

# Implementing a Conjoint Analysis Design In Qualtrics

Kyle A. Dropp<sup>1</sup>

March 7, 2014

Click [here](#) for latest version.

Comments Welcome!

Conjoint Analysis has been widely used in marketing research and has recently been applied to political science for choice-based research questions (e.g., Hainmueller, Hopkins and Yamamoto (2012)). Examples of its use include studying Americans' preferences over immigrants to be admitted to the U.S. and voters selecting candidates for public office. The fully randomized, factorial design allows researchers to causally identify the relative importance of numerous factors, examine interactions between factors, and examine interactions between factors and respondent characteristics in a cost-effective, principled fashion. This document provides step-by-step instructions for implementing a conjoint analysis design on surveys. The tutorial assumes very basic knowledge of R, HTML, and JavaScript. All supporting materials are available at [kyleadropp.com/conjoint](http://kyleadropp.com/conjoint).

---

<sup>1</sup>Assistant Professor, Department of Government, Dartmouth College [kyle.dropp@dartmouth.edu](mailto:kyle.dropp@dartmouth.edu)

---

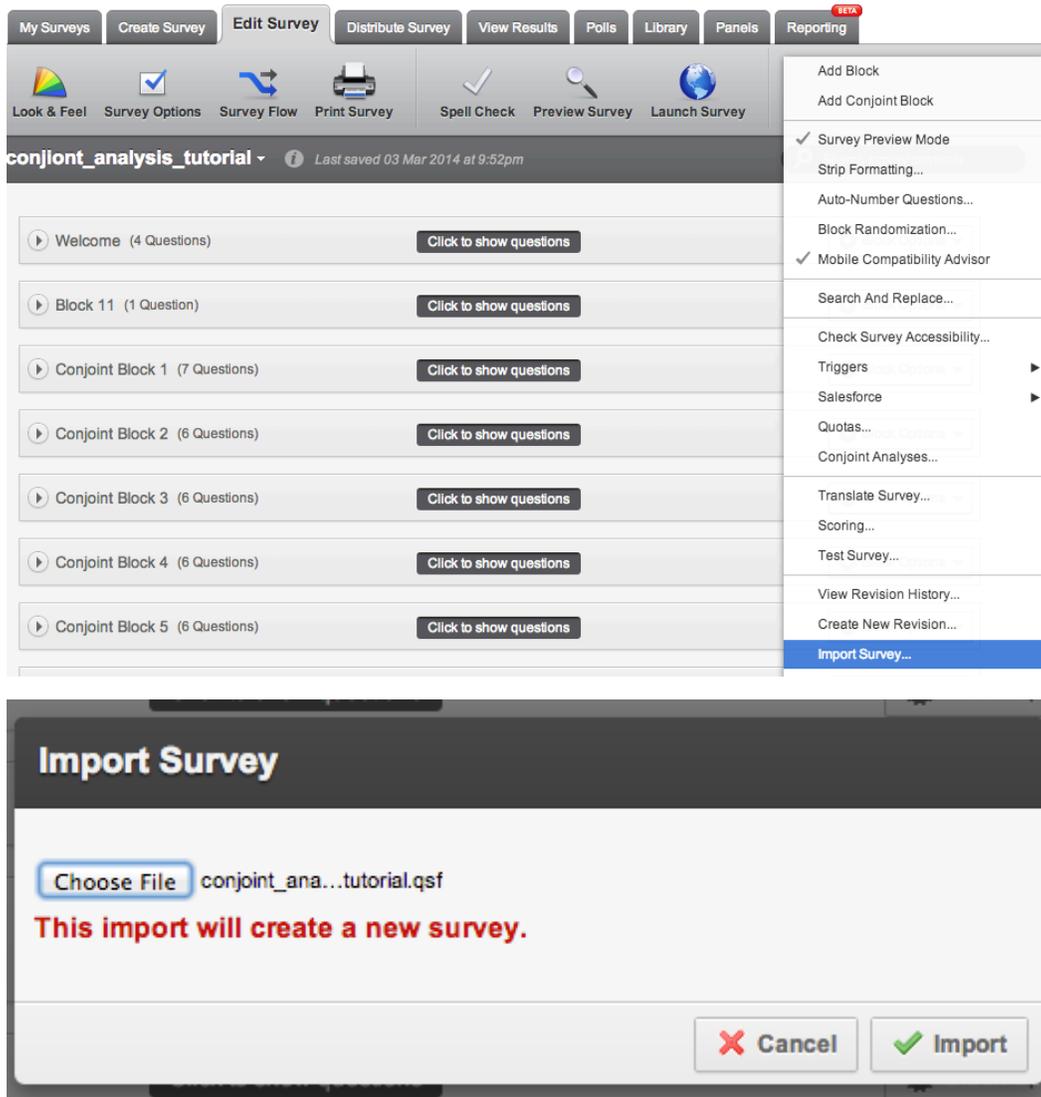
1. Qualtrics Basics	3
1.1. Login	3
1.2. Uploading Survey to Qualtrics	3
1.3. Survey Outline	3
2. Survey Programming for Conjoint Analysis	4
2.1. Dimensions and Traits	4
2.2. Survey Introduction	4
2.3. Navigating to Module 1	4
2.4. Creating HTML Table	5
2.5. Defining Dimensions and Randomizing Dimension Order	5
2.6. Defining Traits	6
2.7. Randomly Selecting Traits for Each Dimension	6
2.8. Assigning Traits to Candidates	6
2.9. Assigning JavaScript values to HTML table	7
2.10. The Main Dependent Variable	7
2.11. Post-Test Dependent Variables	8
2.12. Outputting Randomizations to a .csv	8
2.13. Adding Modules to the Conjoint Analysis	9
3. Downloading and Analyzing the Data file with R	10
4. Modeling and Plots	10

