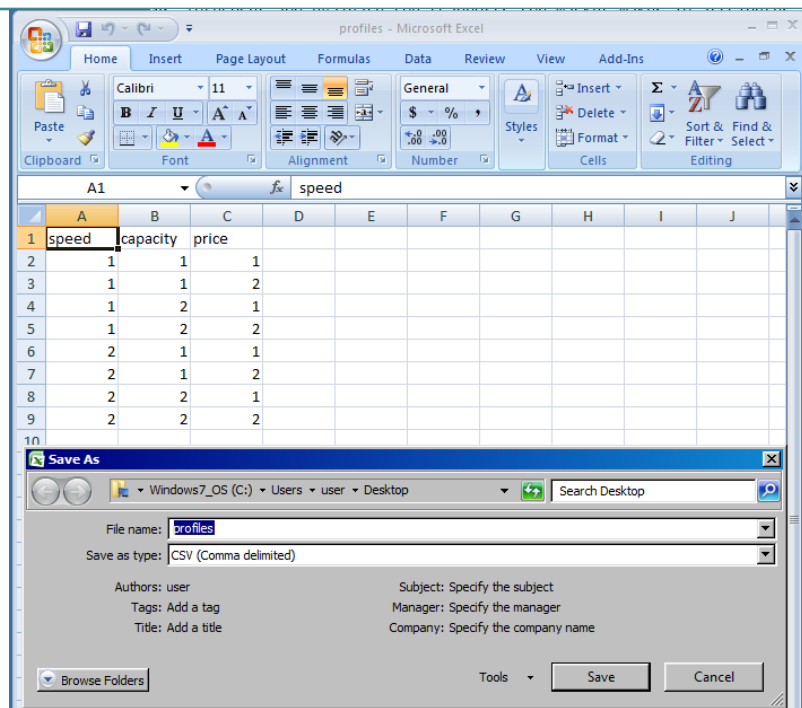
Profiles	Matrix dedicated to description of data structure
----------	---

	A	B	C
1	speed	capacity	price
2	1	1	1
3	1	1	2
4	1	2	1
5	1	2	2
6	2	1	1
7	2	1	2
8	2	2	1
9	2	2	2

