Conjoint experiments offline in Qualtrics

Kim Sass Mikkelsen

V-1-1 Wednesday, October 25, 2016

Conjoint experiments in Qualtrics

Conjoint experiments are becoming increasingly popular in political science, not least after the introduction of user friendly software to design (Strezhnev et al. 2014) and analyse (Hainmueller et al. 2014) them. This opens the opportunity for researchers to implement conjoint experiments so long as they are sufficiently familiar with R to use Hainmueller et al.'s *cjoint* package and suffiently savvy to use Strezhnev et al.'s very simple stand-alone *conjointSDT* software.

In a conjoint respondents are presented with multiple *tasks* comparing typically two *profiles* each featuring *values* of a number of *attributes*. Figure 1 shows an implementation from an experiment I ran with Jan Meyer-Sahling and Christian Schuster in Chilean central government. The image shows a table and two subsequent questions (the study in fact has four such questions).

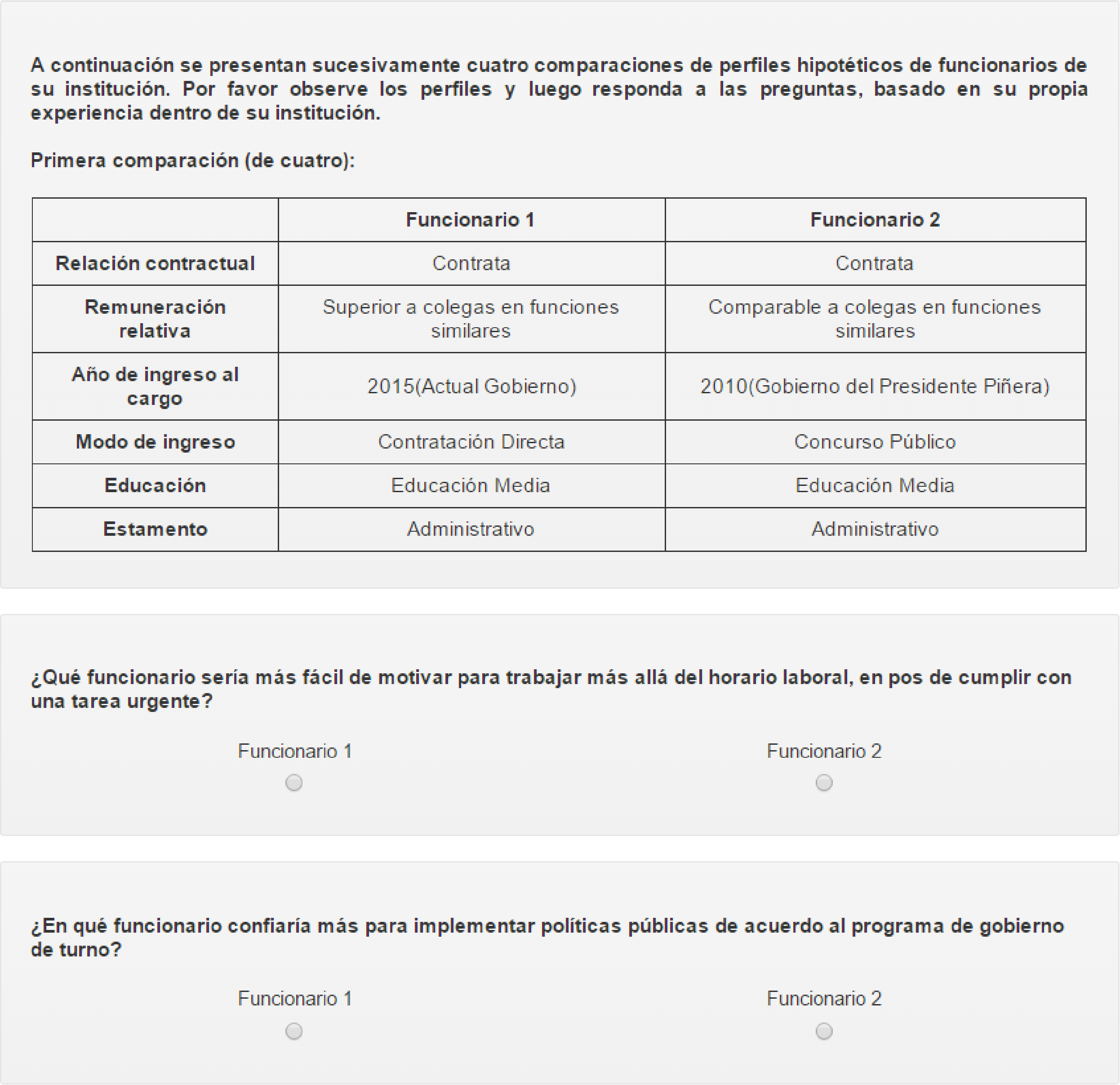
In the table, each row is an attribute, columns marked **Functionario** are profiles, and each cell in each profile contains the values for an attribute for that profile. The two questions below are questions posed to respondents, asking them to choose one profile over the other (in this case, which profile will be easiest to motivate to work hard and which profile is more loyal to a government's policy program).