Contents

## 1. QUALTRICS BASICS

1.1. **Login.** Login to your Qualtrics account (e.g., <http://stanforduniversity.qualtrics.com/> for Stanford University), [princeton.qualtrics.com](http://princeton.qualtrics.com/) for Princeton University, or [tuck.qualtrics.com/](http://tuck.qualtrics.com/) for Dartmouth College. See your department or university administrator if you cannot access Qualtrics.

1.2. **Uploading Survey to Qualtrics.** The conjoint analysis survey is pre-programmed for this tutorial, so you can simply upload it to Qualtrics. On the 'Edit Survey' tab, select 'Advanced Options' on the far right, then select 'Import Survey,' and browse to the file 'congoing\_analysis\_tutorial.qsf' online in the supporting materials folder.



1.3. **Survey Outline.** This is a fully functional conjoint analysis design aimed at understanding the relative weight Americans place on various candidate dimensions such as gender, policy positions, newspaper endorsements, and occupations. Scroll through the survey and familiarize yourself with the content.

---

The survey contains 10 blocks: 1) a welcome with informed consent ('Welcome'), 2) introduction to the conjoint analysis exercise ('Block 11'), 3) conjoint module #1 ('Conjoint Block 1'), 4) conjoint module #2 ('Conjoint Block 2'), 5) conjoint module #3 ('Conjoint Block 3'), 6) conjoint module #4 ('Conjoint Block 4'), 7) conjoint module #5 ('Conjoint Block 5'), 8) policy item battery ('Policy'), 9) newspaper readership questions ('DC newspapers'), and 10) background characteristics such as age, race, party affiliation, and educational attainment ('Demographics').

