

Casino AB Test - Megapixel Entertainment Limited Casinos

Olamide_Adu

2023-09-01

Contents

1	Project Description	2
1.1	Objectives	2
1.1.1	Hypothesis	2
2	Experiment Design	2
2.1	Experimental Setup:	3
3	Data Collection	3
4	Data Analysis	3
4.1	Investigate the data structure	4
4.2	Missing Values	4
4.3	Data structure	4
4.4	Duplicate Values	5
4.5	Data Manipulation	5
4.6	Conversion Rate Per Group	6
4.7	Chi-Square Analysis	6
4.7.1	Observed values	7
4.7.2	Expected values	8
4.7.3	Chi-square statistics and p-value	8

1 Project Description



The A/B Test experiment for Megapixel Entertainment Limited provides a comprehensive overview of a series of experiments conducted to optimize player conversion and enhance the overall player experience on our online gaming platform. This report highlights the key experiments, methodologies, and outcomes that have contributed to data-driven decision-making at Megapixel Entertainment Ltd Online Casino.

1.1 Objectives

The primary objective of this A/B test was to assess the impact of a redesigned homepage on player conversion rates at Megapixel Online Casino. We aimed to create a more engaging and user-friendly homepage to encourage new players to sign up and experience our online gaming platform.

1.1.1 Hypothesis

For this experiment, we have the following hypothesis:

- Null Hypothesis (H0):** The test group and their conversion are independent at 0.05 confidence interval.
- **Alternative Hypothesis (H1):** The test group and their conversion are not independent at 0.05 confidence interval.

2 Experiment Design

Two different homepage were designed for the casino, this pages are Treatment A and Treatment B

2.1 Experimental Setup:

We randomly assign visitors to the Megapixel Entertainment Ltd Casino homepage into two test groups:

- A (Treatment A): Visitors are exposed to Treatment A, the first homepage design variant.
- B (Treatment B): Visitors are shown Treatment B, the second homepage design variant.

3 Data Collection

The data for this experiment was provided by Joshua Schnessl. The Key data point collected are: - UserID: the userid of the visitor,

- Brand: the casino brand,
- Test Group: the group the visitor is randomly assigned to, and
- Converted: user convert yes or no, yes is 1 and no is 0.

4 Data Analysis

Import the data and load the necessary libraries

```
## Load the library
library(tidyverse) ## All tidyverse package for visualization and data manipulation and management
library(readxl) ## Load excel files
library(ggthemes) ## Extra themes for ggplot
library(broom) ## Convert statistical objects into data frames (tibbles)
library(gplots) ## Extra plots for ggplot
library(corrplot) #For visualizing correlation matrix

## Import the data
megapixel <- read_excel("/home/xrander/Documents/Job Application/Megapixel/Statistics Assessment Data.xl")
```

A preview of the data

```
head(megapixel, 10)
```

```
## # A tibble: 10 x 4
##   UserID Brand 'Test Group' Converted
##   <dbl> <chr>   <chr>         <dbl>
## 1 7573340 Casino A B           1
## 2 5465339 Casino A A           0
## 3 7516520 Casino A B           0
## 4 7616973 Casino A A           1
## 5 7341455 Casino A B           0
## 6 6804145 Casino A B           0
## 7 7339822 Casino A B           1
## 8 5750273 Casino A A           0
## 9 7126556 Casino A B           0
## 10 6478997 Casino A B           1
```

4.1 Investigate the data structure

Data dimension

```
dim(megapixel)
```

```
## [1] 351 4
```

The data is having 351 observations and 4 variables. Next we check for missing values, understand the structure of the data and undertake quick descriptive summary of the data

4.2 Missing Values

```
unique(is.na(megapixel))
```

```
##      UserID Brand Test Group Converted  
## [1,]  FALSE FALSE      FALSE      FALSE
```

There are no missing values.