The problem this note addresses

The conjoint experiment in figure 1 is implemented in Qualtrics and is one of four similar tasks. In each task, new profiles are presented to respondents and the same questions are asked. There is a lot of randomisation involved in this implementation to avoid design effects.

1. Attribute order is randomised for a respondent but consistent across tasks
2. Values are randomised for each profile in each task
3. The order of the two questions are randomised for a respondent but consistent across tasks

Fortunately, Strezhnev et al.'s *conjointSDT* software takes care of 1. and 2. However, this software does not implement 3 (see my other note on conjoints in Qualtrics for a solution to this problem in an online setting).



A continuación se presentan sucesivamente cuatro comparaciones de perfiles hipotéticos de funcionarios de su institución. Por favor observe los perfiles y luego responda a las preguntas, basado en su propia experiencia dentro de su institución.

Primera comparación (de cuatro):

	Funcionario 1	Funcionario 2
Relación contractual	Contrata	Contrata
Remuneración relativa	Superior a colegas en funciones similares	Comparable a colegas en funciones similares
Año de ingreso al cargo	2015(Actual Gobierno)	2010(Gobierno del Presidente Piñera)
Modo de ingreso	Contratación Directa	Concurso Público
Educación	Educación Media	Educación Media
Estamento	Administrativo	Administrativo

¿Qué funcionario sería más fácil de motivar para trabajar más allá del horario laboral, en pos de cumplir con una tarea urgente?

