‹ Return to Classroom

DISCUSS ON STUDENT HUB

Analyze A/B Test Results

REVIEW

HISTORY

Requires Changes

1 specification requires changes

Great work really

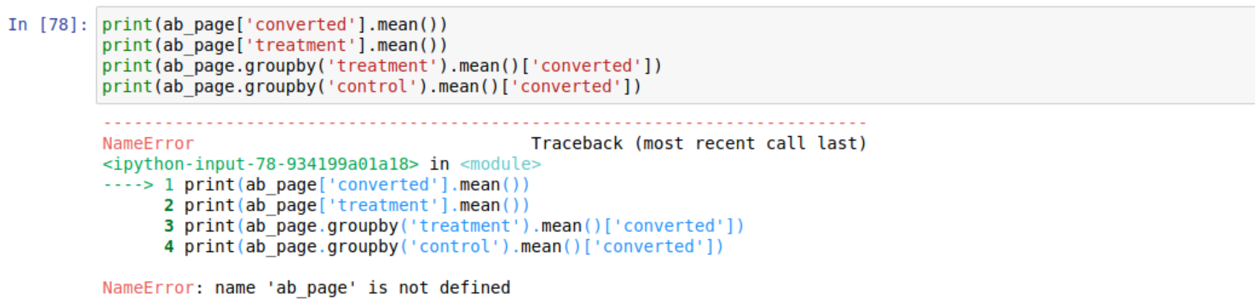
well commented and formatted code along with correct interpretations , this is one of the best projects i've ever reviewed really , well organized reasoned analysis .your previous career experience is concrete here really. you just missed one item (for sake of fulfilling rubrics) which am sure won't take you minutes to complete ,you already did it all right ,i left you some hints hope they help you .

Keep it up , you almost DONE 💪

Code Quality

All code cells can be run without error.

[**Suggestion**] : Part 3 , this cell produces error , but i guess it was for your own analysis , i guess you meant *df2* you may forgot to save work



```
In [78]: print(ab_page['converted'].mean())
         print(ab_page['treatment'].mean())
         print(ab_page.groupby('treatment').mean()['converted'])
         print(ab_page.groupby('control').mean()['converted'])

---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-78-934199a01a18> in <module>
----> 1 print(ab_page['converted'].mean())
      2 print(ab_page['treatment'].mean())
      3 print(ab_page.groupby('treatment').mean()['converted'])
      4 print(ab_page.groupby('control').mean()['converted'])

NameError: name 'ab_page' is not defined
```

Docstrings, comments, and variable names enable readability of the code.

-very well commented and formatted code 👌

-When possible, it is always more computationally efficient to use numpy built-in operations over explicit for loops. The short reason is that numpy -based operations attack a computational problem based on vectors by computing large chunks simultaneously.
Additionally, using loops to simulate 10000 can take a considerable amount of time vs using numpy
for example :

```
new_converted_simulation = np.random.binomial(n_new, p_new, 10000)/n_new
```

```
old_converted_simulation = np.random.binomial(n_old, p_old, 10000)/n_old
```

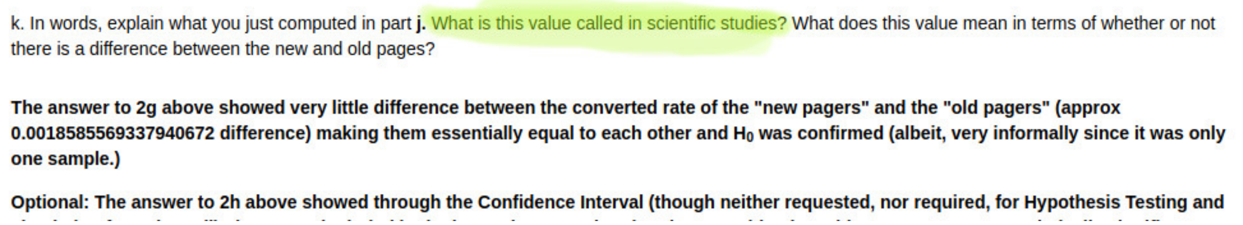
```
p_diffs = new_converted_simulation - old_converted_simulation
```

Statistical Analyses

All results from different analyses are correctly interpreted.

For all numeric values, you should provide the correct results of the analysis.

