CSCI 130 – A/B Testing and User Testing

Release Date: October 18, 2018
Due Date: November 1, 2018, 12:00 pm

***Note:** This assignment has **an intermediate deadline on Tuesday, 10/23 at 1:00pm**. Get started early, stay on top of your tasks, and come to TA Hours!*

**If you need help at any point during this assignment, remember that you can post on Piazza (publicly or privately) and we can get back to you fairly quickly!*

Overview

In this assignment, you will learn about quantitative and qualitative user tests for evaluating interface designs. You will collect and analyze user behavior through quantitative data in A/B Testing and qualitative data in User Testing.

Suggested Timeline

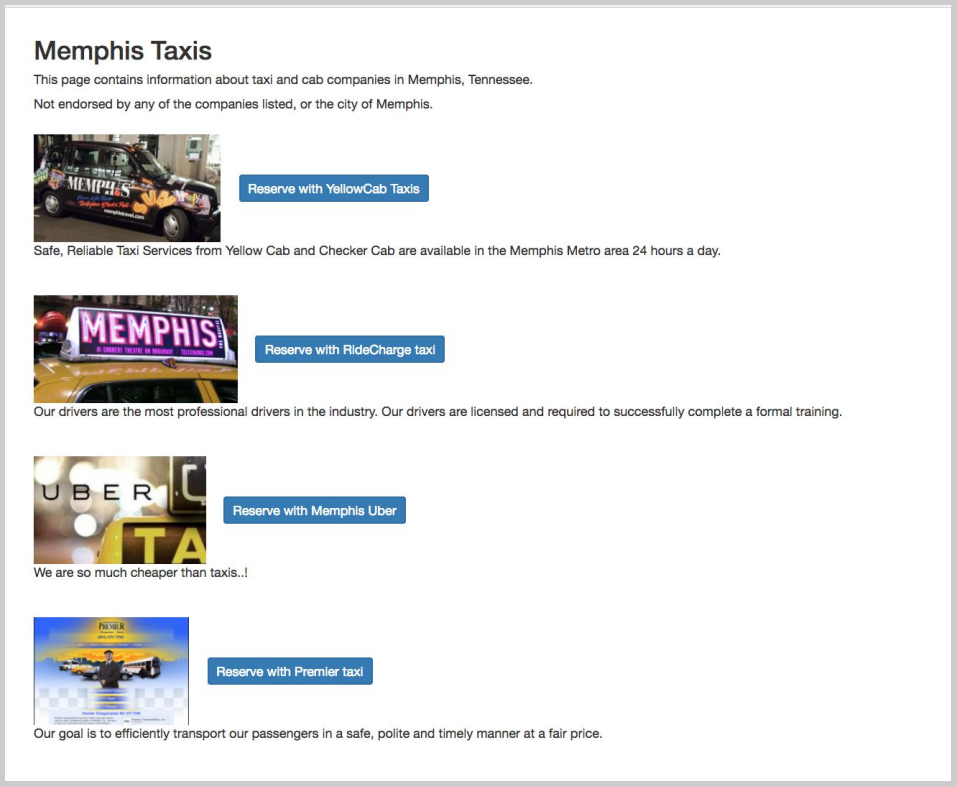
Task	Finish By
Terminal/Git and Calculating Metrics Gear Up	**Saturday, 10/20 2:00-3:00pm, CIT 269**
Design your A and B versions of the website	Sunday, 10/21
Formulate your A/B Testing hypotheses	Monday, 10/22
Make sure your data collection is working properly before class on Tuesday, 10/23!	Monday, 10/22
Complete your A/B Testing website (hosted on Heroku) before class. Everyone must attend this in-class activity, otherwise you won't have data that you need to complete the assignment!	**Tuesday, 10/23, 1:00pm**
Formulate your User Testing hypothesis	Thursday, 10/25
Complete your 3 user tests on UserTesting.com. This is just a suggested date, but you should try to send out your test on 10/27 to make sure you get your result before the deadline.	**Saturday, 10/27**
Analyze your data	Sunday, 10/28
Finalize your write-up	Wednesday, 10/31
Hand in!	Thursday, 11/1

Part I: A/B Testing

A/B testing is a UX method commonly used to evaluate an interface's performance. You generate hypotheses about visitor metrics, and visitors to your interface are randomly presented version A or B, and you collect data about their session. Then, you analyze the data to draw conclusions about your hypotheses and use them to inform design changes.