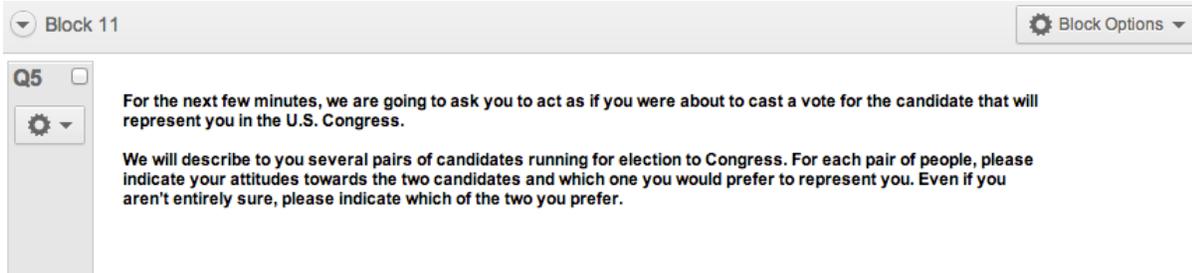
## 2. SURVEY PROGRAMMING FOR CONJOINT ANALYSIS

This section describes the coding for all survey components in the conjoint analysis. These include creating an HTML table, defining the dimensions, randomizing the dimensions, creating arrays containing traits, randomizing traits, populating the HTML table with the traits and dimensions, and saving the data to a .csv file.

**2.1. Dimensions and Traits.** From now on, we will describe dimensions as the key attributes we are varying (e.g., gender, endorsement) and traits as the particular values for a given dimension (e.g., male, female, *Washington Post*)

The dimensions and traits are hard-coded in the HTML / JavaScript below. This facilitates sharing the code and avoids dealing with server permissions. The code contains extensive comments, and all code in this section and foregoing sections can be found at [kyleadropp.com/conjoint](http://kyleadropp.com/conjoint).

**2.2. Survey Introduction.** Head to Q5 in Block 11. This is the description of the exercise.



**2.3. Navigating to Module 1.** Navigate to Q6, the first question in the 'Conjoint Boxes 1' module, click on the content in the question, and then select 'HTML' view. Your screen should look like the figure below. You also can download the HTML code at [kyleadropp.com/conjoint](http://kyleadropp.com/conjoint).



2.4. **Creating HTML Table.** The HTML code contains JavaScript and performs a number of functions. Lines 1 through 48 create an HTML table with columns for the dimensions, Candidate 1 traits, and Candidate 2 traits, respectively. The ‘id’ markers in the table serve as placeholders for the randomized dimension and trait content. The current table has room for six dimensions, but it is trivial to increase or decrease the number of dimensions to suit your research question, as shown in lines 1 through 48 below. See the Appendix for a discussion of adding or deleting dimensions.

```

2   <span> Scenario 1 out of 5 </span>
3
4   <!--Create HTML table-->
5   <font size="3">
6   <table border="4" cellpadding="15" cellspacing="0" width="90%">
7     <col width="140">
8     <col width="30">
9     <col width="30">
10    <tbody><tr>
11      <th><br></th>
12      <th><b>Candidate 1</b></th>
13      <th><b>Candidate 2</b></th>
14    </tr>
15    <!--Create Placeholders for dimensions names (e.g., "att1") -->
16    <!--Create Placeholders for candidate traits (e.g., "a1" for first candidate, "b1" for second candidate) -->
17    <tr>
18      <td id="att1"><br></td>
19      <td id="a1"><br></td>
20      <td id="b1"><br></td>
21    </tr>
22    <tr>
23      <td id="att2"><br></td>
24      <td id="a2"><br></td>
25      <td id="b2"><br></td>
26    </tr>
27    <tr>
28      <td id="att3"><br></td>
29      <td id="a3"><br></td>
30      <td id="b3"><br></td>
31    </tr>
32    <tr>
33      <td id="att4"><br></td>
34      <td id="a4"><br></td>
35      <td id="b4"><br></td>
36    </tr>
37    <tr>
38      <td id="att5"><br></td>
39      <td id="a5"><br></td>
40      <td id="b5"><br></td>
41    </tr>
42    <tr>
43      <td id="att6"><br></td>
44      <td id="a6"><br></td>
45      <td id="b6"><br></td>
46    </tr>
47  </tbody></table>
48 </font>

