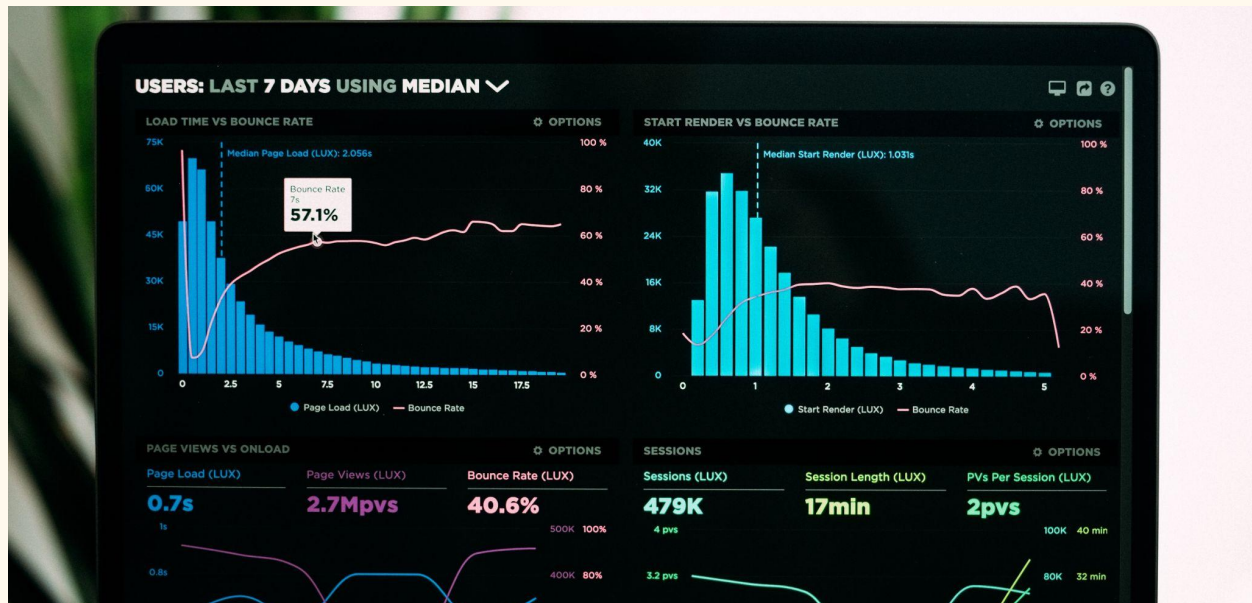


A/B Testing USING PYTHON

By Ogunlaiye Mosopefoluwa



INTRODUCTION

A/B testing is also known as split testing and in simple terms, it means when two or more versions of a variable e.g a web page or a web feature, are used on different users to determine which version has the best impact and drives sales/business metrics.

A/B testing is important for website optimization because it eliminates guesswork and enables one to make data backed decisions. In A/B testing, 'A' refers to the original version or the 'control version' while 'B' refers to the new version or the 'variation' of the original test variable.

This project attempts to analyze two datasets containing data from an original version and a variation of a website's marketing strategy/campaign. The best version is the version that moves the business metrics forward in the best direction and is also known as the 'winner'.

The analysis shows the following:

- The relationship between the spend and the number of impressions
- A comparison of the number of searches done on the website from both campaigns
- The total number of webclicks
- An analysis of the content viewed
- The total products added to cart
- Total spend on both campaigns
- An analysis of total purchases made by users in both campaigns
- The relationship between the number of clicks and the content viewed in both campaigns
- The relationship between the content views and the number of items added to cart in both campaigns
- The relationship between the number of items added to cart and the number of purchases in both campaigns

Data source:

The data source and project idea was gotten from [kaggle](#)

Methodology:

First of all I imported all the necessary python libraries and loaded the two data sets into a jupyter notebook. The two datasets we explored are:

- The Control dataset
- The Test data set

After viewing the dataset, it was cleaned to a point where there were no issues for analysis, then the data was analyzed. To see the technicalities of the data cleaning process visit my github and check the jupyter notebook .ipynb file [here](#).

The following are the results of the analysis.