1. Formulate your Hypotheses

1. **Think about how you will change this template to create two new and improved versions** (A and B) of a website. Consider principles like Affordances and Usability. The template looks like this:



2. Then, **write down your null and alternative hypotheses** for each of the following 4 metrics (8 total). **Explain the reasoning behind your alternative hypotheses.** Check out the Analysis slides from lecture if you're not sure how to do this.
    a. **Click rate:** proportion of sessions that have at least one click
    b. **Time to click:** average time it took a session to do the first click
    c. **Dwell time:** average time a session spent on an external page before returning to the landing page
    d. **Return rate:** proportion of sessions that left the landing page and returned

Here's an example of a set of hypotheses for click through rate:

- ***Null Hypothesis:*** *The click through rate get on my Version A will be equal to the click through rate on my Version B.*
- ***Alternative Hypothesis:*** *The click through rate I get on my Version B is going to be greater than that of Version A, because I changed my Version B to be more attractive with flashy images.*

3. Design the Website

You will host your website using Heroku, which will allow you to display your A or B version randomly. It's important that you set it up correctly in order to collect the right data for your test, so **follow the steps below carefully!**

1. Go to https://signup.heroku.com/ and create a free account. Choose Python as the primary development language while signing up. Don't forget your username and password because you'll need it later!

2. If you don't already have Git installed, go here to download and install it.
    a. If you have a Mac, you may run into a pop-up that says git can't be opened because it's from an unidentified developer. If this happens, go to *System Preferences* > *Security & Privacy* > *General*, and under "*Allow apps downloaded from*" there should be a note saying that "git was blocked from opening," and you can hit the "Open Anyway" button.

3. On your computer's file browser, **create a new folder** where you want to keep all of the files for this project (this is where the stencil code will be downloaded).

4. **Download the project files**: Open Terminal (on Mac) or PowerShell (on Windows) and navigate to the folder that you just created. If you're not familiar with navigating to folders in Terminal/PowerShell, there's a guide here.

    Once in the correct folder, run the following command:

    ```
    git clone https://github.com/matsuim/a-b-testing.git
    ```

    You should see a new folder called a-b-testing in your project folder now! This folder should contain some text files, Python files, and three folders.

5. **Modify Version A and Version B** of the website by editing the A.html and B.html files found in the "templates" folder of the a-b-testing folder you downloaded, **based on your hypotheses.**
    a. If you are new to web development, check out this guide for a tutorial. Also note that you'll need to open the html files in a text editor like Atom or Sublime in order to make changes.
    b. Your changes can be small and simple, since we are more focused on the A/B testing in this assignment, and not the interface design itself. We are not expecting you to make any major design changes. Things like changing the size, color, and/or typeface of an element would work.

c. TAs are happy to help you with your html during hours!
d. Don't copy and paste the HTML in A.html to B.html and vice versa. This will mess up your data collection.
e. *Note that simply opening the .html files won't fully load the page. To see your changes, you will need to push to Heroku (see next step).*

6. To push your website to Heroku, **follow the instructions in the Heroku guide** for your operating system: Mac/Linux Guide | Windows Guide

★ *If you need help with this section, come to the **A/B Testing Gear-up** on Saturday, 10/20, from 2-3pm at CIT 269!*

4. Collect the Data

Your website needs to be up and running on Heroku for this part October 23, 1:00pm, so please make sure that data collection is also working properly, too. You will visit other students' websites in class on this day. If your website isn't running, you won't have any data to work with for the rest of the assignment!

You can test your data collection by following the steps below and clicking around on both the A and B versions of your Heroku site. When you look at the data, make sure entries are being recorded for both the A and B versions.

BEFORE you start collecting data, navigate into your a-b-testing folder and run the command:

Mac Terminal:
```
heroku logs --tail > mylog.txt
```

Windows PowerShell:
```
heroku logs --tail | Out-File mylog.txt
```

After you hit enter, you'll see a blank line at the bottom of your terminal, which means that the command is running. This means that your computer is listening for data (page loads and clicks) from your website to save. Make sure that the command is running **the entire time** you are collecting data, otherwise you will lose data!