Step 6: Data Preparation; E: Convert Data Set to R Format

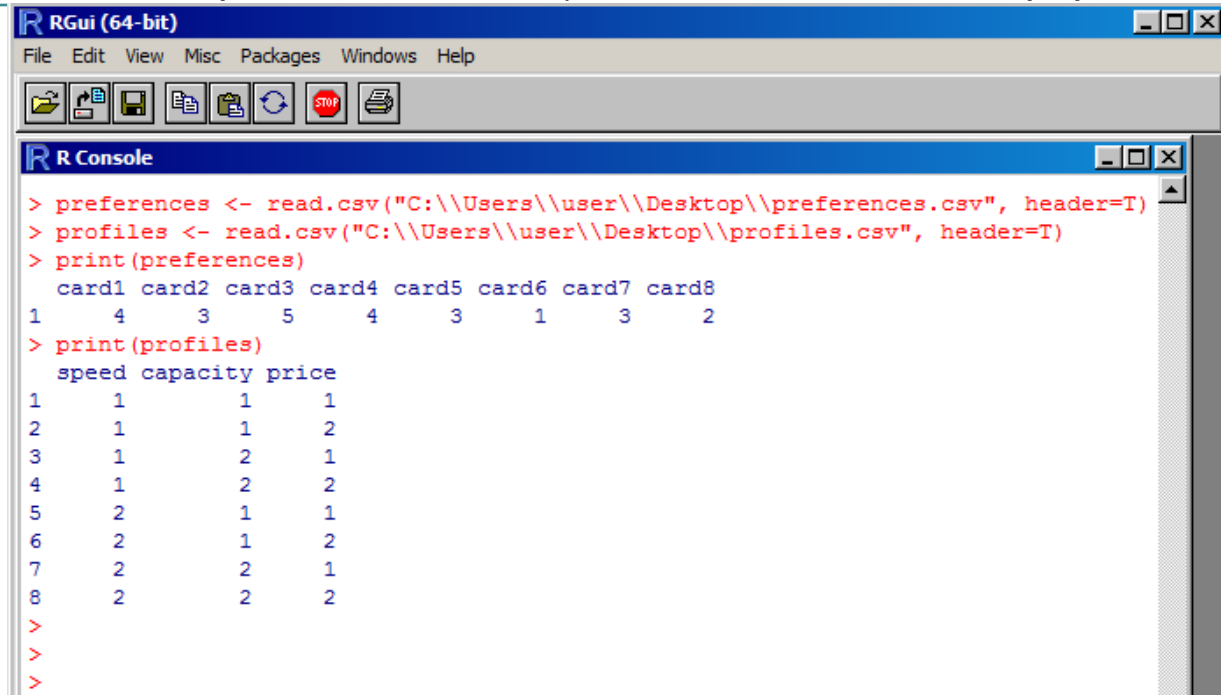
Topic Discussion

Profiles Save As CSV



Step 6: Data Preparation; F: Load Datasets Into R

Topic	Discussion
Preferences	<code>preferences<-read.csv("C:\\Users\\user\\Desktop\\preferences.csv", header=T)</code>
Profiles	<code>profiles<-read.csv("C:\\Users\\user\\Desktop\\profiles.csv", header=T)</code>



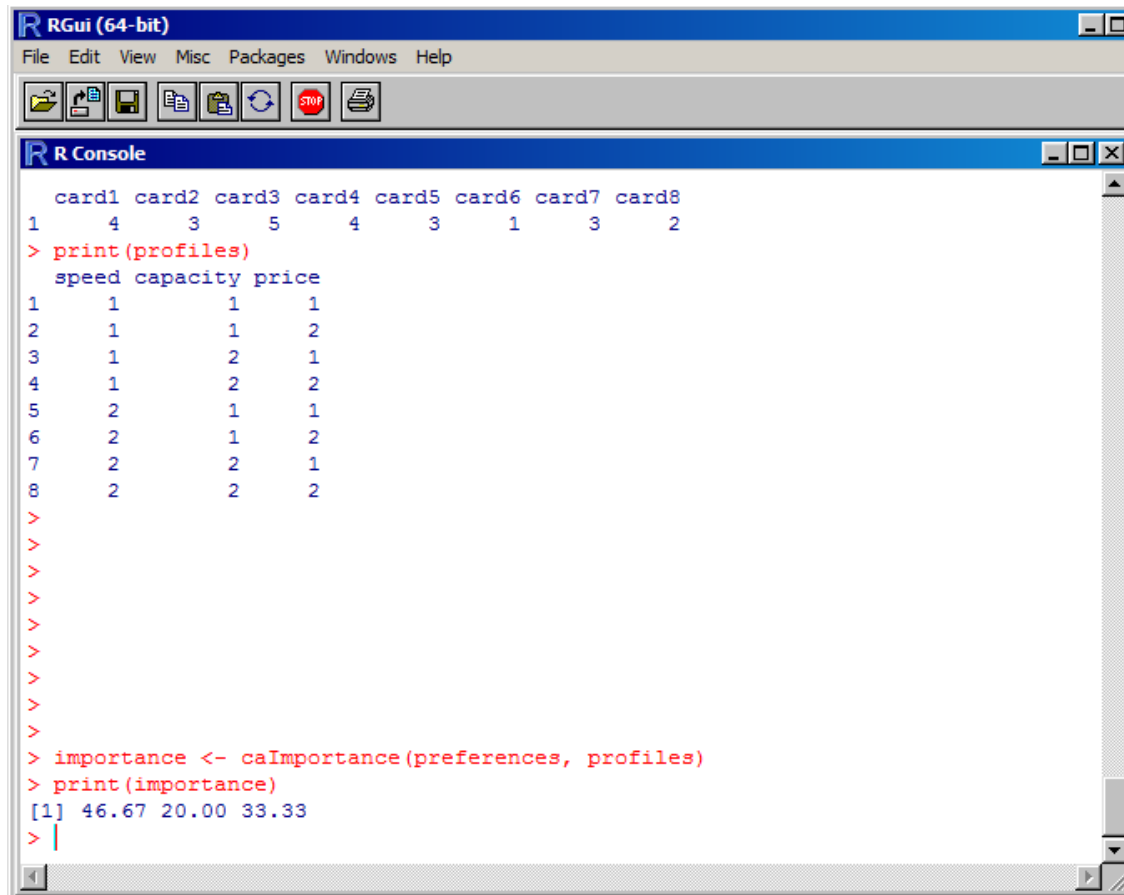
The screenshot shows the RGui (64-bit) window with the R Console. The console displays the following code and output:

```
> preferences <- read.csv("C:\\Users\\user\\Desktop\\preferences.csv", header=T)
> profiles <- read.csv("C:\\Users\\user\\Desktop\\profiles.csv", header=T)
> print(preferences)
  card1 card2 card3 card4 card5 card6 card7 card8
1     4     3     5     4     3     1     3     2
> print(profiles)
  speed capacity price
1     1         1     1
2     1         1     2
3     1         2     1
4     1         2     2
5     2         1     1
6     2         1     2
7     2         2     1
8     2         2     2
>
>
>
```