The relationship between the spend and the number of impressions:

The python syntax:

```

Now that the data is clean enough we can carry out A/B testing to find out which marketing strategy is the best. We first use a scatter plot to checkout the relationship between the spend and number of impressions

In [83]: 1 scatter_plot = px.scatter(data_frame = anb_data,
2                                     x = 'Number of Impressions',
3                                     y = 'Spend(USD)',
4                                     size = 'Spend(USD)',
5                                     color = 'Campaign Name',
6                                     trendline = 'ols')
7 scatter_plot.show()

```

The resulting visualization:



Observations:

We can see that the Control campaign is the better marketing strategy in this regard, because it has more impressions than the Test campaign. The highest number of impressions for the Control campaign is 145,248 impressions and the total spend is 1,875(USD). While the highest number of impressions for the Test Campaign is 133,771 impressions and the total spend is 3,112(USD). We can

see from this figure that we spend more to get less impressions with the Test campaign while we spend less to get more impressions with the Control campaign. Here, the Control campaign wins.

With the use of a pie chart and a bar chart, we will now compare the number of searches done on the website from both campaigns

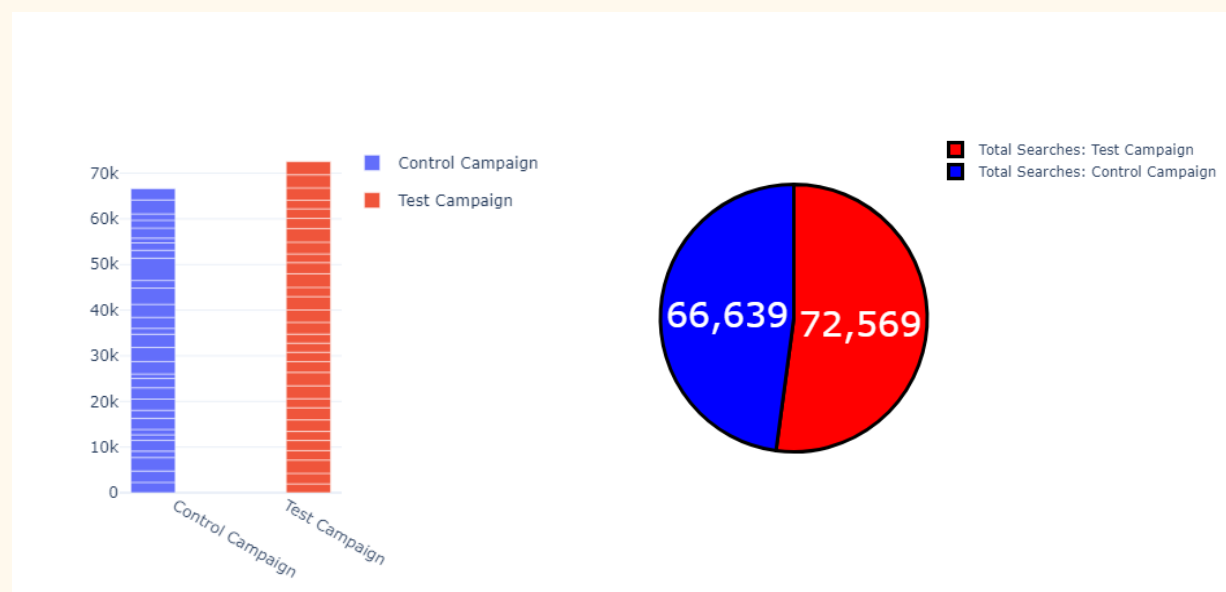
A comparison of the number of searches done on the website from both campaigns:

The python syntax:

With the use of a pie chart we will now compare the number of searches done on the website from both campaigns

```
in [66]: 1 #piechart
2 label = ['Total Searches: Control Campaign', 'Total Searches: Test Campaign']
3 counts = [sum(control_data['Number of Searches']), sum(test_data['Number of Searches'])]
4 colors = ['blue', 'red']
5 piechart = go.Figure(data = [go.Pie(labels = label, values = counts)])
6 piechart.update_layout(title_text = 'Control vs Test: Searches')
7 piechart.update_traces(hoverinfo = 'label + percent', textinfo = 'value', textfont_size = 30, marker = dict(colors = colors,
8 line = dict(color = 'black', width = 3)))
9
10 #bar chart
11 barchart = px.bar(anb_data, x = 'Campaign Name', y = 'Number of Searches', color = 'Campaign Name')
12 barchart.update_layout(title = 'control vs test: Searches')
13
14 #side by side positioning
15 widgets.HBox(
16 [ go.FigureWidget(barchart.data, layout = {'width':420, 'height':450}),
17   go.FigureWidget(piechart.data, layout = {'width':580, 'height':500}),
18 ])
```

The resulting visualization:



Observations:

The Test campaign has more web searches with 72,569 (52.1% of the total searches) while the Control campaign has 66,639 web searches 47.9% of the total searches. Here the Test campaign is more effective and is the winner.