When you're done collecting data (having people use your website), press **Ctrl - C** in your terminal to stop listening for data. To get just the lines of data that we want for this project, run

Mac Terminal:
```
grep AB_TEST mylog.txt > myfilteredlog.txt
```

Windows PowerShell:
```
Get-Content mylog.txt | Select-String -Pattern "AB_TEST" | Out-File myfilteredlog.txt
```

***If you run the heroku logs command again, it will overwrite your previous data unless you change "`mylog.txt`" to a different file name, i.e. "`mylog2.txt`"**

5. Read the Data

Once you've gotten enough visitor data, you will read the data logs for the next portion of the assignment in order to calculate your metrics.

1. You should now have a file called **myfilteredlog.txt** in your a-b-testing folder. **Open myfilteredlog.txt in a text editor**. Each line represents an event, and each part of each line corresponds to a metric:

```
<timestamp> AB_TESTING <ver. a/b> <page load time> <click time> <id of clicked element> <sessionID>
```

2. For example, your log data should look like this:

```
2017-10-05T17:26:19.598801+00:00 app[web.1]: AB_TESTING: B 1507224379455 0 0 fndowmclai
2017-10-05T17:26:20.579381+00:00 app[web.1]: AB_TESTING: B 1507224380415 0 0 fndowmclai
2017-10-05T17:26:24.840825+00:00 app[web.1]: AB_TESTING: B 1507224380415 1507224384783 mp1 fndowmclai
2017-10-05T17:26:28.808994+00:00 app[web.1]: AB_TESTING: A 1507224388741 0 0 itzjseohpj
2017-10-05T17:26:32.252820+00:00 app[web.1]: AB_TESTING: A 1507224388741 1507224392189 mp2 itzjseohpj
...
```

   - ➢ You can just subtract one timestamp from another (`<page load time>` - `<click time>`) to get the difference in milliseconds.
   - ➢ If the `<click time>` in an event line is 0, it means that the event was just a page load. If the user does click a link, there will be another log line with the same `<page load time>` and `<sessionID>` but a non-zero `<click time>`.

3. To get clean data that you can load into Excel, run

   Mac Terminal:
   ```
   tr ' ' ',' < myfilteredlog.txt > myfilteredlog.csv
   ```

   Windows PowerShell:
   ```
   cat myfilteredlog.txt | % { $_ -replace " ", "," } > myfilteredlog.csv
   ```

   You can now open myfilteredlog.csv with Excel.

6. Analyze the Data

★ *If you need help with this section, come to the **A/B Testing Gear-up** on Saturday, 10/20, from 2-3pm at CIT 269!*

The slides from the Analysis lecture will be helpful here. You can compute the following by hand, but we highly recommend using Excel (or writing a script) to find your metrics to avoid tedium!

1. **Compute the following 4 metrics for versions A and B** (8 metrics in total). **Explain how you calculated each metric** (1 sentence or formula each).
   a. **Click rate:** proportion of sessions that have at least one click

b. **Time to click:** average time it took a session to do the first click
c. **Dwell time:** average time a session spent on an external page before returning to the landing page
d. **Return rate:** proportion of sessions that left the landing page and returned

2. For each of those 4 metrics, conduct a statistical test (see lecture slides) **to determine whether or not the difference between versions A and B is statistically significant.** For each metric,

    a. **Choose the correct type of statistical test**, and **explain why** you chose it.
    b. **Compute the statistic** and **include** it in your write-up.
        i. If you are performing a Chi Square test, you should **show your work** to derive your statistic by including: photos of your hand-written calculations, an Excel worksheet, a script, etc., but make sure to include it in your hand-in. You may **not** use online libraries that do Chi Square *entirely* for you, but you can use them to check your answer!
        ii. If you are performing a t-test, you **can** use online libraries to compute the statistic for you.
    c. **Interpret where your p-value** lies in the appropriate statistic table, and **explain its meaning** in terms of your hypotheses.

3. Compute a 95% confidence interval for **the difference between the average time to click for version A and the average time to click for version B**. You can use online libraries to compute this. **Interpret** and explain the meaning of your confidence interval in the context of your null and alternative hypotheses about time to click.

