A/B Testing Implementation in Python

Use statistical tests (Chi-square and t-test) to determine if there's a statistically significant difference in engagement (CTR and conversion) between the two email subject lines.

Created by: Felice Benita

Dataset: Online Learning Platform - Email Engagement Campaign

Context

You're working with an online learning platform that offers various courses and certifications. The goal of this A/B test is to increase user engagement by testing two types of email subject lines aimed at encouraging users to return to the platform and resume a course they previously enrolled in.

Objective of the A/B Test

Analyze which email subject line variant results in a higher:

- Click-through rate (CTR)
- Conversion rate (resuming the course)

Data Structure

User ID: Unique identifier for each user.
Age: Age group (e.g., 18-25, 26-35, etc.) of the user, which could impact engagement levels.
Enrollment Type: Whether the user enrolled for a free trial or paid subscription.
Course Category: The category of the course enrolled in (e.g., Data Science, Marketing, Design).
Previous Engagement Level: Historical data indicating low, medium, or high engagement based on past actions.
Email Variant (A/B): The variant of the subject line used for each user (e.g., "Variant A: 'Complete Your Course in Record Time!' " or "Variant B: 'Your Learning Journey Awaits - Resume Now!'").
Click-Through Rate (CTR): Whether the user clicked the email (0 for no, 1 for yes).
Conversion Rate: Whether the user resumed their course after opening the email (0 for no, 1 for yes).

In [1]:
```
# Import libraries

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn import preprocessing
```

In [2]:
```
# Reading data
df = pd.read_csv('File Dirr/online_learning_email_engagement.csv')
df.head()
```

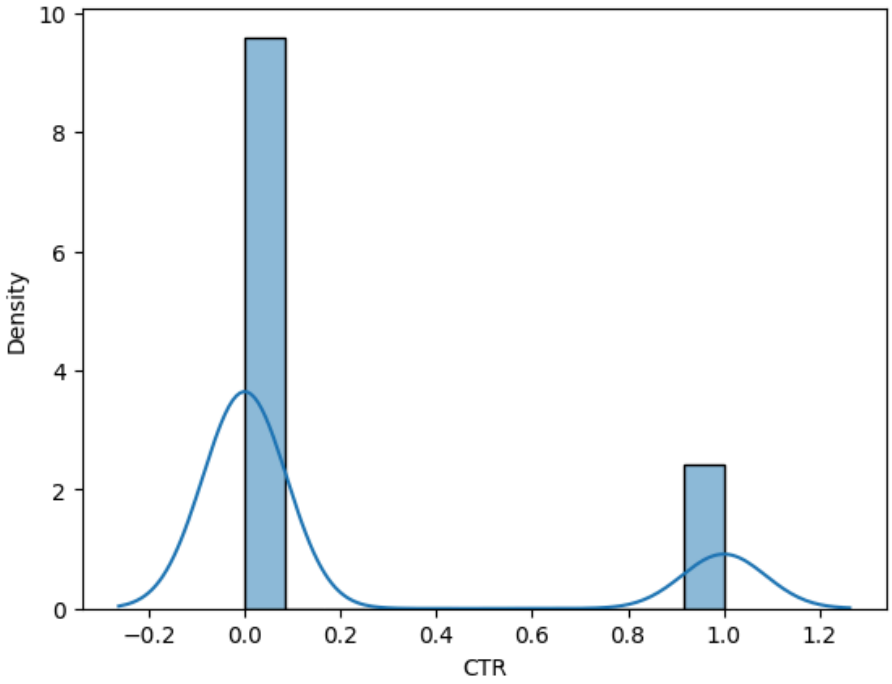
Out[2]:

	User_ID	Age_Group	Enrollment_Type	Course_Category	Previous_Engagement_Level	Email_Variant	CTR	Con
0	User_1	26-35	Paid Subscription	Data Science	High	B	0	
1	User_2	56+	Paid Subscription	Data Science	High	A	1	
2	User_3	18-25	Free Trial	Data Science	Medium	A	1	
3	User_4	56+	Paid Subscription	Marketing	Low	B	0	
4	User_5	36-45	Paid Subscription	Programming	Medium	B	0	