Next, let's analyze the total number of web clicks.

The total number of webclicks:

The python syntax:

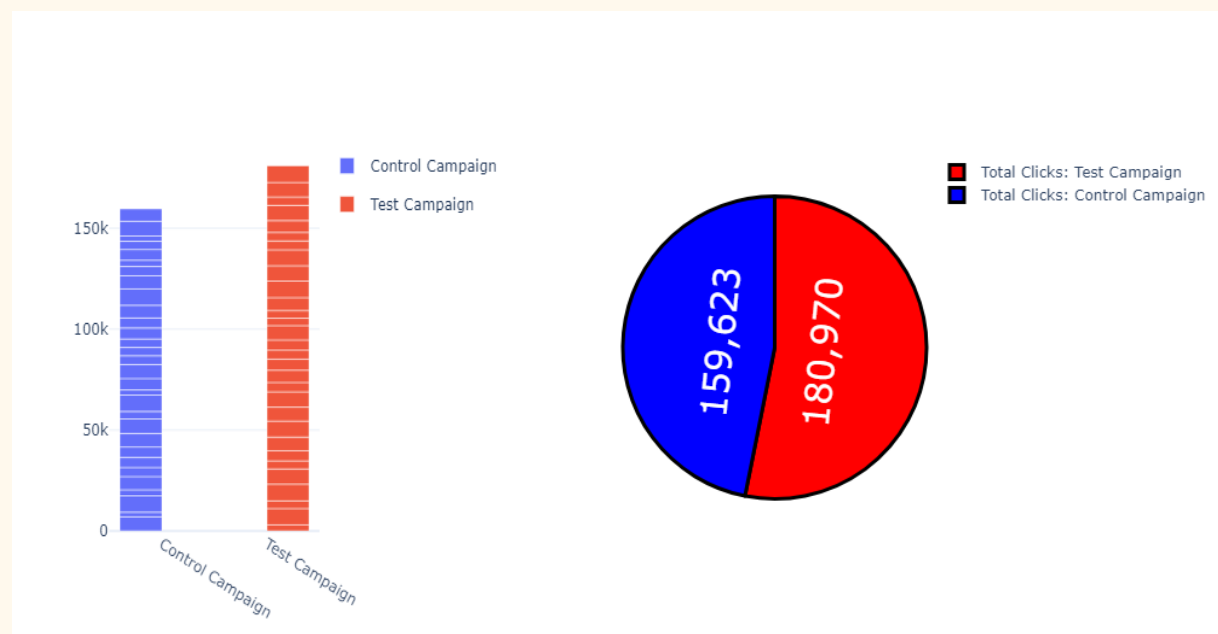
```

Next, lets analyze the total number of web clicks

In [74]: 1 #piechart
2 label = ['Total Clicks: Control Campaign', 'Total Clicks: Test Campaign']
3 counts = [sum(control_data['Number of Website Clicks']), sum(test_data['Number of Website Clicks'])]
4 colors = ['blue', 'red']
5 piechart = go.Figure(data = [go.Pie(labels = label, values = counts)])
6 piechart.update_layout(title_text = 'Control vs Test: Clicks')
7 piechart.update_traces(hoverinfo = 'label + percent', textinfo = 'value', textfont_size = 30, marker = dict(colors = colors,
8 line = dict(color = 'black', width = 3)))
9
10 #bar chart
11 barchart = px.bar(anb_data, x = 'Campaign Name', y = 'Number of Website Clicks', color = 'Campaign Name')
12 barchart.update_layout(title = 'control vs test: Clicks')
13
14 #side by side positioning
15 widgets.HBox(
16 [ go.FigureWidget(barchart.data, layout = {'width':430, 'height':470}),
17   go.FigureWidget(piechart.data, layout = {'width':580, 'height':500}),
18 ]
19 )

```

The resulting visualization:



Observations:

The Test campaign generated more clicks with 180,970(53.1% of total clicks). While the Control campaign generated 159,623(46.9% of total clicks) clicks. In this regard, the test campaign is the clear winner.

