Lab 2: The Stroop Effect: Does Congruence Matter?

Lewis O. Harvey, Jr. and Samuel P. Paskewitz

PSYC 4165: Psychology of Perception, Fall 2018

Department of Psychology and Neuroscience
University of Colorado Boulder
Boulder, Colorado 80309

25 and 27 September 2018

John Ridley Stroop (1897-1973)

Overview of the Experiment

So far this semester, our lab activities have involved fixing "broken" PsychoPy experiment scripts by adding missing components. You've added stimuli components, such as Text and Images); you've also added the response components Keyboard and Mouse components. These stimuli and response components are the most concrete elements of what the subjects (i.e., you) experience during the experiment. You even have experience with the more abstract elements of experimentation: By inserting Loops into the flow of an experiment and by loading conditions files (.xlsx or .csv), you have specified the experimental manipulations to the stimuli. On the analytic end of the experiments, you have become proficient in running segments of R scripts, performing basic computations, and you even modified short segments of R scripts.

The purpose of Lab 2 is to pull these tasks together into a simple experiment where you build the entire PsychoPy script, collect data, then perform individual and group analyses by assembling the basic data analysis elements R (import, reorganize, analyze, plot). Some parts of the experiment are, by necessity, highly specified (e.g., some components need exact names). Other parts of the experiment are don't need to be so exact, so you have some freedom to customize your experiment.

Lab 2 will recreate a classic psychology experiment on the Stroop Effect. We will complete Lab 2 over the course of 2 weeks in two stages:

1. Create the Stroop experiment, collect your own data, and analyze them;
2. Build a script to analyze the group data, then analyze them

By the end of this lab, you should be able to:

• Build a PsychoPy experiment from scratch • Assemble R script components to analyze individual and group data • Test a hypothesis using response times as a dependent variable

We'll begin by performing a brief literature search to define the Stroop Effect, such as what stimuli are traditionally used, and how it's measured. We will also compare/contrast a couple of the major online databases (Google Scholar & Web of Knowledge), each of which have strengths and weaknesses, depending on the purpose of your search. Very often, those new to research have inefficient search strategies, such starting with individual research articles, which provide minimal research context. In Figure 1, we recommend a search strategy that organizes different type of sources by scope (general to specific):

Search Encyclopediae

In a web browser find the Wikipedia entry on the *Stroop Effect*. copy and paste the Wikipedia description into the Introduction.Rmd file in the Lab2 report folder, for later use.

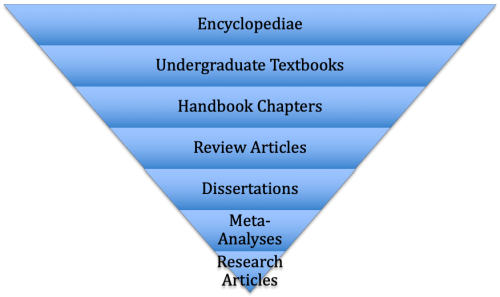

Figure 1: A hierarchy of literature search strategy. Start with the general (e.g., Wikipedia) and move to more and more specialized sources like textbooks and individual research articles.

Search for Review Articles

In the web browser go to Google Scholar (https://scholar.google.com) and search for *Stroop Effect*. The top hit should be Colin MacLeod Psychological Bulletin review paper entitled *Half a century of research on the Stroop effect: an integrative review* (MacLeod, 1991). A pdf file of the MacLeod article is included in the *Lab_2_Documentation* folder. Read the first few paragraphs and stop when you get to the heading *Stroop's Classic Article* (midway down the left column of page 164). Add notes to the Introduction section answering three questions:

1. What was the phenomenon that was so interesting to MacLeod and the papers he reviews?
2. What was the empirical controversy?
3. What was Stroop's contribution?

Search for Research Articles

Now go to the Web of Knowledge web site (https://webofknowledge.com) and search for Stroop's 1935 article, entitled *Studies of interference in serial verbal reactions*. A pdf of the Stroop paper (Stroop, 1935) is included in the *Lab_2_Documentation* folder. You may have to go to the CU Library A_Z Databases: W page and then click on Web of Science: https://libguides.colorado.edu/az.php?a=w.