```

2.5. **Defining Dimensions and Randomizing Dimension Order.** Lines 49 through 63 define the dimensions (Race / Ethnicity, Newspaper Endorsements, Party Identification, Occupation, Position on Assault Weapons Ban, Position on Abortion) and then randomize the order of the dimensions. In the HTML code, we will refer to these dimensions as ‘race,’ ‘endorse,’ ‘pid,’ ‘occupation,’ ‘weapons,’ and ‘abortion,’ respectively.

```

49 <script>
50 // Start of JavaScript
51
52 // Define the Dimensions
53 var attRaw = ["Race/Ethnicity", "Newspaper Endorsements",
54 "Party Identification", "Occupation", "Position on Assault Weapons Ban", "Position on Abortion"];
55 var att = ["Race/Ethnicity", "Newspaper Endorsements",
56 "Party Identification", "Occupation", "Position on Assault Weapons Ban", "Position on Abortion"];
57 var attributes = [ "", "", "", "", "", "" ];
58
59 // Randomize the Order of Dimensions (avoid recency and primacy)
60 for (i=0; i<attRaw.length; i++) {
61 var rand1 = Math.floor(Math.random()*((attRaw.length-i)-0));
62 attributes[i] = att[rand1];
63 att.splice(rand1,1);}

```

2.6. **Defining Traits.** Lines 64 through 76 create the traits associated with each dimension. The dimension traits are typically fully randomized. However, one can constrain the randomization so that both candidates do not have an identical value. For example, in the newspaper endorsement experiment, both candidates cannot receive an endorsement from the same newspaper.

```

// Create Variables for Traits associated with each dimension.
var race_raw = ["White", "Black", "Hispanic", "Asian American"];
var weapons_raw = ["Supports Ban", "Opposes Ban"];
var abortion_raw = ["Pro-life", "Pro-choice"];
var occupation_raw = ["Member of Congress", "Lawyer", "High school teacher", "Business Owner",
"Doctor", "Car Dealer", "Carpenter", "Military Veteran"];
var pid_raw = ["Democrat", "Republican", "Independent"];

// Endorsements constrained not to duplicate
var endorse_rawa = ["Washington Times", "Washington Times", "Washington Post", "Washington Post", "None", "None", "None"];
var endorse_rawb = ["Washington Post", "None", "Washington Times", "None", "Washington Times", "Washington Post", "None"];
var endorse_rand1 = Math.floor(Math.random()*endorse_rawa.length);

```

2.7. **Randomly Selecting Traits for Each Dimension.** Lines 78-91 uses the ‘Math.random()’ command to randomly select traits for each dimension for both candidates.

```

78 //Use math.random to randomly select traits for each dimension for both candidates.
79 var race_a = race_raw[Math.floor(Math.random()*race_raw.length)];
80 var weapons_a = weapons_raw[Math.floor(Math.random()*weapons_raw.length)];
81 var abortion_a = abortion_raw[Math.floor(Math.random()*abortion_raw.length)];
82 var endorse_a = endorse_rawa[endorse_rand1];
83 var pid_a = pid_raw[Math.floor(Math.random()*pid_raw.length)];
84 var occupation_a = occupation_raw[Math.floor(Math.random()*occupation_raw.length)];
85
86 var race_b = race_raw[Math.floor(Math.random()*race_raw.length)];
87 var weapons_b = weapons_raw[Math.floor(Math.random()*weapons_raw.length)];
88 var abortion_b = abortion_raw[Math.floor(Math.random()*abortion_raw.length)];
89 var endorse_b = endorse_rawb[endorse_rand1];
90 var pid_b = pid_raw[Math.floor(Math.random()*pid_raw.length)];
91 var occupation_b = occupation_raw[Math.floor(Math.random()*occupation_raw.length)];