Step 7: Importance Weights Calculation

Topic	Discussion
Importance	Calculate importance weights in conjoint Refers to importance of each attribute, such as speed
calImportance	R command built into the conjoint analysis (ca) package
Syntax	calImportance(preferences, profiles) calImportance = conjoint analysis function for importance preferences = matrix with preferences for each card profiles = matrix describing each profile or card
Command	importance <- calImportance (preferences, profiles)

Step 7: Importance Weights Calculation



The screenshot shows the RGui (64-bit) window with the R Console. The console contains the following code and output:

```
card1 card2 card3 card4 card5 card6 card7 card8
1      4      3      5      4      3      1      3      2
> print(profiles)
speed capacity price
1      1          1      1
2      1          1      2
3      1          2      1
4      1          2      2
5      2          1      1
6      2          1      2
7      2          2      1
8      2          2      2
>
>
>
>
>
>
>
>
> importance <- caImportance(preferences, profiles)
> print(importance)
[1] 46.67 20.00 33.33
> |
```

Step 7: Importance Weights Calculation

Topic	Discussion
Comparison	We compare the Excel results with the R results

Variable	Excel Coefficient	% of Total	R Results
Speed	1.75	$1.75/3.75=46.67\%$	46.67%
Capacity	0.75: absolute value	$0.75/3.75=20.00\%$	20.00%
Price	<u>1.25</u>	<u>$1.25/3.75=33.33\%$</u>	<u>33.33%</u>
Total	3.75	$3.75/3.75=100\%$	100%