Distribution of CTR and Conversion Rate

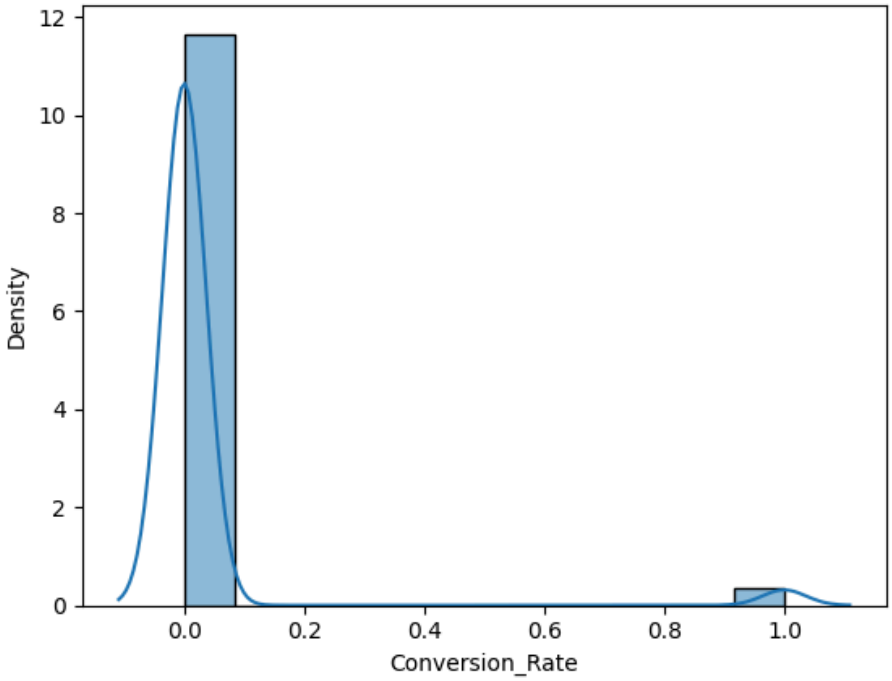
In [3]:
```
sns.histplot(df["CTR"], kde=True, stat="density", kde_kws=dict(cut=3))
```

Out[3]: <Axes: xlabel='CTR', ylabel='Density'>



In [4]:
```
sns.histplot(df["Conversion_Rate"], kde=True, stat="density", kde_kws=dict(cut=3))
```

Out[4]: <Axes: xlabel='Conversion_Rate', ylabel='Density'>



T-Test

In [5]:
```
from scipy.stats import ttest_ind

# Separate the data for each email variant
clicks_A = df[df['Email_Variant'] == 'A']['CTR']
clicks_B = df[df['Email_Variant'] == 'B']['CTR']

conversions_A = df[df['Email_Variant'] == 'A']['Conversion_Rate']
conversions_B = df[df['Email_Variant'] == 'B']['Conversion_Rate']

# Perform t-tests for click-through rate and conversion rate
t_stat_click, p_val_click = ttest_ind(clicks_A, clicks_B, equal_var=False)
t_stat_conversion, p_val_conversion = ttest_ind(conversions_A, conversions_B, equal_var=False)
```

```
# Display the results
print("A/B Testing Results Using T-Test:")
print(f"Click-Through Rate T-Test: t-statistic = {t_stat_click:.3f}, p-value = {p_val_click:.3f}")
print(f"Conversion Rate T-Test: t-statistic = {t_stat_conversion:.3f}, p-value = {p_val_conversion:.3f}

# Interpretation
if p_val_click < 0.05:
    print("The difference in Click-Through Rates between variants A and B is statistically significant.
else:
    print("The difference in Click-Through Rates between variants A and B is not statistically signific

if p_val_conversion < 0.05:
    print("The difference in Conversion Rates between variants A and B is statistically significant.")
else:
    print("The difference in Conversion Rates between variants A and B is not statistically significant
```

```
A/B Testing Results Using T-Test:
Click-Through Rate T-Test: t-statistic = -2.654, p-value = 0.008
Conversion Rate T-Test: t-statistic = -1.880, p-value = 0.060
The difference in Click-Through Rates between variants A and B is statistically significant.
The difference in Conversion Rates between variants A and B is not statistically significant.
```

Slice data by user demographics (age & enrollment type) to see if certain groups respond better to one email variant over another.