```

2.8. **Assigning Traits to Candidates.** Lines 93-122 create empty arrays for each candidate’s traits and then assigns each trait in the proper order, based on the randomization of the dimensions.

```

97 // Take indexOf the Dimension variable. This is the order of the Dimension variable
98 var race_index = attributes.indexOf("Race/Ethnicity");
99 var weapons_index = attributes.indexOf("Position on Assault Weapons Ban");
100 var abortion_index = attributes.indexOf("Position on Abortion");
101 var endorse_index = attributes.indexOf("Newspaper Endorsements");
102 var pid_index = attributes.indexOf("Party Identification");
103 var occupation_index = attributes.indexOf("Occupation");
104
105 // Use IndexOf Dimension and replace with appropriate trait for Candidate A
106 // For instance, if "Gender" is the 3rd element in the Dimension variable,
107 //this will place Male/Female as the 3rd element of the first candidate's attribute
108
109 att_a_traits[race_index] = race_a;
110 att_a_traits[weapons_index] = weapons_a;
111 att_a_traits[abortion_index] = abortion_a;
112 att_a_traits[endorse_index] = endorse_a;
113 att_a_traits[pid_index] = pid_a;
114 att_a_traits[occupation_index] = occupation_a;
115
116 // Do the same for the second candidate
117 att_b_traits[race_index] = race_b;
118 att_b_traits[weapons_index] = weapons_b;
119 att_b_traits[abortion_index] = abortion_b;
120 att_b_traits[endorse_index] = endorse_b;
121 att_b_traits[pid_index] = pid_b;
122 att_b_traits[occupation_index] = occupation_b;

```

2.9. **Assigning JavaScript values to HTML table.** Lines 123-140, the last chunk in this module, create a list of variables corresponding to the placeholders in the initial HTML table. There are lists of placeholders for the dimensions, for Candidate 1's traits, and for Candidate 2's traits. Then, the code include a for loop to assign the values in the JavaScript variables to the HTML table.

```

124 // Create list of variables in the for loop
125 att_list = ["att1","att2","att3","att4","att5","att6"];
126 a_list = ["a1","a2","a3","a4","a5","a6"];
127 b_list = ["b1","b2","b3","b4","b5","b6"];
128
129 // For elements 1 to 6 in the Dimensions, First Candidate and Second Candidate variables.
130 // The first line assigns Dimensions to each "id" tag in the first column of the HTML table
131 // The second line assigns first candidate traits to each "id" tag in the second
132 //column of the HTML table
133 // The third line assigns second candidate traits to each "id" tag in the third
134 //column of the HTML table
135 for(i=0;i<6;i++){
136     document.getElementById(att_list[i]).innerHTML= attributes[i];
137     document.getElementById(a_list[i]).innerHTML= att_a_traits[i];
138     document.getElementById(b_list[i]).innerHTML= att_b_traits[i];    }
139
140 </script>

```

2.10. **The Main Dependent Variable.** The HTML in the previous section produces the table below. Column 1 displays the dimensions, Column 2 displays traits for Candidate 1 and Column 3 displays traits for Candidate 3. The traits have been randomized for each candidate.<sup>2</sup>

<sup>2</sup>Newspaper Endorsements cannot be duplicated.

Scenario 1 out of 5

	Candidate 1	Candidate 2
Occupation	Member of Congress	Military Veteran
Position on Assault Weapons Ban	Opposes Ban	Supports Ban
Position on Abortion	Pro-choice	Pro-life
Party Identification	Independent	Independent
Race/Ethnicity	Black	Asian American
Newspaper Endorsements	Washington Times	Washington Post