Funcionario 1	Funcionario 2
○	○

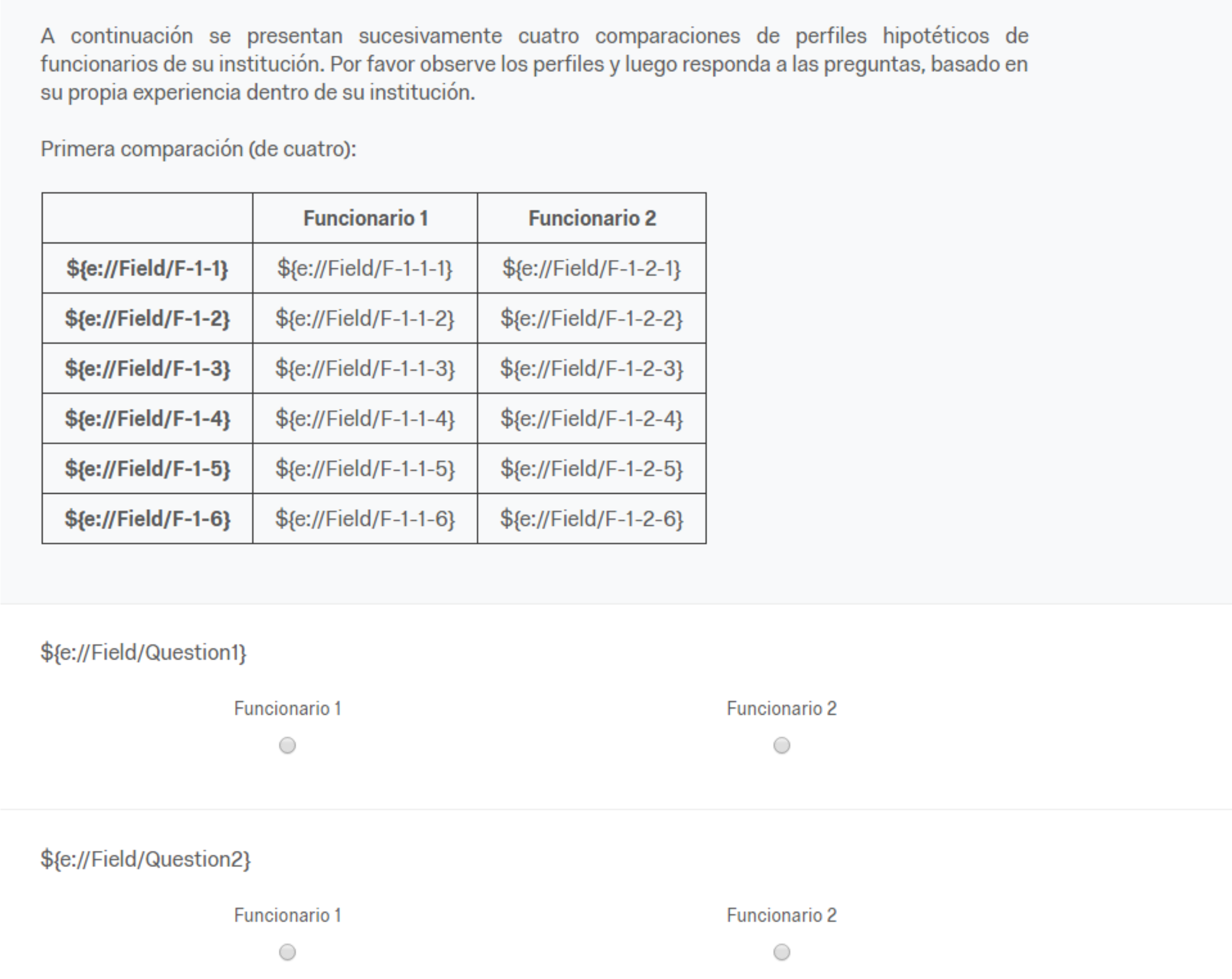
¿En qué funcionario confiaría más para implementar políticas públicas de acuerdo al programa de gobierno de turno?

Funcionario 1	Funcionario 2
○	○

Figure 1: Conjoint from a study of Chile’s central government

Unfortunately, however, none of this works offline. This is unfortunate since, increasingly, surveys in political and social sciences are being implemented in developing countries or other settings where power cuts, poor connections, or other impracticalities prevent reliable internet acces.

In standard implementations of conjoint experiments in Qualtrics, access to the internet is required since the randomisation relies on an online PHP. The content of this PHP is embedded into a table in the survey as shown in figure 2.



A continuación se presentan sucesivamente cuatro comparaciones de perfiles hipotéticos de funcionarios de su institución. Por favor observe los perfiles y luego responda a las preguntas, basado en su propia experiencia dentro de su institución.

Primera comparación (de cuatro):

	Funcionario 1	**Funcionario 2**
${e://Field/F-1-1}	${e://Field/F-1-1-1}	${e://Field/F-1-2-1}
${e://Field/F-1-2}	${e://Field/F-1-1-2}	${e://Field/F-1-2-2}
${e://Field/F-1-3}	${e://Field/F-1-1-3}	${e://Field/F-1-2-3}
${e://Field/F-1-4}	${e://Field/F-1-1-4}	${e://Field/F-1-2-4}
${e://Field/F-1-5}	${e://Field/F-1-1-5}	${e://Field/F-1-2-5}
${e://Field/F-1-6}	${e://Field/F-1-1-6}	${e://Field/F-1-2-6}

${e://Field/Question1}

Funcionario 1 ○ Funcionario 2 ○

${e://Field/Question2}

Funcionario 1 ○ Funcionario 2 ○

Figure 2: The conjoint table and subsequent questions in Qualtrics

The problem I adress in this note is how to fill the table in figure 2 without relying on an online PHP.