4.3 Data structure

```
str(megapixel) # data structure
```

```
## tibble [351 x 4] (S3: tbl_df/tbl/data.frame)  
## $ UserID      : num [1:351] 7573340 5465339 7516520 7616973 7341455 ...  
## $ Brand       : chr [1:351] "Casino A" "Casino A" "Casino A" "Casino A" ...  
## $ Test Group: chr [1:351] "B" "A" "B" "A" ...  
## $ Converted  : num [1:351] 1 0 0 1 0 0 1 0 0 1 ...
```

```
summary(megapixel) # descriptive statistics summary
```

```
##      UserID      Brand      Test Group      Converted  
## Min.   :4513769 Length:351 Length:351 Min.    :0.0000  
## 1st Qu.:6841147 Class :character Class :character 1st Qu.:0.0000  
## Median :7274405 Mode  :character Mode  :character Median :0.0000  
## Mean   :7024352          Mean   :0.4131  
## 3rd Qu.:7558878          3rd Qu.:1.0000  
## Max.   :7818143          Max.    :1.0000
```

It will be better to convert Test Group and Converted variables to factors or categorical data.

Next I will investigate each columns.

```
length(unique(megapixel$UserID))
```

```
## [1] 351
```

```
length(unique(megapixel$Brand))
```

```
## [1] 1
```

```
length(unique(megapixel$`Test Group`))
```

```
## [1] 2
```

```
length(unique(megapixel$Converted))
```

```
## [1] 2
```

4.4 Duplicate Values

Investigate to see if there are duplicate entries.

```
unique(duplicated(megapixel))
```

```
## [1] FALSE
```

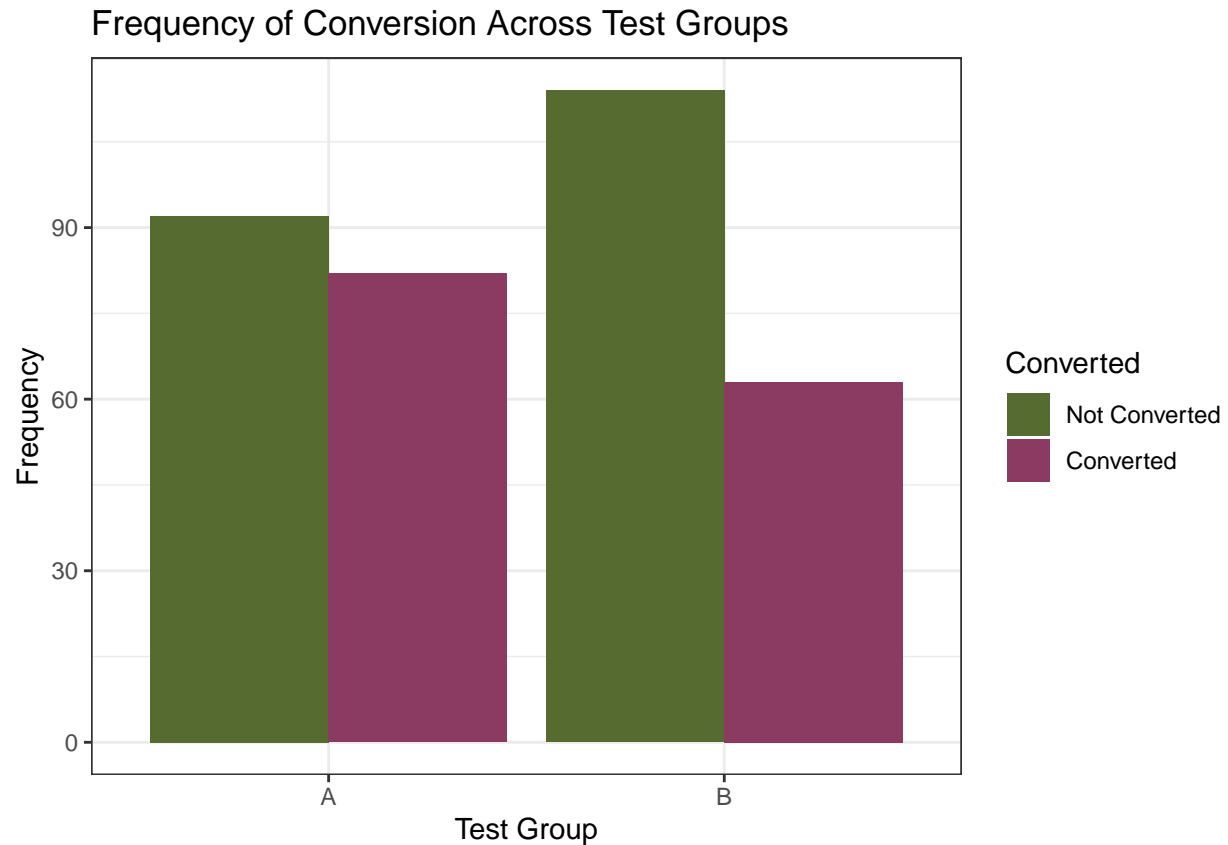
4.5 Data Manipulation

```
megapixel <- megapixel %>%  
  mutate(`Test Group` = factor(`Test Group`))  
  
str(megapixel)
```

```
## tibble [351 x 4] (S3: tbl_df/tbl/data.frame)  
##   $ UserID      : num [1:351] 7573340 5465339 7516520 7616973 7341455 ...  
##   $ Brand       : chr [1:351] "Casino A" "Casino A" "Casino A" "Casino A" ...  
##   $ Test Group: Factor w/ 2 levels "A","B": 2 1 2 1 2 2 2 1 2 2 ...  
##   $ Converted  : num [1:351] 1 0 0 1 0 0 1 0 0 1 ...
```