Next I'll analyze the content viewed

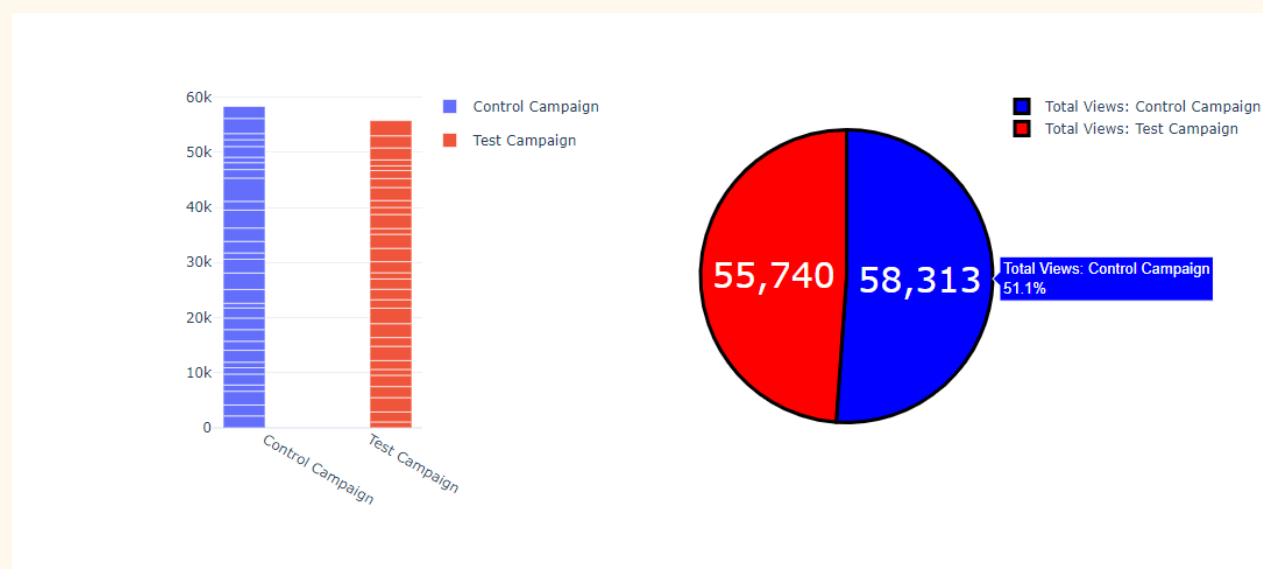
An analysis of the content viewed:

The python syntax:

```
Next I'll analyse the content viewed

In [75]: 1 #piechart
2 label = ['Total Views: Control Campaign', 'Total Views: Test Campaign']
3 counts = [sum(control_data['Number of Viewed Content']), sum(test_data['Number of Viewed Content'])]
4 colors = ['blue', 'red']
5 piechart = go.Figure(data = [go.Pie(labels = label, values = counts)])
6 piechart.update_layout(title_text = 'Control vs Test: Viewed Content')
7 piechart.update_traces(hoverinfo = 'label + percent', textinfo = 'value', textfont_size = 30, marker = dict(colors = colors,
8 line = dict(color = 'black', width = 3)))
9
10 #bar chart
11 barchart = px.bar(anb_data, x = 'Campaign Name', y = 'Number of Viewed Content', color = 'Campaign Name')
12 barchart.update_layout(title = 'control vs test: Views')
13
14 #side by side positioning
15 widgets.HBox(
16 [ go.FigureWidget(barchart.data, layout = {'width':430, 'height':470}),
17   go.FigureWidget(piechart.data, layout = {'width':580, 'height':500}), |
18 ])
```

The resulting visualization:



Observations:

The Control campaign has a lot more content views with 58,313 views(51.1% of total content viewed) even though it had less clicks, while the Test campaign has 55,740 content views(48.9% of total content viewed). Here the Control campaign is the winner

Let's view the amount of products added to the cart

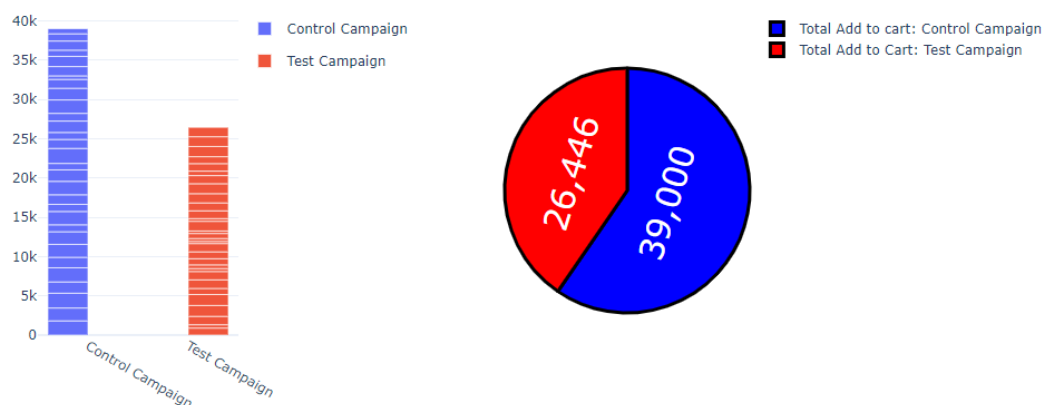
The total products added to cart:

The python syntax:

```
Lets view the amount of products added to the cart

In [76]: 1 #piechart
2 label = ['Total Add to cart: Control Campaign', 'Total Add to Cart: Test Campaign']
3 counts = [sum(control_data['Number of Add to Cart']), sum(test_data['Number of Add to Cart'])]
4 colors = ['blue', 'red']
5 piechart = go.Figure(data = [go.Pie(labels = label, values = counts)])
6 piechart.update_layout(title_text = 'Control vs Test: Add to Cart')
7 piechart.update_traces(hoverinfo = 'label + percent', textinfo = 'value', textfont_size = 30, marker = dict(colors = colors,
8 line = dict(color = 'black', width = 3)))
9 #bar chart
10 barchart = px.bar(anb_data, x = 'Campaign Name', y = 'Number of Add to Cart', color = 'Campaign Name')
11 barchart.update_layout(title = 'control vs test: Add to Cart')
12
13 #side by side positioning
14 widgets.HBox(
15 [ go.FigureWidget(barchart.data, layout = {'width':430, 'height':470}),
16   go.FigureWidget(piechart.data, layout = {'width':580, 'height':500}),
17 ])
```

The resulting visualization:



Observations:

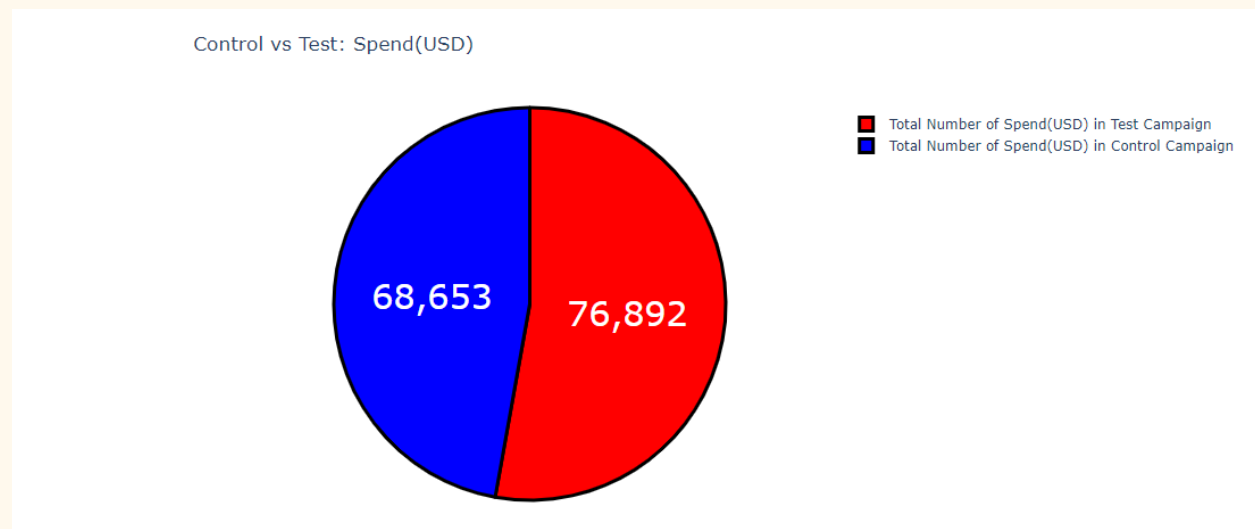
Even though the website clicks for the control campaign were lower than the test campaign, a lot more people added to cart under the control campaign when compared to the test campaign. Here, Control wins.