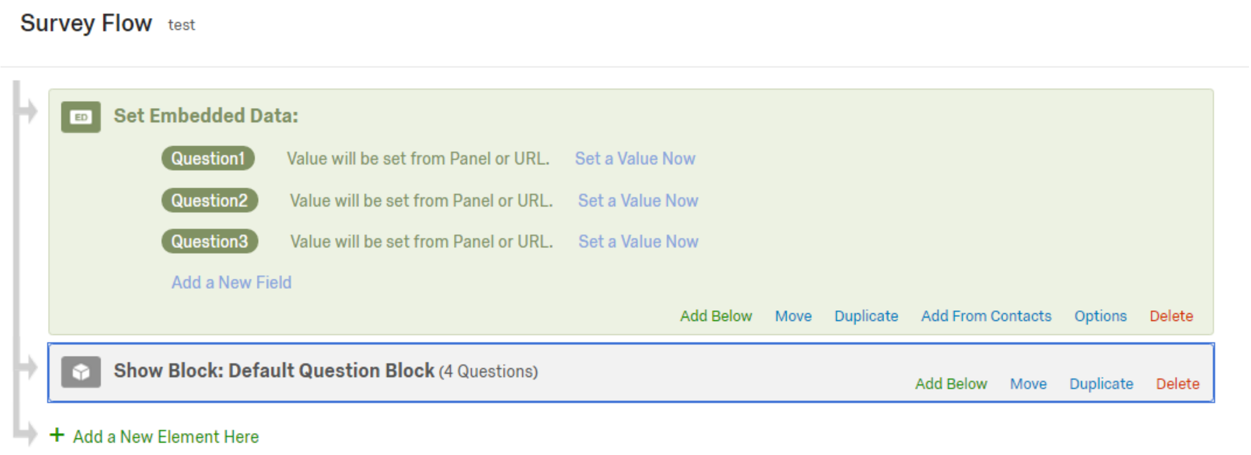
Conjoint questions offline

To randomise conjoint questions consistently across respondents but not across tasks, it is necessary to embed the questions in the survey rather than simply write them up. Online, this can be achieved by writing into the PHP file that is output from *conjointSDT* before it is uploaded. Offline, it is necessary to approach the problem in two steps.

1. Create an embedded data container for the questions.
2. Randomise the questions internally in Qualtrics.

The container needs to be created since Qualtrics does not yet allow the creation of embedded data fields from java script. This problem may be fixed by Qualtrics at some stage, at which point this first step becomes unnecessary. Until then, we need the survey flow.

Figure 3 shows how to include the container for questions in the survey flow. Notice that, unlike the online implementation of conjoint experiments, the embedded data block is used (rather than the web service block for online implementation). Further, notice that all embedded data fields are left empty.



Survey Flow test

Set Embedded Data:

Question1 Value will be set from Panel or URL.

Question2 Value will be set from Panel or URL.

Question3 Value will be set from Panel or URL.

Show Block: Default Question Block (4 Questions)

Figure 3: Embedded data containers for conjoint questions

The second step is to include randomisation of the conjoint questions internally in Qualtrics. To do this, select a question from the survey that everyone will see (e.g. the first question). Shift to html view. And include the following in the field:

```
<html>
<body>

<p> Question to be displayed to the respondent </p>

<script>

  function shuffle(array) {
  var m = array.length, t, i;
  while (m) {
    i = Math.floor(Math.random() * m--);
    t = array[m];
    array[m] = array[i];
    array[i] = t;
  }
  return array;
}

var questionlist = ["question 1","question 2","question 3"];
shuffle(questionlist);
var cjointQuestion1 = questionlist[0];
var cjointQuestion2 = questionlist[1];
var cjointQuestion3 = questionlist[2];

Qualtrics.SurveyEngine.setEmbeddedData("Question1",cjointQuestion1);
Qualtrics.SurveyEngine.setEmbeddedData("Question2",cjointQuestion2);
Qualtrics.SurveyEngine.setEmbeddedData("Question3",cjointQuestion3);

</script>

</body>
</html>
```

This shows the paragraph that contains what the respondent sees and a script that shuffles and embeds questions without the respondent noticing. Inside the script is:

1. A function that implements randomisation of an array (unlike in PHP this is not a base function in java).
2. The list of questions (*questionlist*) that is then shuffled.
3. Creation of variables from the now randomised elements in the question array.
4. Use of a function to embed the randomised content in Qualtrics (Qualtrics recognises this but other java editors do not).

The questions after the conjoint table in figure 1 should now be displayed, randomised by respondent but not by task.

Conjoint attributes and values offline

Randomising conjoint attributes and values is a little more complicated. The first step is similar to the above, including *all* the relevant embedded data fields (e.g. "F-1-1" and "F-1-2-1") in the survey flow.

