**Split Test Analysis**

**Assumptions:**

1. Since issuing of the quotes is one of the revenue sources, I would ask if re-issuing of the same quote (a repeat quote) is a paid service or free. If repeat quotes

are free, then it would be good to know how many repeat quotes were issued per each bucket in order to subtract them from the Quotes column and obtain cleaner results.

For simplicity of our calculations, I'll assume all quotes are paid the same.

2. To gauge performance of the split test, I introduced one more column to the output dataset called “**Viewer Quotes Index” (VQI).** It is a measure of quotes issued by an individual viewer per bucket during the tests. For our purposes VQI describes how many quotes are generated from the form views assuming:

A provider visits a quote-form and either sends a

quote or leaves.

**Split Test:**

Formula:

**Viewer Quotes Index =**

**N of quotes / Views / 1 View per Visitor**

Output:

Bucket,Quotes,Views,"Viewer Quotes Index"

Baseline,32,595, 0.054

Variation 1,30,599, 0.05

Variation 2,18,622, 0.029

Variation 3,51,606, 0.084

Variation 4,38,578, 0.066

Winner is Variation 3  
Viewer Quotes Index = 0.084

Source Code for the test:

#

# Roman Gekhman -

# Galvanize Coding Exercise #3 - 7/3/18

# [roman.gekhman@gmail.com](mailto:roman.gekhman@gmail.com)

#  
#developed / tested on Python 3.6

import csv

#common vars

\_strVQIdx = "'Viewer Quotes Index'"

\_lstHeader = ["Bucket","Quotes","Views"]

with open("out\_data.csv", 'w', newline='') as outfile:

outWriter = csv.writer(outfile, delimiter=',')

with open("in\_data.csv", 'r') as infile:

inReader = csv.reader(infile, delimiter=',')

for line in inReader:

line = [x.strip(' ') for x in line]

if line == \_lstHeader:

outfile.write("\nOutput: \n")

line.append(\_strVQIdx)

outWriter.writerow(line)

continue

else:

try:

\_ratio = round(int(line[1])/int(line[2]), 3)

print(line[0], int(line[1]), int(line[2]), \_ratio)

line.append(\_ratio)

outWriter.writerow(line)

except Exception as ex:

print("Error in line : ", ex.args[0])

Input file:

Bucket,Quotes,Views

Baseline,32,595

Variation 1,30,599

Variation 2,18,622

Variation 3,51,606

Variation 4,38,578  
  
Generated output:

Output:

Bucket,Quotes,Views,'Viewer Quotes Index'

Baseline,32,595,0.054

Variation 1,30,599,0.05

Variation 2,18,622,0.029

Variation 3,51,606,0.084

Variation 4,38,578,0.066