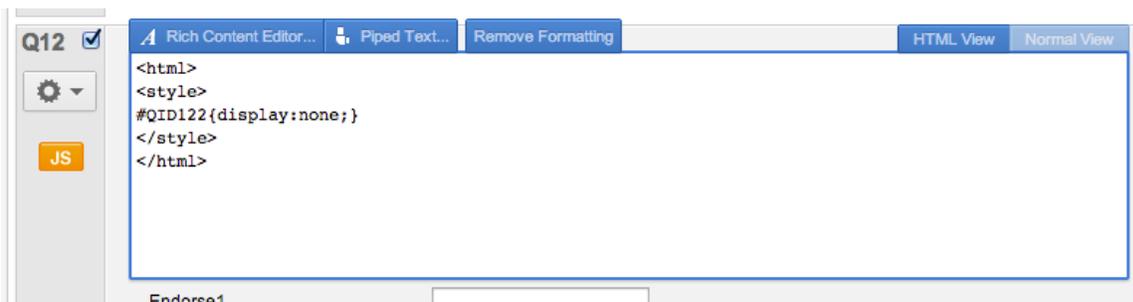
Which of these candidates would you be more likely to support for Congress?

Candidate 1  Candidate 2

**2.11. Post-Test Dependent Variables.** Follow up questions on the same survey block ask respondents to rate how often they would support each candidate, to evaluate the qualifications of each candidate, and to evaluate the ideology of each candidate. These questions are designed to examine whether a newspaper endorsements influences candidate support because it affects perceived qualifications or because it affects perceived ideology of the candidate.

**2.12. Outputting Randomizations to a .csv.** Scroll down to Q12. This is a “Text Entry-Form” Question that will output the list of traits and the dimension order to a .csv file. Note that this question has a bunch of open-ended response forms. Each of the dimensions and traits will be assigned to one of these forms, but respondents will not see this question.

Click the content of Q12 and then select ‘HTML View.’ You will see a few lines of CSS designed to hide this question from the respondent.



The ‘QID122’ portion does not always correspond with the question identifier next to it. You can select ‘Piped Text,’ then ‘Q12,’ then ‘All Choices - Entered Text’ to identify the correct QID that should be used to hide this question from respondents.

Click on the orange ‘JS’ button on the left hand column of Q12. This is a list of variables from the HTML code in previous steps. It saves each variable to an appropriate form number, and then these variables are exported to a .csv file. Note: if you add or subtract dimensions, you must add cells to this form.

---

```

Qualtrics.SurveyEngine.addOnload(function()
{
  /*Place Your Javascript Below This Line*/
  this.setChoiceValue(1,race_a);
  this.setChoiceValue(2,weapons_a);
  this.setChoiceValue(3,abortion_a);
  this.setChoiceValue(4,occupation_a);
  this.setChoiceValue(5,endorse_a);
  this.setChoiceValue(6,pid_a);
  this.setChoiceValue(7,race_b);
  this.setChoiceValue(8,weapons_b);
  this.setChoiceValue(9,abortion_b);
  this.setChoiceValue(10,occupation_b);
  this.setChoiceValue(11,endorse_b);
  this.setChoiceValue(12,pid_b);
  this.setChoiceValue(13,attributes);

});

```

You have now completed programming module 1. It is simple to add extra modules to the survey.

**2.13. Adding Modules to the Conjoint Analysis.** A principal benefit of using conjoint analysis is that respondents can answer multiple modules. This increases statistical power and enables a within- and between-subjects analysis. We make a few key assumptions such as preference stability, profile randomization, and no ordering effects, each of which can be examined empirically in diagnostic tests. This section details the process for adding extra modules to the survey.

Navigate to Q13 in the block called ‘Conjoint Block 2.’ Click on the content and select ‘HTML View,’ or download the HTML code for module 2 at [kyleadropp.com/conjoint](http://kyleadropp.com/conjoint).

There are two very minor changes you must make to the HTML file from module 1. First, in Lines 1 through 10, change ‘Scenario 1’ to ‘Scenario 2’ and changes the names of the candidates from 1 and 2 to 3 and 4, respectively.

```

1  <b> Scenario 2 out of 5 </b>
2  <font size="3">
3  <table border="4" cellpadding="15" cellspacing="0" width="90%">
4  <col width="140">
5  <col width="30">
6  <col width="30">
7  <tbody><tr>
8  <th><br></th>
9  <th><b>Candidate 5</b></th>
10 <th><b>Candidate 6</b></th>

```

Second, in lines 47-49, rather than defining the dimensions, you will use the same randomized dimension order from Module 1. The embedded data variable in line 48 takes the list of dimensions from open text item Q12 (identifier Q122) and line 48 splits this comma delimited list.