Part II: User Testing

★ *Make sure you read through this part before constructing your test!*

Having test participants try out an interface is an important part of testing and a valuable source of feedback. There are many ways to do this, starting with good old-fashioned usability testing. In this assignment, **you will conduct your own usability test through a remote user testing service** (UserTesting.com), using your newly created, interactive hi-fi prototype. Again, we stress that it is important that you **send your test out early**! Most of these results should come back within a day, but it's a good idea to give yourself as much time as possible to analyze feedback and complete the write-up.

1. Turn Your Website into a High-Fidelity Mobile App Prototype

Use ***one*** version of your Memphis Taxis site as the basis to create a prototype of a mobile app with multiple screens. The prototype should have **interactive click-through**, which can be done using programs such as InVision, Proto.io, and Figma (no programming required for any of these). The prototype should reflect all the distinguishing characteristics of your original site (such as the things you changed in part 3 "Design the Website"), and add additional screens that you feel necessary for users to perform certain tasks. Your prototype doesn't have to be too detailed, but it should be detailed enough for your usability testers to be able to perform **at least one task**. **You must end with a publicly accessible link to your app prototype in order to continue!**

2. Formulate Your Hypothesis

Select a task you want users to complete on your interactive prototype, and come up with a short **qualitative** hypothesis about how users will perform on this task.
You only need one task (which may consist of several sub-tasks). The tasks on User Testing can be sub-parts of performing the function or performing the same function with different conditions/priorities in mind. Focus on the **primary function(s)** of the app, and come up with one overarching hypothesis on how you believe users will fare in performing this function.
You only need 1 hypothesis, though you may choose to have more. Consider possible areas of confusion and the relative amount of time users will spend on different sub-tasks.
After you have your hypothesis, create your UserTesting.com experiment with list of questions and prompts for the user. **Record both your hypothesis and your testing instructions in your final write-up**. TAs should be able to reconstruct your experiment from your instructions alone.

3. Conduct Your Remote Usability Tests

Read through the User Testing Guide and follow step-by-step to set up and ship your very first remote usability test. You have to gather feedback from 3 users. UserTesting.com and similar services allow UX researchers to test their interface on remote participants and ask them about

their experience using a prototype. These questions can be as general as '*Did you find what you were looking for?*', and as specific as '*Can you easily read the light gray disclaimer text at the bottom?*' Try to isolate specific aspects of your interface that you want feedback on and design your usability test around them. When the tests are completed, you'll receive an email containing a video with the user's feedback, as well as answers to your specific questions and prompts. Remember, you'll want to be as specific as possible with what you want users to do, such as how to perform certain actions or accomplish certain tasks. Check out the Usability lecture slides [16-20] for advice.

Keep in mind, you'll want users performing a specific task or action - **be as explicit as possible**! There are sample interface/usability questions on the UserTesting site. You can also ask users for general feedback - this should help with overall analysis! **Put a screenshot of the email you receive from UserTesting.com with the results of your tests in your handin.**

4. Analyze Your Feedback

If you do not receive feedback before the submission deadline, please email the TA staff. Once you've received feedback, explain what the UserTesting results mean by **directly addressing hypotheses in terms of the results** and **analyzing it by calculating the metrics we discussed in Usability lecture**. Specifically, you should include **one table** calculating the metrics for each sub-task, including tasking completion rate, error counts, and time to complete the task (like the example below). You should also write **a short paragraph** including more qualitative analysis of the tasks, including what do the metrics in the table reflect, what are the types of errors users made (slips, lapse, or mistake), and how satisfied users were when completing the tasks.

	Completion Rate	Error Count	Time on Task
Sub-Task 1			
Sub-Task 2			
...			

Next, write about **potential interface changes** you would make based on your UserTesting results and feedback. For example, perhaps your participants were confused by the back button - how would you change it to make it more intuitive? Finally, **write about your testing experience.** Were there any challenges? What did you learn? What was successful?