The second step requires quite a few lines of code. As before, select a question that respondents will see and change to html view. Then input code that is equivalent to the code shown below. To give some substance to the example, the code below implements (roughly) the English language version of the Spanish conjoint in figure 1. To help the reading, I have inlcuded comments in the code. These should not be copied into Qualtrics.

```
<html>
<body>

<p> Question to be displayed to the respondent </p>

<script>

  function shuffle(array) {
  var m = array.length, t, i;
  while (m) {
    i = Math.floor(Math.random() * m--);
    t = array[m];
    array[m] = array[i];
    array[i] = t;
  }
  return array;
}

var salary, Position, education, year, recruitment, status, attributes;

#generate attributes and randomise them

attributes = ["Relative salary","Position","Education","Year",...];

shuffle(attributes);

#generate profiles

var values1 = ["","","","","",""];
var values2 = ["","","","","",""];

# assign values to profile 1
# (repeat from here)

salary = ["Similar to colleagues in similar positions",...];
shuffle(salary);
values1[attributes.indexOf("Relative salary")] = salary[0];

Position = ["Administrative Support","Policy analytical / professional"];
shuffle(Position);
values1[attributes.indexOf("Position")] = Position[0];
```

```
education = ["Secondary education","University Degree<br>(BA, MA, PhD)"];
shuffle(education);
values1[attributes.indexOf("Education")] = education[0];

year = ["2014","2005","1999"];
shuffle(year);
values1[attributes.indexOf("Year of appointment into the public sector")] = year[0];

recruitment = ["Public competition","Internal competition","Appointment"];
shuffle(recruitment);
values1[attributes.indexOf("Recruitment into the administration")] = recruitment[0];

empstatus = ["Civil servant <br> Indefinite term",...];
shuffle(empstatus);
values1[attributes.indexOf("Status and employment term")] = empstatus[0];

# assign values to profile 2

shuffle(salary);
values2[attributes.indexOf("Relative salary")] = salary[0];
shuffle(Position);
values2[attributes.indexOf("Position")] = Position[0];
shuffle(education);
values2[attributes.indexOf("Education")] = education[0];
shuffle(year);
values2[attributes.indexOf("Year of appointment into the public sector")] = year[0];
shuffle(recruitment);
values2[attributes.indexOf("Recruitment into the administration")] = recruitment[0];
shuffle(empstatus);
values2[attributes.indexOf("Status and employment term")] = empstatus[0];

#create variables from the randomised attributes and values.
```

```
var F11 = attributes[0];
var F12 = attributes[1];
var F13 = attributes[2];
var F14 = attributes[3];
var F15 = attributes[4];
var F16 = attributes[5];
var F111 = values1[0];
var F121 = values1[1];
var F131 = values1[2];
var F141 = values1[3];
var F151 = values1[4];
var F161 = values1[5];
var F112 = values2[0];
var F122 = values2[1];
var F132 = values2[2];
var F142 = values2[3];
var F152 = values2[4];
var F162 = values2[5];

#return to Qualtrics.

Qualtrics.SurveyEngine.setEmbeddedData("F-1-1",F11);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-2",F12);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-3",F13);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-4",F14);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-5",F15);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-6",F16);

Qualtrics.SurveyEngine.setEmbeddedData("F-1-1-1",F111);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-2-1",F121);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-3-1",F131);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-4-1",F141);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-5-1",F151);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-6-1",F161);

Qualtrics.SurveyEngine.setEmbeddedData("F-1-1-2",F112);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-2-2",F122);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-3-2",F132);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-4-2",F142);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-5-2",F152);
Qualtrics.SurveyEngine.setEmbeddedData("F-1-6-2",F162);

</script>

</body>
</html>
```

As is clear from this code, creating the embedded data for the conjoint table is somewhat troublesome. The code above fills out the table for the first task. To fill out a second, repeat the code from where it says “repeat from here”, replacing variable names and the strings in the setEmbeddedData function with the appropriate names for the second profile comparison in the second task.

At the end

I hope this note will be helpful. If you found it helpful and use it in published work, please do cite the note as:

Mikkelsen, Kim Sass. 2016. *Conjoint experiments offline in Qualtrics*. Research Note. Odense: University of Southern Denmark