In [6]:

```
from scipy.stats import ttest_ind

# Define demographic groups to analyze
demographic_groups = ['Age_Group', 'Enrollment_Type']
results = []

for group in demographic_groups:
    print(f"\nAnalyzing response rates by '{group}' and Email Variant\n")

    # Loop through each unique value in the demographic group
    for val in df[group].unique():
        # Filter data by the current demographic value
        subgroup = df[df[group] == val]

        # Separate data for each email variant within the demographic subgroup
        clicks_A = subgroup[subgroup['Email_Variant'] == 'A']['CTR']
        clicks_B = subgroup[subgroup['Email_Variant'] == 'B']['CTR']

        conversions_A = subgroup[subgroup['Email_Variant'] == 'A']['Conversion_Rate']
        conversions_B = subgroup[subgroup['Email_Variant'] == 'B']['Conversion_Rate']

        # Perform t-tests for click-through rate and conversion rate
        t_stat_click, p_val_click = ttest_ind(clicks_A, clicks_B, equal_var=False)
        t_stat_conversion, p_val_conversion = ttest_ind(conversions_A, conversions_B, equal_var=False)

        # Append results for each demographic value
        results.append({
            'Demographic Group': group,
            'Demographic Value': val,
            'Click-Through Rate T-Statistic': t_stat_click,
            'Click-Through Rate P-Value': p_val_click,
            'Conversion Rate T-Statistic': t_stat_conversion,
            'Conversion Rate P-Value': p_val_conversion
        })

        # Display results
        print(f"'{group}' = {val}")
        print(f"CTR T-Test: t-statistic = {t_stat_click:.3f}, p-value = {p_val_click:.3f}")
        print(f"Conversion Rate T-Test: t-statistic = {t_stat_conversion:.3f}, p-value = {p_val_convers
        print("-" * 50)

# Convert results to DataFrame for easier analysis
results_df = pd.DataFrame(results)
print("\nSummary of t-test results for each demographic group:")
print(results_df)
```

```
Analyzing response rates by 'Age_Group' and Email Variant

'Age_Group' = 26-35
CTR T-Test: t-statistic = -2.187, p-value = 0.029
Conversion Rate T-Test: t-statistic = -0.482, p-value = 0.630
--------------------------------------------------
'Age_Group' = 56+
CTR T-Test: t-statistic = -0.943, p-value = 0.346
Conversion Rate T-Test: t-statistic = -1.546, p-value = 0.123
--------------------------------------------------
'Age_Group' = 18-25
CTR T-Test: t-statistic = -0.372, p-value = 0.710
Conversion Rate T-Test: t-statistic = 0.782, p-value = 0.435
--------------------------------------------------
'Age_Group' = 36-45
CTR T-Test: t-statistic = -1.129, p-value = 0.260
Conversion Rate T-Test: t-statistic = -2.689, p-value = 0.008
--------------------------------------------------
'Age_Group' = 46-55
CTR T-Test: t-statistic = -1.365, p-value = 0.173
Conversion Rate T-Test: t-statistic = -0.470, p-value = 0.639
--------------------------------------------------

Analyzing response rates by 'Enrollment_Type' and Email Variant

'Enrollment_Type' = Paid Subscription
CTR T-Test: t-statistic = -2.356, p-value = 0.019
Conversion Rate T-Test: t-statistic = -1.112, p-value = 0.267
--------------------------------------------------
'Enrollment_Type' = Free Trial
CTR T-Test: t-statistic = -1.420, p-value = 0.156
Conversion Rate T-Test: t-statistic = -1.529, p-value = 0.127
--------------------------------------------------

Summary of t-test results for each demographic group:
  Demographic Group  Demographic Value  Click-Through Rate T-Statistic  \
0         Age_Group              26-35                       -2.187410
1         Age_Group                56+                       -0.943120
2         Age_Group              18-25                       -0.372080
3         Age_Group              36-45                       -1.128988
4         Age_Group              46-55                       -1.364579
5   Enrollment_Type  Paid Subscription                       -2.356180
6   Enrollment_Type         Free Trial                       -1.420145

   Click-Through Rate P-Value  Conversion Rate T-Statistic  \
0                    0.029257                    -0.481923
1                    0.346197                    -1.546003
2                    0.710038                     0.781971
3                    0.259633                    -2.688774
4                    0.173156                    -0.469613
5                    0.018662                    -1.111761
6                    0.155872                    -1.528857

   Conversion Rate P-Value
0                 0.630107
1                 0.123076
2                 0.434763
3                 0.007812
4                 0.638889
5                 0.266523
6                 0.126626
```

