



CSCI 4620U / SOFE 4850U Lab 9



Purpose

To get experience a simulation of A/B testing techniques.

Background

A/B testing compares two versions of an element (A and B) simultaneously. An A/B test is measured by a defined metric for success (i.e. there is not a lot of analysis required). Usually, the test is automatically deployed online by giving visitors to a website either option A or B and continuously measuring the metric (e.g. signups). Depending on the performance of each option, the algorithms can automatically promote one option and stop showing the other.

Another type of test deployed online is the 5 second test – this test seeks immediate feedback on a design. It might be subjective feedback (e.g. “Do you think this website looks professional?”) or quantitative measures of success on timed tasks (e.g. “Click the search box.”). The timing is applied to make sure that a visitor can complete the task quickly, under the assumption (backed by data) that if people can’t figure out a web interface quickly, they will leave. First impressions matter.

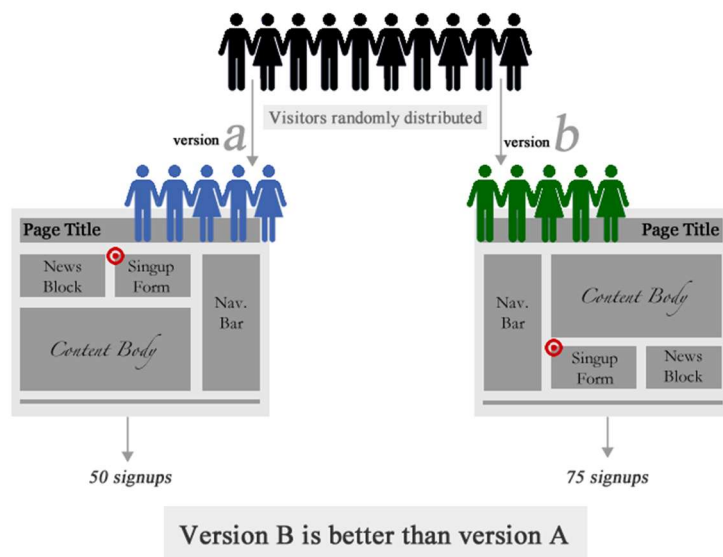


Figure 1: A/B Testing Method Overview

Even though every A/B test is unique, certain elements are usually tested:

- wording, size, color and placement of buttons
- headline or descriptions
- form length and types of fields
- layout and style of website
- images on landing and product pages
- amount of text on the page (short vs. long)
- icon design and placement
- navigation flow

Usually an AB Test should vary just one of these things at a time, so that it is clear what the effect of any differences is. But sometimes radically different designs are tested, then refined.

Overview

This lab will simulate a hybrid of these two methods. You will design and conduct a 5 second test using the A/B method.

We have created two programs: **designer** and **tester** for you to use. These programs are written and can be opened in *Processing* (processing.org). Source code is provided for your reference, but you should *use the executable provided as appropriate for your O/S*.

Designer:

Allows you to load image two files and set active regions on each, then save the properties file.

Tester:

Loads the designer properties file and runs the AB test repeatedly, putting the output in a log file for analysis. A & B versions are randomly presented for 5 second tests. Task success (click the correct region) and time are recorded. Any tasks which take more than 5 seconds are ended and considered a failure. As soon as a test is completed, the next test is queued.

Each run of tester outputs a time stamp to the log file, then a list of results consisting of the test type (A or B), the success or failure of the test, and the time taken to click in milliseconds.

The log file and properties file are found in C:\users\\abtester

Note on design: Your design problem asks you to mock up a web interface for a particular problem, to draw specific attention to an important feature. Note that we expect you to use the principles learned in class for useable and aesthetic design: appropriate whitespace, fonts, colors, button sizes, etc. You may use images and screenshots from the web, or icons from

resources such as nounproject.com. Your design should not unrealistically draw attention to the element of focus. For example, if your problem was to design so people can find how to change a hotel booking, it would be inappropriate to have the home page of the booking website have a huge red “change my booking” button on an otherwise bare screen! **Remember the overall purpose and function of the website (in this case, finding and booking hotels), and don’t undermine that to make your particular test successful.**

Tasks – Design (35 minutes)

1. You will be assigned a problem on your handout – check your testing handout for your assigned problem. Each problem presents a design problem where a visitor is expected to notice and correctly click a particular element of the screen within 5 seconds. Note that these problems are *simulations* of a typical test. Normally a test like this would be deployed on a live site and “participants” would not be given instructions, nor would they even know they were in an experiment! So, this study lacks *external validity*.
2. Using image editing tools, PowerPoint, or sketching + your camera, create an interface to satisfy your problem. You may gather resources (screen elements) from existing websites using a screen capture tools, or you may design from scratch. Save this as your “A” test image. *Note that your “A” option can simply be a screenshot from the assigned website.* If the website requires scrolling, in every case you can just screenshot and work with the visible area at the top of the page
3. Now, modify your interface to give an alternative version of the design – this will be your “B” test image. Be careful to ensure that the fidelity of the design of B is as good as A. That is, if A is a sketch, B can be a sketch, but if A is a screenshot, then carefully work with your screenshot to modify convincingly. See the provided example PowerPoint A/B mockup for booking.com. You can modify “A” as much as you want – remove, move and change page elements. Note: We are not trying to make a clearly ‘better version’ but rather two alternative versions where is actually not obvious which might win! Don’t be too perfectionist!
4. Export your A and B images as jpg or png files. If using PowerPoint, you can just screenshot them. Ideal image size is 1024x768 but any image size will be rescaled appropriate.
5. Load your images into the **designer** tool and specify the correct regions for the users to click to satisfy the test.
6. Save the properties file.

Tasks – Test Setup (15 minutes)

1. Every student has been given a handout listing the number 1-30. At the advice of your TA, please cross out any unused numbers (e.g. if there are only 24 people present, cross out 25-30).
2. Copy your station number to the back of your problem handout. Place your problem handout beside your computer, with the **participant instructions side up**.
3. Open the `tester` program.
4. Run a few rounds of your test and check the log file to make sure things are working.
5. When confirmed, set your system in testing more and wait for instructions.

Note: If you close and reopen the tester the log file will not be erased (new tests are appended).

Tasks – Running the Tests (20 minutes)

1. Now every student should visit all other stations to conduct the 5 second test. Use your handout to cross out the numbers of stations you have visited to ensure you reach every machine. **Please be quick and organized with minimal conversation.**
2. At each station, read the instructions, click start, then click what you think is the appropriate location on the screen, as quickly as possible. Once you have clicked or when 5 seconds passes, the experiment ends. Move on to the next station.
NOTE: you DO NOT need to visit the stations in order.

Tasks – Analysis (20 minutes)

1. After everyone has completed all experiments, return to your computer and end the testing.
2. Open and review the log files – they show the success, failures, and times for each test.
3. Write a brief report which contains:
 - a. Images of your A and B tests, clearly labelled
 - b. A brief paragraph discussing the test – which version do you think was more successful? Remember to consider both time and accuracy. How would you next refine the winning version to try another test?
 - c. Paste the output of the log file at the end.

Submission

Submit your report to the assignment dropbox on Blackboard by the end of the lab.