Stroop did not have computers in 1935 to carry out his experiments. Describe his methods in Experiment 1. Put your description in the Introduction.Rmd section of your lab report for later use. What was the result of his Experiment 1, seen in Stroop's Table I and described on page 17 of his paper (Stroop, 1935)?

Build the PsychoPy Experiment

Time to build the experiment in PsychoPy. It goes fast if you work in teams of two, so pair up with your neighbor. We'll start by defining the experimental conditions in the conditions file, appropriately name *conditions.xlsx*.

The Condition File

Open *Lab 2 Tools > Stroop exp > conditions > conditions.xlsx* by double-clicking on the file. It should open in Excel. Note the header names on row 1. Each column becomes a variable in PsychoPy with the header name. You should see four columns named `textWord`, `textColor`, `corrAns`, and `congruency`. **Do not change these names!** We need these exact names later in the R scripts that analyze your data.

- The `textWord` column defines the words that are the stimuli for the experiment. We'll use the words *red*, *green*, and *blue*.
- The `textColor` column defines the color of the words when they appear on the computer screen. The colors we will use are *red*, *green*, and *blue*.
- Subjects will identify the color of the words by pressing the *left*, *down*, and *right* arrow keys on the keyboard. The word colors and the correct responses are given in Table 1:

Table 1: The correct answer key for each of the text colors used in the experiment. These values should be entered in the corrAns column of the conditions file.

letterColor	corrAns
red	left
green	down
blue	right

- Add the correct response in the corrAns column of the conditions file for each row of using the `textColor` column as a guide. Be sure to use only `left`, `down` or `right` with no extra characters or blank spaces after the word. Use only lower case letters.
- In the `congruency` column, add whether the condition is `congruent` or `incongruent`.
- If the `textWord` and the `textColor` are the same, enter `congruent`
- If the `textWord` and the `textColor` are different, enter `incongruent`
- Check your spelling carefully. Make sure that the only words in the `congruency` column are either `congruent` or `incongruent`.

Make the Actual Experiment

- Launch PsychoPy and create a new experiment using the *File > New* Menu item.
  - Save the new experiment to the *Lab 2 Tools > Stroop exp folder*.
  - Name the experiment *Stroop.psyexp* when you save it.

Trials Routine

- Add two components to the existing *trial* routine
  - A *Text* component named *word*
  - A *Keyboard* component named *resp*
- In the *word* text component
  - Make the word appear 0.5 seconds after the start of the *trial* routine
  - Leave the duration field blank so the word is visible during the whole trial
  - Make the text twice as big as the default size

Add a Loop around the Trials Routine

- Add a Loop around the *trial* routine and name it *trials* (that will probably be the default name)
  - In the Conditions field of the *trials* loop use the `Browse` button and navigate to the conditions folder and choose the *conditions.xlsx* file you edited earlier.
  - If all goes well, below the Conditions field you should see **12 conditions, with 4 parameters [textWord, textColor, corrAns, congruency]**
  - Set *nReps*, the number of repletion blocks, to 10
  - Save your work

Set Word and Color in the Trials Routine

- We need to tell PsychoPy which word and which color to show on each trial. Add the following variables to the *word* text component of the *trials* routine. The $ indicates to PsychoPy that these are Python variables, in this case defined in the conditions file.
  - `$textWord` to the Text field of the *word* text component
  - `$textColor` to the Color field of the *word* text component
  - Change the dropdown boxes to the right of these fields from `constant` to `set every repeat`. If your script does not work properly you probably forgot to change these settings.

Keyboard component

- The keyboard component, *resp*, of the trial routine is used to collect the subject's response to each text stimulus. We obviously don't want the keyboard to become active until the word actually appears on the screen.
  - set the start of the keyboard to be the same time as the word
  - set the keyboard to only accept `left`, `down` and `right` buttons.
  - set the keyboard to record whether or not the answer was correct
  - tell the keyboard where the correct answer is to be found (`corrAns` variable of the condition file)
- Now test the program by clicking on the Green Runner button. Run 10 or 15 trials and stop by pressing the `escape` key.
  - Open the most recent .csv data file located in the `data` folder. The file names are time stamped so the most recent normally would be at the bottom of the data folder.
  - Check that the *resp.corr* column of the data file contains 0 and 1s and that they truly reflect whether or not the response in the `resp.keys` column are correct or incorrect.
  - Correct any mistakes you may have made in the code until your test of the program shows it is working correctly.