Test Group is now factor data type.

```
megapixel %>%  
  ggplot(aes(`Test Group`))+  
  geom_bar(aes(fill = factor(Converted)),  
    position = "dodge")+  
  scale_fill_manual(values = c("darkolivegreen", "hotpink4"),  
    labels = c("Not Converted", "Converted"))+  
  labs(x = "Test Group",  
    y = "Frequency",  
    fill = "Converted",  
    title = "Frequency of Conversion Across Test Groups")+  
  theme_bw()
```



4.6 Conversion Rate Per Group

The conversion rate according to groups

```
megapixel %>%
  group_by(`Test Group`) %>%
  summarize(convert = sum(Converted),
            total_in_group = length(Converted)) %>%
  mutate(treatment_conversion_rate = convert/sum(total_in_group))
```

```
## # A tibble: 2 x 4
##   `Test Group` convert total_in_group treatment_conversion_rate
##   <fct>         <dbl>         <int>         <dbl>
## 1 A             82          174          0.234
## 2 B             63          177          0.179
```

4.7 Chi-Square Analysis

Chi-square analysis

```
megapixel_chisqtest <- chisq.test(megapixel$`Test Group`, megapixel$Converted, correct = FALSE)
```

The Result The results from the Chi-square test, excluding the p-value is given below.

```
augment(megapixel_chisqtest)
```

```
## # A tibble: 4 x 9
##   megapixel..Test.Group. megapixel.Converted .observed .prop .row.prop .col.prop
##   <fct>                <fct>          <int> <dbl>    <dbl>    <dbl>
## 1 A                    0              92 0.262    0.529    0.447
## 2 B                    0             114 0.325    0.644    0.553
## 3 A                    1              82 0.234    0.471    0.566
## 4 B                    1              63 0.179    0.356    0.434
## # i 3 more variables: .expected <dbl>, .resid <dbl>, .std.resid <dbl>
```

This will be broken down to aid understanding

4.7.1 Observed values

```
megapixel_chisqtest$observed
```

```
##               megapixel$Converted
## megapixel$`Test Group`    0    1
##               A  92  82
##               B 114  63
```

Contingency table

```
mega_conti_table <- table(megapixel$`Test Group`, megapixel$Converted)
colnames(mega_conti_table) <- c("Not Converted", "Converted")
```

The graphical display of the table is below:

```
balloonplot(t(mega_conti_table), main = "Conversion According to Groups")
```