Part III: Comparison

1. Imagine yourself as a UX Researcher at Memphis Taxis Co. presenting to a committee of stakeholders **and recommend what your company should do next in 3-5 sentences**. Consider what questions they may want to ask a UX researcher, and get creative! Based on your tests, what

do you recommend your company stakeholders to do next (e.g. use version A, use version B, conduct more tests, redesign version A, etc.)? **Use your data analyses from A/B Testing and remote user testing to support your argument.**

2. Still thinking like a UX Researcher, **how does your A/B Testing data compare with your UserTesting data**? Were there any behavioral trends you found in one that you didn't find in the other? **What are the advantages and disadvantages of using A/B testing versus remote user testing? (2-3 sentences)**

Handing in Your Assignment

Upload a zip file to Google Drive and submit it to Gradescope by **12:00pm on November 1, 2018**. **The zip file should include the following in the checklist. All writing and images should be contained in a PDF inside the zip file.** Additional files, such as Excel Spreadsheets, can be included in the zip file.

Part I: A/B Testing

- ❑ **8 hypotheses -** null and alternative hypotheses for the 4 metrics (click through rate, average time to click, dwell time, and return rate for each of versions A and B)
- ❑ **Your Heroku site URL**
- ❑ **The event log file, myfilteredlog.txt.** You don't need to include your entire a-b-testing folder, just the log file.
- ❑ **8 computed metrics -** 4 metrics for each of versions A and B.
- ❑ **4 sentences (or formulas) explaining how you calculated each metric**
- ❑ **4 sentences explaining why you chose which statistical test** - 1 for each of the 4 metrics
- ❑ **4 computed test statistics** with work shown - 1 for each of the 4 metrics
- ❑ **4 p-value interpretations -** 1 for each of the 4 metrics
- ❑ **Confidence interval calculation** with work shown

Part II: User Testing

- ❑ **Link to your high-fidelity prototype**
- ❑ **Emails from your UserTesting sessions, with the confirmation email for video and test data**
- ❑ **A short paragraph including your hypothesis and your testing instructions.** Your testing instructions should include the specific tasks given to participants.
- ❑ **An analysis addressing your hypothesis directly**
- ❑ **A table and a short paragraph calculating and explaining the metrics,** including completion rate, error count, time to complete, error type, and user satisfaction.
- ❑ **A short paragraph (3-5 sentences) evaluating the potential interface changes**
- ❑ **A short paragraph (3-5 sentences) writing about your testing experience**

Part III: Comparison

- ❑ **A short paragraph (3-5 sentences) presenting findings about version A and B and your mobile app** recommending what actions Memphis Taxis Co. should take next, using your data analysis
- ❑ **A short paragraph (2-3 sentences) evaluating the tradeoffs** between A/B Testing and User Testing analysis

The total amount of text of your document should be no more than about 2,000 words (or two pages), but feel free to insert any visuals within the text or in additional pages.

Grading and requirements (22 points)

A/B Testing (11 points)

- 1 pts — Valid HTML changes for version A and version B, working Heroku hosting, and a mylog.txt file containing your Heroku logs
- 1 pt — 8 clear, justified hypotheses of the proposed changes
- 2 pts — Computed and explained the 4 metrics correctly for the two versions
- 3 pts — Chose and explained the correct statistical test for all 4 metrics
- 3 pts — Computed the test statistic, interpreted the p-value, and showed work for each metric
- 1 pt — Calculated and interpreted the confidence interval correctly

User Testing (6 points)

- 1 pt — Interactive prototype that allows at least one task to be completed and confirmation emails from UserTesting
- 1 pt — Written statement of hypotheses, as well as proposed tests informed by the hypotheses. Describe UserTesting.com experiment with list of questions and prompts (TAs should be able to conduct the same experiment by reading your write up)
- 1 pt — Explain UserTesting results by directly addressing hypotheses in terms of experiment results
- 1 pt — Table and short paragraph calculating and evaluating the metrics listed in the handout (completion rate, error count, error type, time to complete, and user satisfaction), with brief summary explaining the metrics
- 1 pt — Consider possible interface changes inspired by results and user feedback
- 1 pt — Analyze and reflect upon your usability testing experience. Address unexpected challenges, successful methods, things you learned, areas for improvement, etc.

Comparison (3 points)

- 2 pts — Short paragraph of a final recommendation on what Memphis Taxis Co. should do next, using concrete evidence from analysis in parts I and II
- 1 pt — Short paragraph comparing data from A/B testing and user testing, discussing tradeoffs of each testing method

Style (2 points)

- Check out the style guide for more details!