Step 8: Part-Worth Calculation: Levelnames Matrix

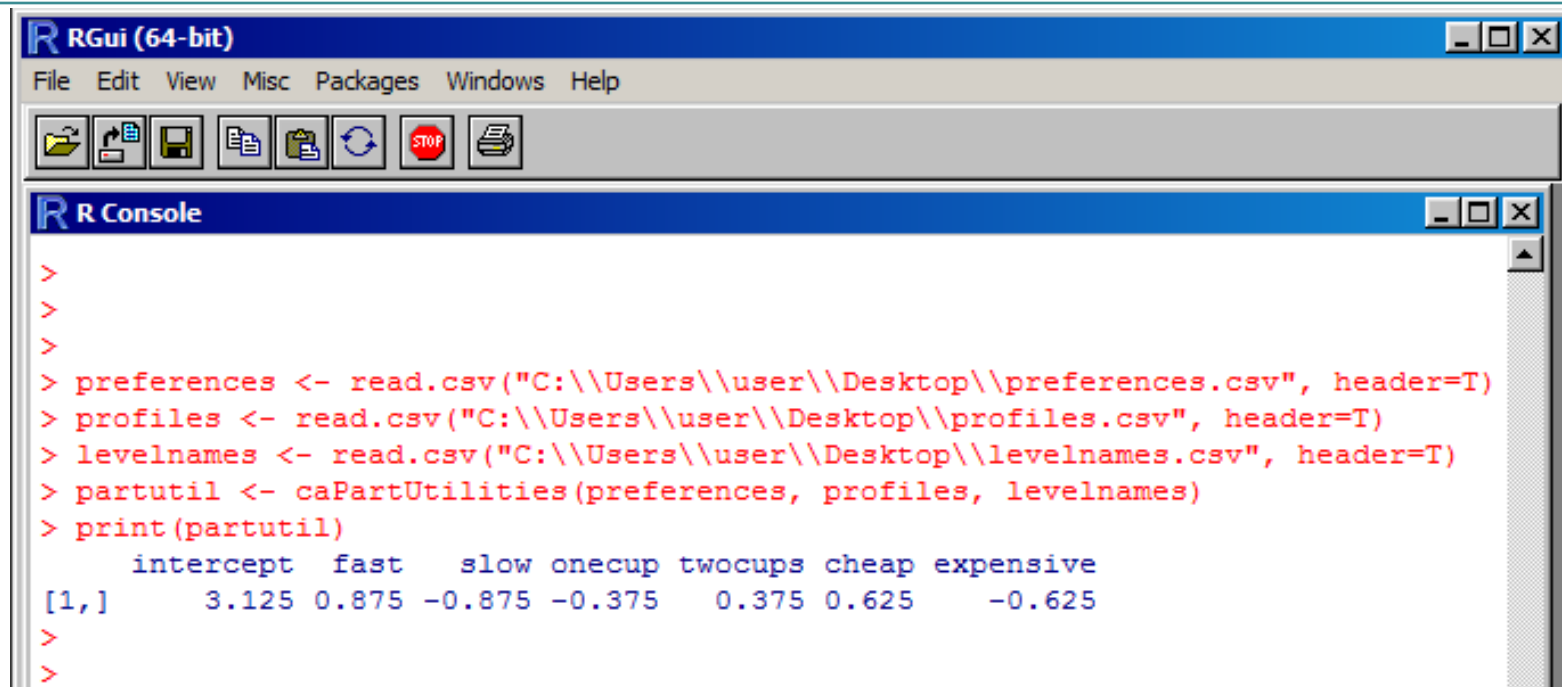
Topic	Discussion
Levelnames	Matrix to capture names of levels for each attribute

Speed: fast, slow
Capacity: onecup, twocup
Price: cheap, expensive

	A	B	C	D
1	levels			
2	fast			
3	slow			
4	onecup			
5	twocup			
6	cheap			
7	expensive			

Step 8: Part-Worth Calculation: Calculate Part Worths

Topic	Discussion
Read data Command	<pre>levelnames<-read.csv("C:\\Users\\user\\Desktop\\levelnames.csv", header=T) partutil <- caPartUtilities (preferences, profiles, levelnames)</pre>