Total spend on both campaigns:

The python syntax:

```
In [77]: 1 label = ['Total Number of Spend(USD) in Control Campaign', 'Total Number of Spend(USD) in Test Campaign']
2 counts = [sum(control_data['Spend(USD)']), sum(test_data['Spend(USD)'])]
3 colors = ['blue', 'red']
4 pie_chart = go.Figure(data = [go.Pie(labels = label, values = counts)])
5 pie_chart.update_layout(title_text = 'Control vs Test: Spend(USD)')
6 pie_chart.update_traces(hoverinfo = 'label + percent', textinfo = 'value', textfont_size = 30, marker = dict(colors = colors
7                             line = dict(color = 'black', width = 3)))
8 pie_chart.show()
```

The resulting visualization:



Observations:

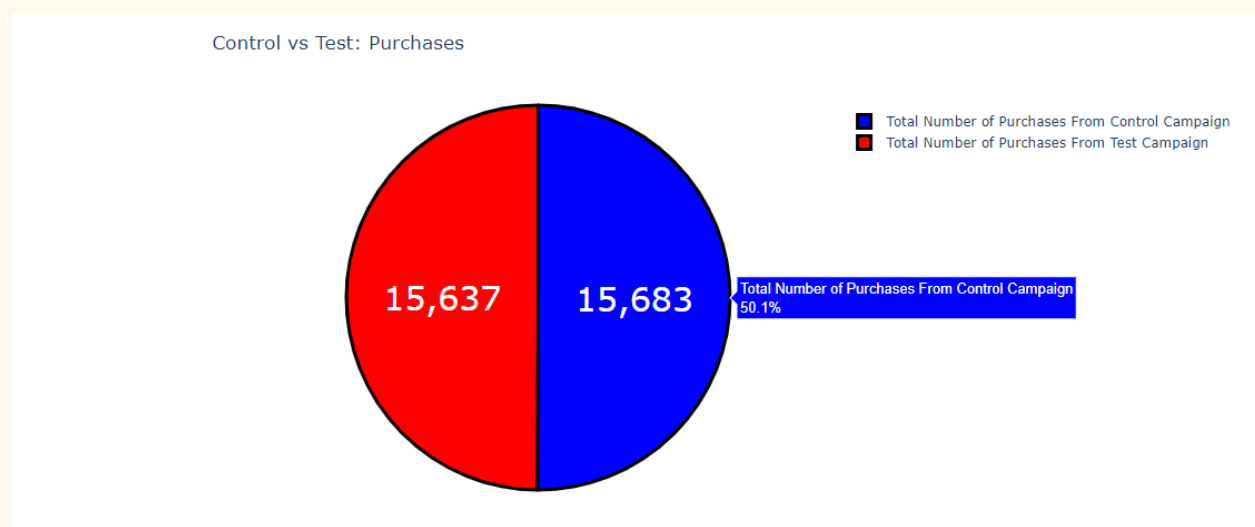
More money was spent on the test campaign than the control campaign. Here the control campaign is the winner because it has more items added to cart and more content views making it more efficient than the test campaign.

An analysis of total purchases made by users in both campaigns:

The python syntax:

```
In [78]: 1 label = ['Total Number of Purchases From Control Campaign', 'Total Number of Purchases From Test Campaign']
2 counts = [sum(control_data['Number of Purchase']), sum(test_data['Number of Purchase'])]
3 colors = ['blue', 'gold']
4 pie_chart = go.Figure(data = [go.Pie(labels = label, values = counts)])
5 pie_chart.update_layout(title_text = 'Control vs Test: Purchases')
6 pie_chart.update_traces(hoverinfo = 'label + percent', textinfo = 'value', textfont_size = 30, marker = dict(colors = colors
7                             line = dict(color = 'black', width = 3)))
8 pie_chart.show()
```

The resulting visualization:



Observations:

The amount of purchases from the two campaigns are surprisingly almost the same with a 1.1% difference. However, the control campaign wins because it has a lower cost spent on marketing and a higher sales on product even though the difference is minimal.