Add a Practice Routine to the Experiment

- Once you are very certain that the *trial* routine is working properly, add a practice routine in the flow diagram before the *trial* routine. The practice routine will give subjects to learn which response key corresponds to which color (red = left, green = down and blue = right).
  - Name the new routine *practice* and put a loop around it called *practice_trials*. (PsychoPy does not permit spaces in names, so be sure to include the underscore in the practice_trials name).
  - The *practice* routine is **identical** to the *trial* routine, except that the text and keyboard components must have different names, for example *word2* and *resp2*.
  - Set the number of blocks in the *practice_trials* loop to 2 (*nReps* in the loop setting)
  - Save and test

Add Instruction Routines

- So far, we've created the core of our experiment, now we need to add routines that include on-screen instructions for subjects when taking the experiment. Each of these Routines need only 2 components: a text and a keyboard component. The text will show a message to the subject, and the keyboard component will advance the experiment by ending the routine. The names of these routines & components do not really matter, but give the routines names that are meaningful to you. The default component names work just fine.

- Add a routine before the practice routine that welcomes the subject and gives them instructions on how to do the experiment. Tell them to press the 'space' key to start the practice trials.
- Add a routne between the practice and the trial routines to give more instructions and to warn the subject that now the real experiment begins. Tell them to press the 'space' key to start the test trials.
- Add a routine after the trial routine thanking the subject and telling them that the experiment is finished. Set the duration of the message to 3 seconds, so you don't need a keyboard component.

Run the Experiment

You are now ready to run the experiment for real. Verify that the csv data file is really recording the data. Open your last csv data file in Excel and check it one more time. Ask one of us to verify your program if you are not sure.

After you have run the experiment (24 practice trials and 120 test trials) check your data file.

If the file is OK upload it to the Lab 2 csv Assignment in Canvas.

Individual Data Analysis

Time to make meaning from your personal responses. Before you start, you need to make an analysis plan. The most common mistake people can make when analyzing data is also the worst: starting the analysis without a plan. Let's avoid this frustrating and time-consuming error by making an analysis plan! We need to explicitly state (a) the variables in our data set (independent, dependent); (b) how these data tell us anything about our phenomenon of interest, in this case color-word interference effects.

Variables

- Every experiment has at least two types of variables:
  - Independent Variable (IV): In experimental settings, the stimulus condition whose values are varied by he experimenter independently of any other variable in the situation (American Psychological Association, 2010). In non-experimental studies the independent variables are often called predictor variables.
  - Dependent Variable (DV): In experimental settings, any variable whose values are the results of changes in one or more independent variables (American Psychological Association, 2010). Dependent variables are also called measurement variables or outcome variables.
  - Check out definitions at https://dictionary.apa.org/browse/d.
- Not all independent variables are created equal, and it sometimes can get pretty confusing figuring out which variable is important to our research questions. Here are a couple terms to help us think about independent variables in an experiment:
  - **Stimuli-level variable**: The way(s) that stimuli are systematically varied that allow measurement by varied responses.
  - **Hypothesis-level variable**: The variable(s) in an experiment that allow the testing of a hypothesis.

In perception research, hypothesis-level variables are the contrasting conditions used to test a hypothesis. The levels in a hypothesis-level variable tend to be few in number (2 or 3 levels), but the differences between those levels tend to be large. On the other hand, stimuli-level variables tend to have many levels, and those differences also tend to be small. While variability of stimuli is important to any hypothesis, the results of a stimuli-level variable tend to be uninformative to the bigger picture of the experiment.

What are the independent variables in our Stroop experiment? Which are so-called stimulus variable(s) and which are hypothesis variable(s). What were the dependent variable(s) in this experiment?