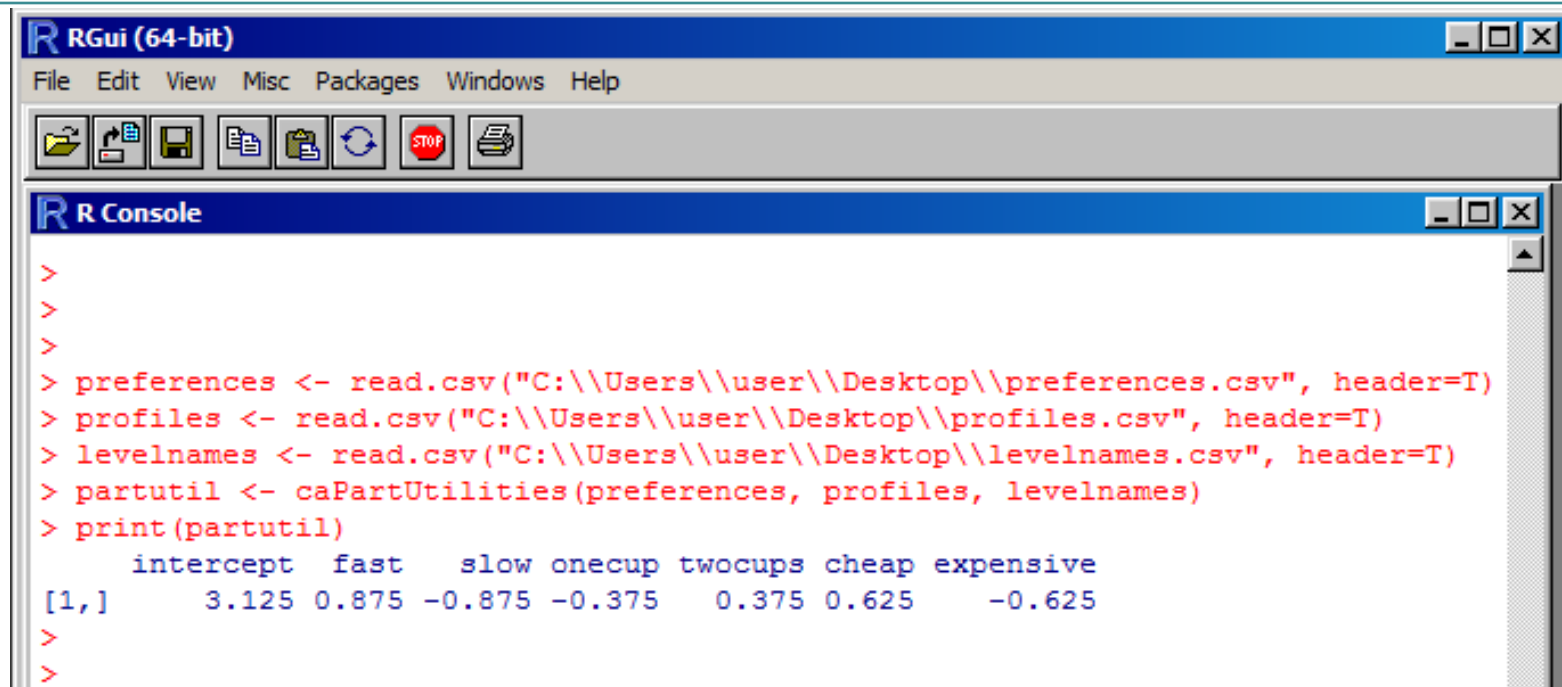


The screenshot shows the RGui (64-bit) window. The menu bar includes File, Edit, View, Misc, Packages, Windows, and Help. The toolbar contains icons for file operations and execution. The R Console window displays the following code and output:

```
>  
>  
>  
> preferences <- read.csv("C:\\Users\\user\\Desktop\\preferences.csv", header=T)  
> profiles <- read.csv("C:\\Users\\user\\Desktop\\profiles.csv", header=T)  
> levelnames <- read.csv("C:\\Users\\user\\Desktop\\levelnames.csv", header=T)  
> partutil <- caPartUtilities(preferences, profiles, levelnames)  
> print(partutil)  
      intercept  fast  slow onecup twocups cheap expensive  
[1,]      3.125 0.875 -0.875 -0.375  0.375 0.625   -0.625  
>  
>
```

Step 8: Part-Worth Calculation: Examine Part Worths

Topic	Discussion						
Part Worths	intercept	fast	slow	onecup	twocups	cheap	expensive
	3.125	0.875	-0.875	-0.375	0.375	0.625	-0.625



```
RGui (64-bit)
File Edit View Misc Packages Windows Help

R Console
>
>
>
> preferences <- read.csv("C:\\Users\\user\\Desktop\\preferences.csv", header=T)
> profiles <- read.csv("C:\\Users\\user\\Desktop\\profiles.csv", header=T)
> levelnames <- read.csv("C:\\Users\\user\\Desktop\\levelnames.csv", header=T)
> partutil <- caPartUtilities(preferences, profiles, levelnames)
> print(partutil)
      intercept  fast  slow onecup twocups cheap expensive
[1,]    3.125 0.875 -0.875 -0.375  0.375 0.625   -0.625
>
>
```

Step 8: Part-Worth Calculation: Interpret Part Worths

Topic	Discussion						
Part Worths	intercept	fast	slow	onecup	twocups	cheap	expensive
	3.125	0.875	-0.875	-0.375	0.375	0.625	-0.625

Preference = Intercept + (PW_fast) + (PW_slow) + (PW_onecup) + (PW_twocups) + (PW_cheap) + (PW_expensive)

Preference = 3.125 + 0.875 * fast - 0.875 * slow - 0.375 * onecup + 0.375 * twocups + 0.625 * cheap - 0.625 * expensive

Test machine 1: fast, twocup, cheap:

Preference = 3.125 + 0.875 * 1 - 0.875 * 0 - 0.375 * 0 + 0.375 * 1 + 0.625 * 1 - 0.625 * 0 = 5.00 (perfect!)