Next, for questions Q14, Q15, Q16, and Q17, change the candidate names from ‘Candidate 1’ and ‘Candidate 2’ to ‘Candidate 3’ and ‘Candidate 4,’ respectively.

Continue to add modules until you have enough for your research question. You typically can click ‘Block Options’ and then ‘Copy Block’ to create a module retaining the essential features of earlier modules.

---

### 3. DOWNLOADING AND ANALYZING THE DATA FILE WITH R

Note: this section requires some familiarity with R. This section provides details for accessing and downloading the .csv file containing results from the survey. I have uploaded a .csv file containing data from a newspaper endorsements experiment conjoint analysis, along with the codebook, to the supporting materials folder online ( [kyleadropp.com/conjoint](http://kyleadropp.com/conjoint) ).

Please open the file ‘read\_clean\_data.R’. In this conjoint analysis design, we assigned newspaper endorsements to individuals based on their ZIP Code. If there was no major newspaper in their zip code, they were excluded from the study. The script file reads and processes the data, runs a few models, and then creates plots. It is heavily commented and you should be able to run it in one click from the command line.

The code performs the following tasks (in order): clear workspace, load R libraries, import .csv file from survey data, omit rows with NAs, recode variables for weighting and append post-stratification weights to the data frame.

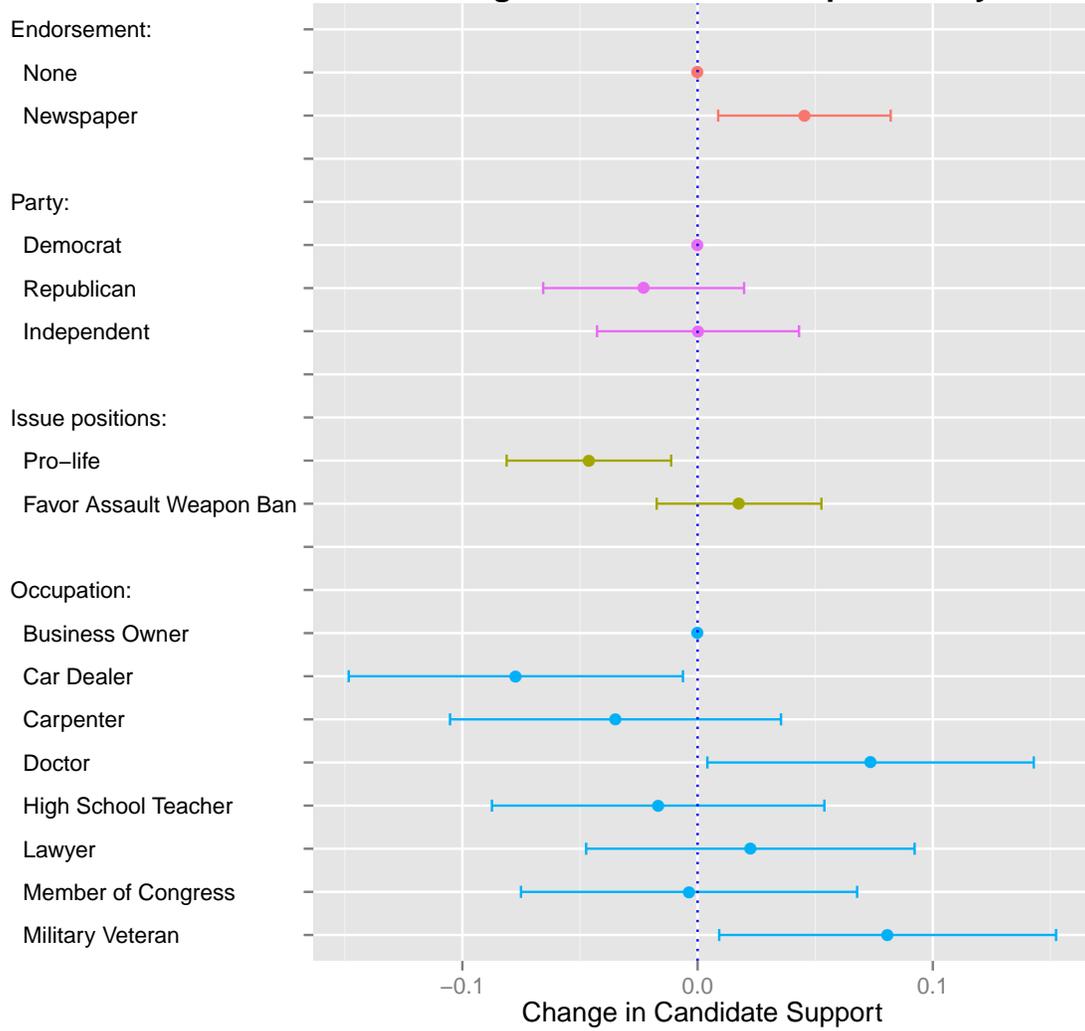
Respondents in this study viewed three modules, for a total of six candidates. We will create six rows for each respondent, with each row containing the full list of a particular candidate’s attributes, whether the respondent selected that candidate, and the perceived support, qualifications, and ideology for that candidate. Then, we include columns with newspaper perceptions, policy questions, and demographic items. In our study, we were interested in understanding how the ideological congruence between a respondent and a newspaper influences the overall effect of a newspaper. Therefore, the next section of this R file constructs a number of distance measures using self-reported respondent ideology, self-reported ideology of newspapers, self-reported policy positions, and newspaper slant scores. Finally, we include variables for the respondent id and the vignette id to include in our models.