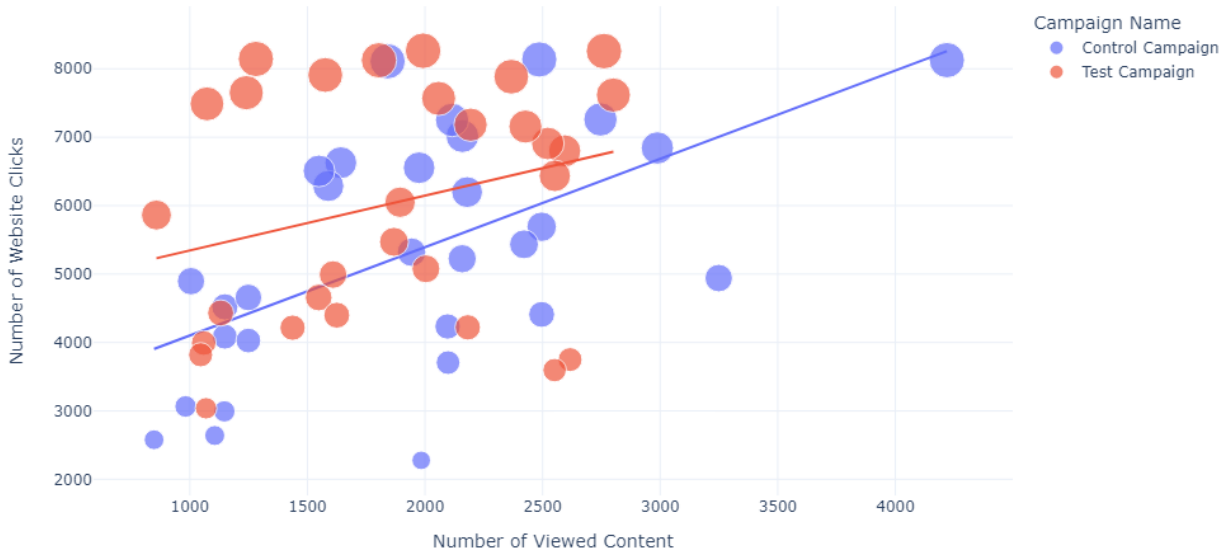
Now I'll check which marketing strategy/campaign converts more by first of all looking at the relationship between the number of clicks and the content viewed from both campaigns

The relationship between the number of clicks and the content viewed in both campaigns :

The python syntax:

```
In [44]: 1 scatter_plot = px.scatter(data_frame = anb_data,
2                               x = 'Number of Viewed Content',
3                               y = 'Number of Website Clicks',
4                               size = 'Number of Website Clicks',
5                               color = 'Campaign Name',
6                               trendline = 'ols')
7 scatter_plot.show()
```

The resulting visualization:



Observations:

Website clicks are higher in the test campaign but the number of content views are higher for the control campaign. Therefore the Control campaign performs better.

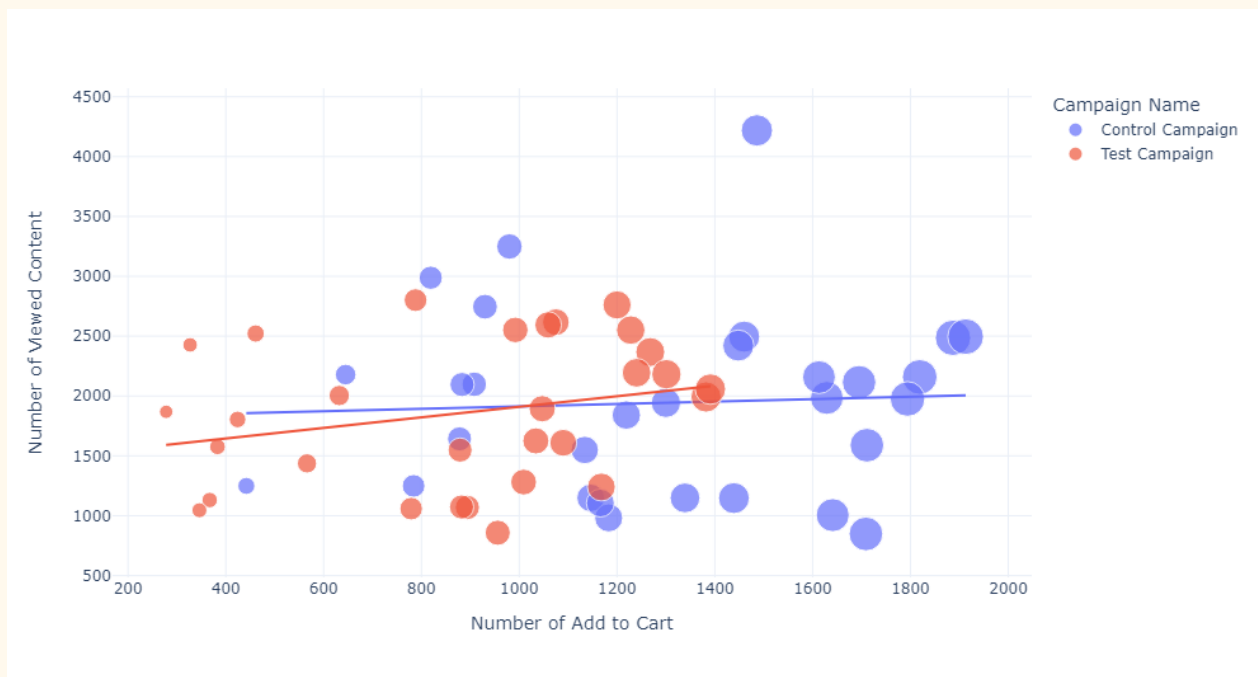
Next let's compare the relationship between the number of content views and the number of items added to cart in both campaigns