Test machine 2: fast, twocup, expensive:

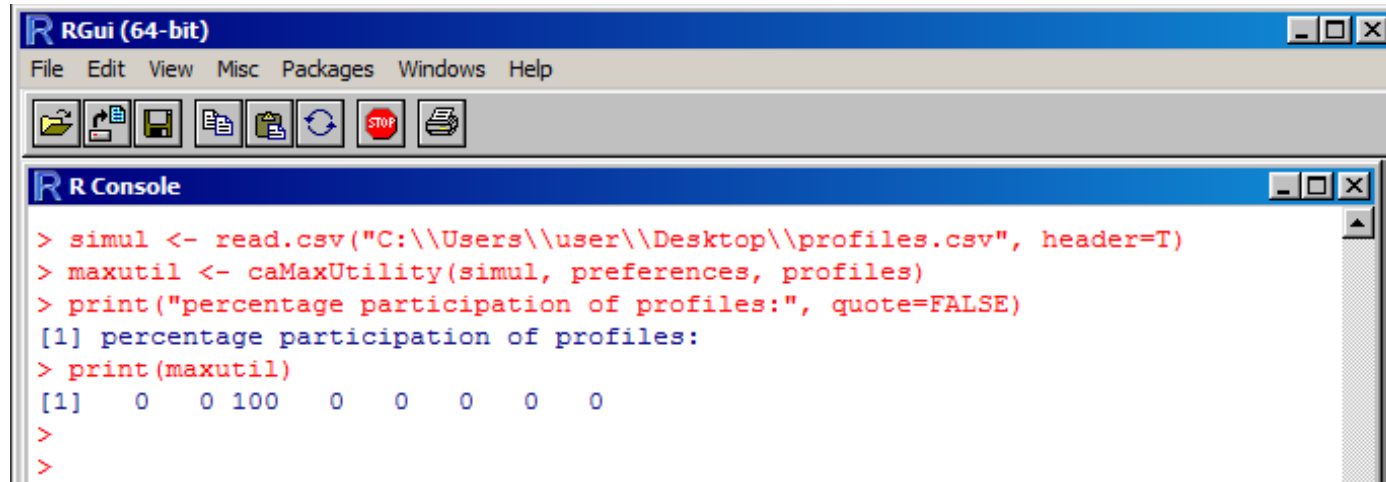
Preference = 3.125 + 0.875 * 1 - 0.875 * 0 - 0.375 * 0 + 0.375 * 1 + 0.625 * 0 - 0.625 * 1 = 4.375 (good)

Test machine 3: slow, twocup, expensive:

Preference = 3.125 + 0.875 * 0 - 0.875 * 1 - 0.375 * 0 + 0.375 * 1 + 0.625 * 0 - 0.625 * 1 = 2 (fair)

Step 9: Market Share Estimation: Maximum Utility

Topic	Discussion
caMaxUtility	maxutil <- caMaxUtility (simulations, preferences, profiles)