### 4. MODELING AND PLOTS

We have now processed the raw data from the conjoint analysis experiment. We will now regress candidate selection (dichotomous), candidate support (continuous), candidate qualifications, and candidate ideology on the five dimensions varied in the conjoint analysis. The models are administered separately with survey weights (and without survey weights), control for respondent age, educational attainment, gender, ethnicity, and region, and include fixed effects by designated market area and the module number.

Here is a figure displaying main effects

### Effects of Candidate Attributes on Candidate Support Among Those Who Read Paper Weekly+



Here is a table displaying the main and interaction effects (below).

TABLE 1. The Relationship between Individuals' Ideological Congruence and the Impact of a Newspaper Endorsement - Results from Full Dataset

	All Voters		Read Local Paper >= Weekly		Read any Paper in Past 24 Hrs.	
	DV = 1 if Selected Candidate, 0 if Did Not Select Candidate					
	(1)	(2)	(3)	(4)	(5)	(6)
Endorsement	0.022 *	0.071 *	0.034 *	0.078 *	0.036 *	0.077 *
	(0.009)	(0.013)	(0.011)	(0.017)	(0.011)	(0.017)
Absolute Ideological Distance		0.055 *		0.053		0.048
		(0.019)		(0.028)		(0.026)
Endorsement X Absolute Ideological Distance		-0.157 *		-0.143 *		-0.132 *
		(0.032)		(0.044)		(0.042)
<i>N</i>	15648	15528	9024	8952	8916	8850
<i>R</i> <sup>2</sup>	0.010	0.012	0.011	0.013	0.013	0.014
Resid. sd	0.502	0.502	0.505	0.504	0.504	0.504

Models control for age, gender, education, and race. They include fixed effects for the binary comparison order and at the Metropolitan Statistical Area (MSA) level.

Standard errors in parentheses

\*\* indicates significance at  $p < 0.05$ , \* indicates significance at  $p < 0.10$