Conversion According to Groups

		Not Converted	Converted	
	A	92	82	174
	B	114	63	177
		206	145	351

4.7.2 Expected values

```
megapixel_chisqtest$expected
```

```
##               megapixel$Converted
## megapixel$'Test Group'      0      1
##               A 102.1197 71.88034
##               B 103.8803 73.11966
```

4.7.3 Chi-square statistics and p-value

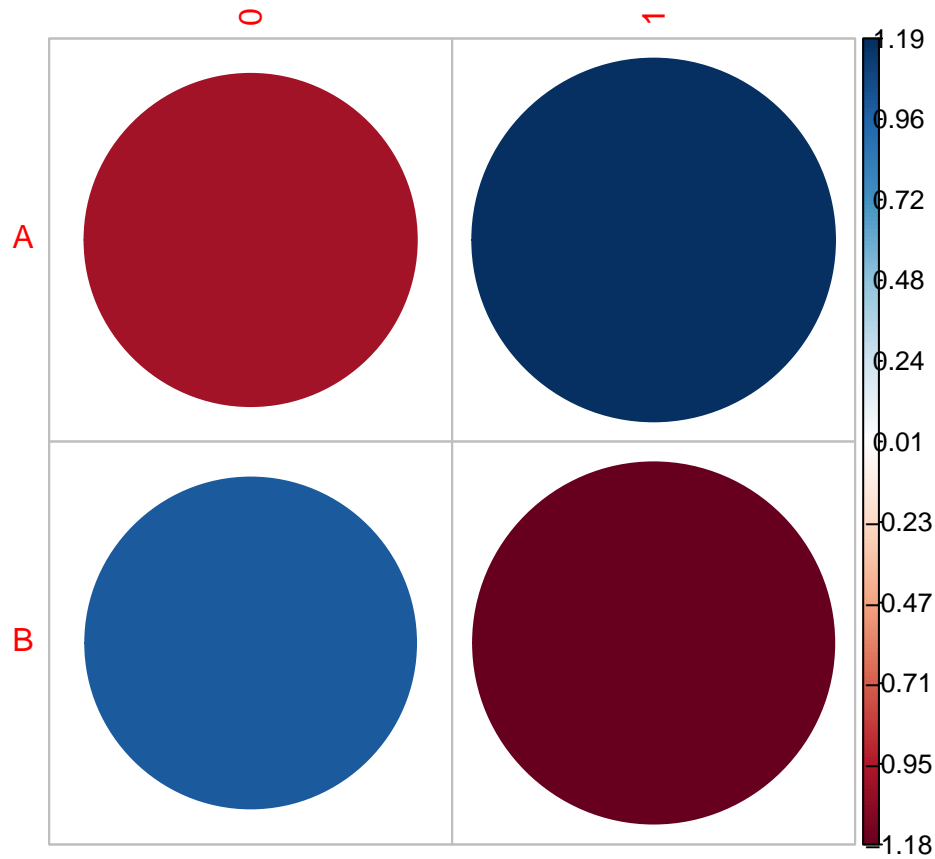
```
tidy(megapixel_chisqtest)
```

```
## # A tibble: 1 x 4
##   statistic p.value parameter method
##   <dbl>    <dbl>    <int> <chr>
## 1      4.81 0.0282         1 Pearson's Chi-squared test
```

Given the result we reject the null-hypothesis and say the groups are dependent.

Next, the contribution of each category will be investigated.


```
corrplot(megapixel_chisqtest$residuals, is.cor = FALSE)
```



For a given cell the size of the circle is proportional to the amount of the cell contribution.

Positive residuals are blue and negative are in red. The result here implies that Group A Treatment is having a strong positive association with being converted and a strong negative association with not being converted. The vice-versa is the case for Group B Treatment.

In conclusion, it can be seen that:

- Group A is strongly associated with being converted.
- Group B is not strongly associated with being converted in comparison to group A.

This implies that Group A website should be implemented, however, this is not final as more analysis can be done.