The relationship between the content views and the number of items added to cart in both campaigns:

The python syntax:

```
In [45]: 1 scatter_plot = px.scatter(data_frame = anb_data,
2                               x = 'Number of Add to Cart',
3                               y = 'Number of Viewed Content',
4                               size = 'Number of Add to Cart',
5                               color = 'Campaign Name',
6                               trendline = 'ols')
7 scatter_plot.show()
```

The resulting visualization:



Observations:

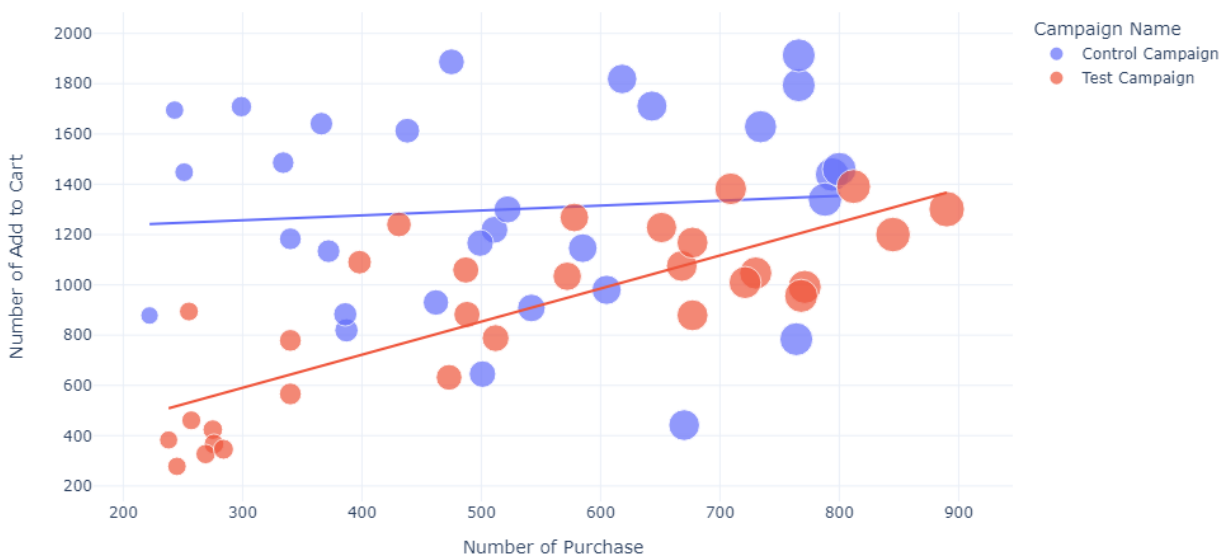
The control campaign performs better because it has a higher number of add to cart and content views. Next we observe the number of items added to cart and the number of purchases from both campaigns

The relationship between the number of items added to cart and the number of purchases in both campaigns:

The syntax:

```
In [46]: 1 scatter_plot = px.scatter(data_frame = anb_data,
2                               x = 'Number of Purchase',
3                               y = 'Number of Add to Cart',
4                               size = 'Number of Purchase',
5                               color = 'Campaign Name',
6                               trendline = 'ols')
7 scatter_plot.show()
```

The resulting visualization:



Observations:

Even though the control campaign has a higher number of overall purchases and even items added to cart, the above visualization shows that the test campaign has a higher conversion rate i.e the number of items in the cart that eventually converts to actual purchases is higher in the test campaign.

Conclusion:

The control campaign has a higher rate of content viewed, it has more items added to cart, it resulted in more purchases and is the least costly of the two campaigns. However, the test campaign also has the advantage of having a higher conversion rate. Like I observed above, the test campaign has more items in the cart being converted to purchases (added to cart: 26,446/purchases: 15,637) unlike the control campaign which has less items in the cart that converted to purchases (added to cart: 39,000/purchases: 15,683) . This means that the control campaign can be used to market multiple products to a wider audience while the test campaign can be used to market specific products to a specific audience. Overall, the control campaign wins!