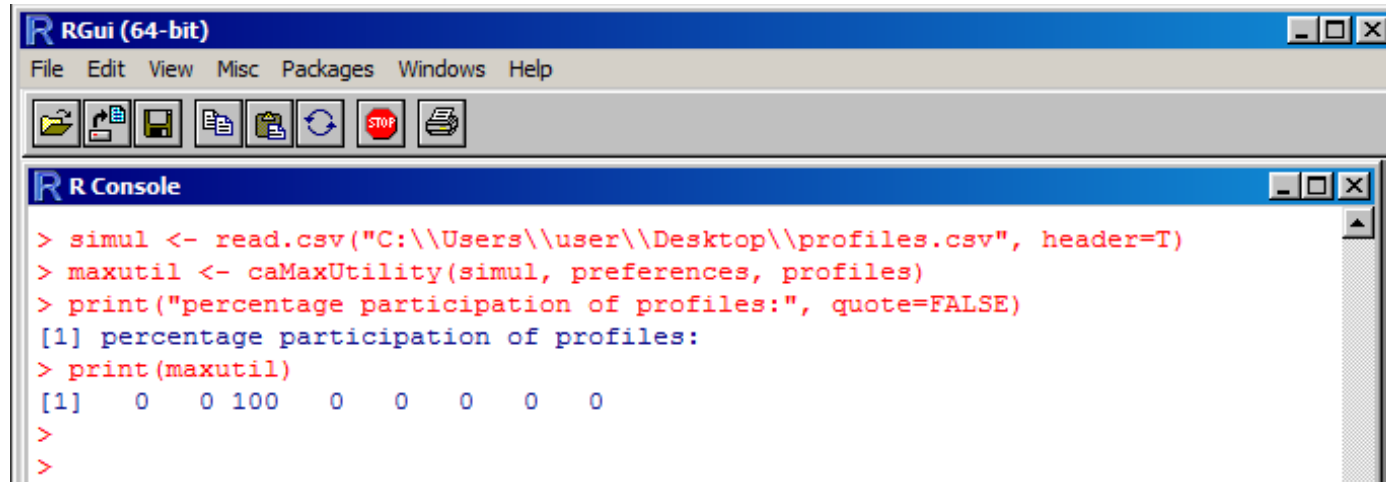


```
> simul <- read.csv("C:\\Users\\user\\Desktop\\profiles.csv", header=T)
> maxutil <- caMaxUtility(simul, preferences, profiles)
> print("percentage participation of profiles:", quote=FALSE)
[1] percentage participation of profiles:
> print(maxutil)
[1] 0 0 100 0 0 0 0 0
>
>
```

Step 9: Market Share Estimation: Maximum Utility

Topic Discussion

caMaxUtility `maxutil <- caMaxUtility(simul, preferences, profiles)`

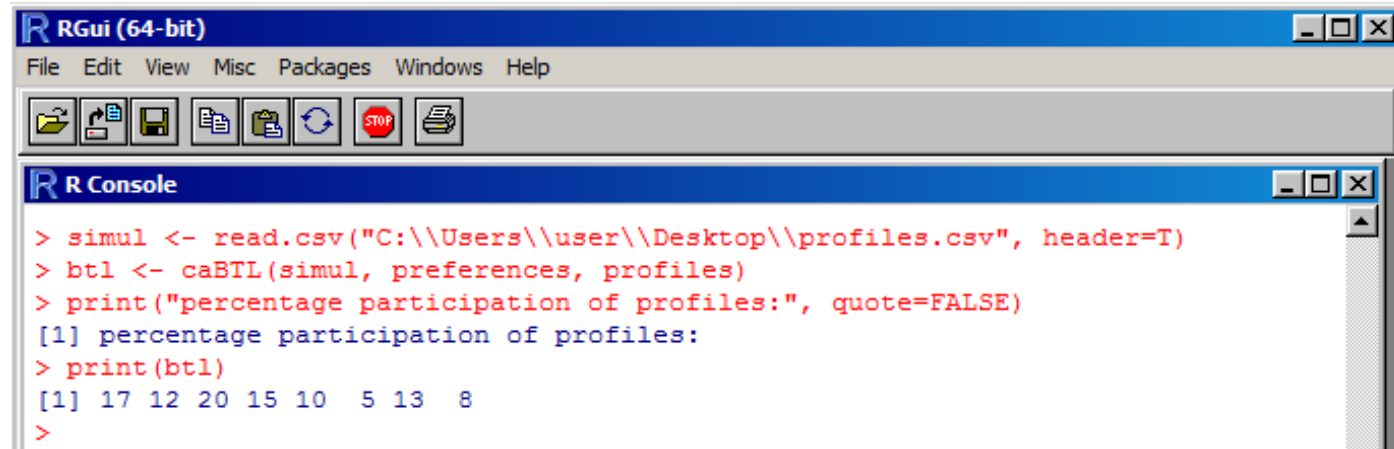


```
> simul <- read.csv("C:\\Users\\user\\Desktop\\profiles.csv", header=T)
> maxutil <- caMaxUtility(simul, preferences, profiles)
> print("percentage participation of profiles:", quote=FALSE)
[1] percentage participation of profiles:
> print(maxutil)
[1] 0 0 100 0 0 0 0 0
>
>
```