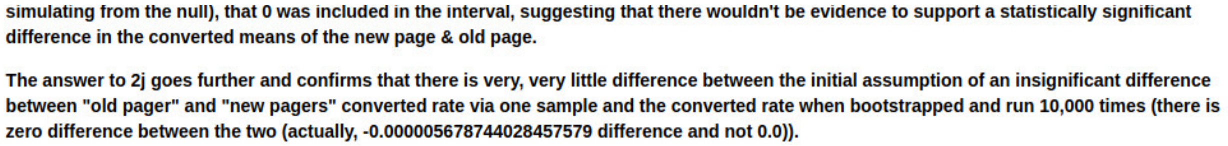
[**Suggestion**] Part II (k) kindly mention what you just computed and what's is this value called in scientific terms ?



k. In words, explain what you just computed in part **j.** What is this value called in scientific studies? What does this value mean in terms of whether or not there is a difference between the new and old pages?

The answer to 2g above showed very little difference between the converted rate of the "new pagers" and the "old pagers" (approx 0.0018585569337940672 difference) making them essentially equal to each other and H_0 was confirmed (albeit, very informally since it was only one sample.)

Optional: The answer to 2h above showed through the Confidence Interval (though neither requested, nor required, for Hypothesis Testing and



simulating from the null), that 0 was included in the interval, suggesting that there wouldn't be evidence to support a statistically significant difference in the converted means of the new page & old page.

The answer to 2j goes further and confirms that there is very, very little difference between the initial assumption of an insignificant difference between "old pager" and "new pagers" converted rate via one sample and the converted rate when bootstrapped and run 10,000 times (there is zero difference between the two (actually, -0.000005678744028457579 difference and not 0.0)).

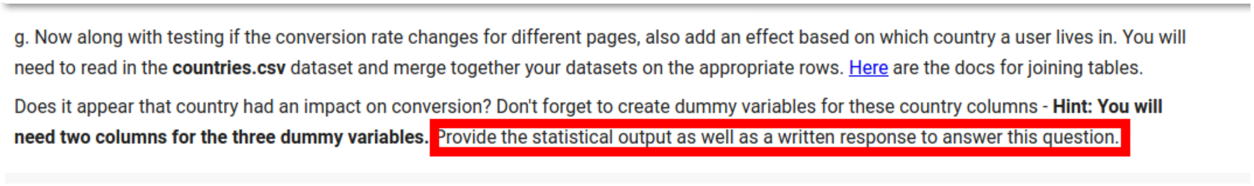
[**Required**] Part II (m): wrong p-value , looking to your *nobs* and *counts* , your *stats.proportions_ztest* should include the *alternative* argument set to *'smaller'* , like so :

```
count = np.array([convert_old, convert_new])
nobs = np.array([n_old, n_new])
stat, pval = proportions_ztest(count, nobs,alternative='smaller')
```

now you'll get correct p-value

[**Awesome**] Part III (e) :well done reasoning for different P-value obtained from Part II. 👏 , the regression model is set up as two tailed test where as in part II it was one tailed or one sided test .

[**Required**] Part III (g) : you successfully joined the tables and created the dummy variables , however you need to instantiate the model here to put **written answer** for the question **Does it appear that country had an impact on conversion?** as shown in attached screenshot ,in fact it's what you did in question (h) so move those cells here and answer the question .



g. Now along with testing if the conversion rate changes for different pages, also add an effect based on which country a user lives in. You will need to read in the **countries.csv** dataset and merge together your datasets on the appropriate rows. Here are the docs for joining tables.

Does it appear that country had an impact on conversion? Don't forget to create dummy variables for these country columns - **Hint: You will need two columns for the three dummy variables.** Provide the statistical output as well as a written response to answer this question.

[**Required**] Part III (h): your conclusions are correct however this is little tricky as we here like to find **if there is an interaction between page and country to see if there significant effects on conversion**. so you need to create additional interaction columns .

hint:

```
df2['UK_ab_page'] = df2['UK'] * df2['ab_page']
df2['CA_ab_page'] = df2['CA'] * df2['ab_page']
```

fit the model again with the new columns ,provide results and put your conclusion according to obtained results .

Conclusions should include not only statistical reasoning, but also practical reasoning for the situation.

RESUBMIT

DOWNLOAD PROJECT

Best practices for your project resubmission

Ben shares 5 helpful tips to get you through revising and resubmitting your project.

Watch Video (3:01)

RETURN TO PATH