Expected Results

You have seen the tilde math symbol (~) used in R formulae to represent "depends upon." For example:

```
alertness ~ coffee consumption
```

According to this formula, mental alertness is *dependent upon* an amount of coffee consumed: a testable hypothesis. In the Stroop experiment we want to test the effects of congruency on reaction time to name a color. You will express this hypothesis as a formula:

```
reaction time ~ congruency
```

Taking the time (a) to explicitly identify the function of variables, (b) to define a mathematical relationship between those variables, and (c) to explicitly test that relationship not only saves you time in the long run, but also ensures you understand what it is that you are doing at every step of your analysis. For most people this type of reasoning takes time to develop, but is crucial to the research process.

Stroop's Data

Stroop's main finding, Figure 1 in his paper (Stroop, 1935), is reproduced in Figure 2. You will get a chance to compare these results with ours when we analyze the data from the whole class.

Your Own Data

After you have uploaded your .csv data file to Canvas, move a copy of it to the *my_lab_2.0_additional_files* folder in your *Lab_2_Report_Folder that you downloaded from the course web site earlier today. Launch RStudio by double-clicking on the project file (*Lab_2_Report_Folder_2018_Fall_Broken.Rproj*). RStudio will launch and you can see all the files in the folder in the Files tab in the lower right pane of the RStudio window.

Double-click the *my_lab_2.5a_Individual_Results.Rmd* file in the file pane to open it for editing in the edit pane (upper left) of RStudio. Fix the broken code chunks in the markdown file that need fixing and carry out your analysis on your individual data.

Once you get the file to work properly (choose 'Run All' from the Run tab in RStudio) you can add your own text with your own descriptions and your own conclusions about your individual data. Once `Run All` works you can knit the file to a pdf to make sure that it will produce a pdf file. Edit the title page to put your own name where indicated. You can knit the whole report by knitting the *my_lab_2.0_Report.Rmd* file. It will not have any group results, however, until next week when we will add them.

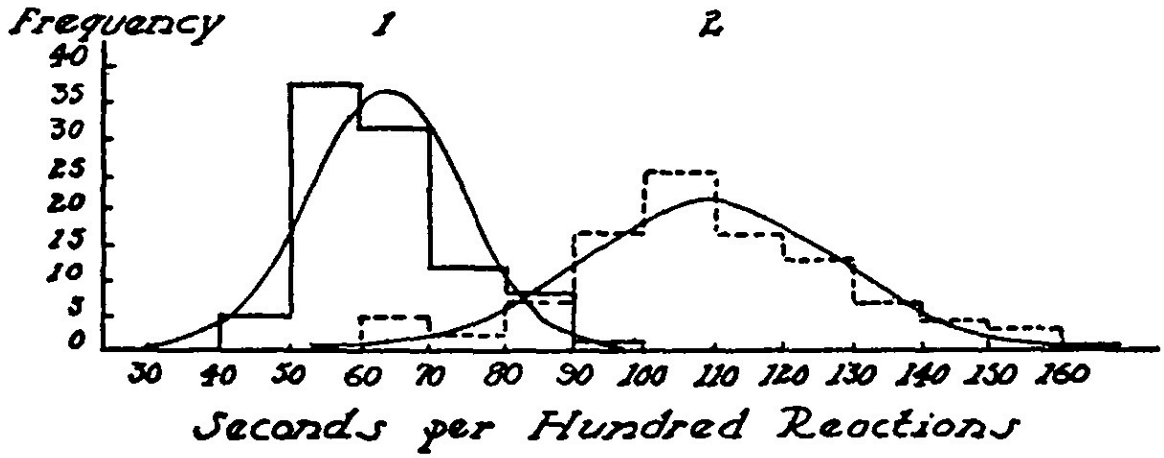

FIG. 1. Showing the effect of interference on naming colors. No interference (1); interference (2).

Figure 2: Figure 1 from Stroop (1935).

References

American Psychological Association. (2010). *Publication manual of the american psychological association* (6th ed., pp. xviii, 272p.). Book, Washington, DC: American Psychological Association.

MacLeod, C. M. (1991). Half a century of research on the stroop effect: An integrative review. *Psychological Bulletin*, *109*(2), 163–203. Journal Article.

Stroop, J. R. (1935). Studies of interference in serial verbal reactions. *Journal of Experimental Psychology*, *18*(6), 643–662. Journal Article.