	A	B	C	D	E	F	G	H
1	card1	card2	card3	card4	card5	card6	card7	card8
2	4	3	5	4	3	1	3	2

Step 9: Market Share Estimation: Bradley-Terry-Luce (BTL)

Topic	Discussion
caBTL	btl <- caBTL (simul, preferences, profiles)



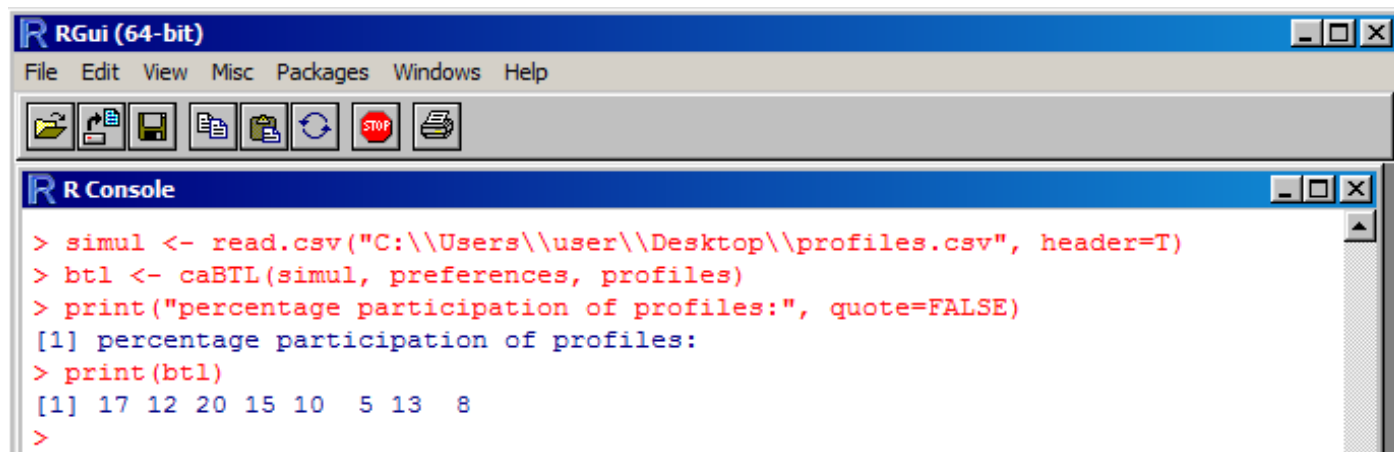
The screenshot shows the RGui (64-bit) window with the R Console pane active. The console displays the following R code and its output:

```
> simul <- read.csv("C:\\Users\\user\\Desktop\\profiles.csv", header=T)
> btl <- caBTL(simul, preferences, profiles)
> print("percentage participation of profiles:", quote=FALSE)
[1] percentage participation of profiles:
> print(btl)
[1] 17 12 20 15 10 5 13 8
>
```

Step 9: Market Share Estimation: Bradley-Terry-Luce (BTL)

Topic Discussion

caBTL `btl <- caBTL (simul, preferences, profiles)`



```
RGui (64-bit)
File Edit View Misc Packages Windows Help

> simul <- read.csv("C:\\Users\\user\\Desktop\\profiles.csv", header=T)
> btl <- caBTL(simul, preferences, profiles)
> print("percentage participation of profiles:", quote=FALSE)
[1] percentage participation of profiles:
> print(btl)
[1] 17 12 20 15 10 5 13 8
>
```

	A	B	C	D	E	F	G	H
1	card1	card2	card3	card4	card5	card6	card7	card8
2	4	3	5	4	3	1	3	2

Step 10: Data Interpretation

Topic	Discussion
Comparison	Compare R results with actual market behavior
R Results	Most important attribute: Speed
Article	Smithsonian article: “The Long History of Espresso Machine” First espresso machines: 1000 cups/ hour
Reviews	Online product reviews, such as those on home-barista.com Many evaluation criteria: - Aesthetic design - Quality of grinder - Degree of automation