Chi-square Test

In [7]:

```python
# Re-import necessary libraries and re-generate the dataset to continue where we left off.
import pandas as pd
import numpy as np
from scipy.stats import chi2_contingency

# Set random seed for reproducibility
np.random.seed(42)

# Parameters
num_users = 1000
```

```
# Generate user IDs
user_ids = np.arange(1, num_users + 1)

# Age groups
age_groups = np.random.choice(['18-25', '26-35', '36-45', '46-55', '56+'], num_users)

# Enrollment type (free trial or paid subscription)
enrollment_types = np.random.choice(['Free Trial', 'Paid Subscription'], num_users, p=[0.6, 0.4])

# Course categories
course_categories = np.random.choice(['Data Science', 'Marketing', 'Design', 'Business', 'Programming']

# Previous engagement level
engagement_levels = np.random.choice(['Low', 'Medium', 'High'], num_users, p=[0.5, 0.3, 0.2])

# Email variant (A or B)
email_variants = np.random.choice(['A', 'B'], num_users)

# Click-through rate (CTR) - Generate some variability based on the email variant and engagement level
click_through_rates = [
    1 if (email == 'A' and np.random.rand() < 0.35) or
         (email == 'B' and np.random.rand() < 0.4) else 0
    for email in email_variants
]

# Conversion rate (if the user resumes the course after clicking)
conversion_rates = [
    1 if (click == 1 and engagement == 'High' and np.random.rand() < 0.5) or
         (click == 1 and engagement != 'High' and np.random.rand() < 0.2) else 0
    for click, engagement in zip(click_through_rates, engagement_levels)
]

# Compile the data into a DataFrame
df = pd.DataFrame({
    'User_ID': user_ids,
    'Age_Group': age_groups,
    'Enrollment_Type': enrollment_types,
    'Course_Category': course_categories,
    'Previous_Engagement_Level': engagement_levels,
    'Email_Variant': email_variants,
    'Click_Through_Rate': click_through_rates,
    'Conversion_Rate': conversion_rates
})

# Summarize data for the A/B test
ab_summary = df.groupby('Email_Variant').agg(
    total_clicks=('Click_Through_Rate', 'sum'),
    total_users=('User_ID', 'count'),
    total_conversions=('Conversion_Rate', 'sum')
).reset_index()

# Calculate click-through rate and conversion rate per email variant
ab_summary['CTR'] = ab_summary['total_clicks'] / ab_summary['total_users']
ab_summary['Conversion_Rate'] = ab_summary['total_conversions'] / ab_summary['total_users']

# Create contingency table for click-through rates
click_contingency = pd.crosstab(df['Email_Variant'], df['Click_Through_Rate'])

# Chi-square test for independence to compare click-through rates
chi2_click, p_click, _, _ = chi2_contingency(click_contingency)

# Create contingency table for conversion rates
conversion_contingency = pd.crosstab(df['Email_Variant'], df['Conversion_Rate'])

# Chi-square test for independence to compare conversion rates
chi2_conversion, p_conversion, _, _ = chi2_contingency(conversion_contingency)

# Display the results
ab_summary[['Email_Variant', 'CTR', 'Conversion_Rate']].round(3), p_click, p_conversion
```

Out[7]:
```
(  Email_Variant    CTR  Conversion_Rate
 0             A  0.379            0.094
 1             B  0.399            0.094,
 0.5593938977973092,
 1.0)
```

Slice data by user demographics (age & enrollment type) to see if certain groups respond better to one email variant over another.

In [8]:
```python
import pandas as pd
from scipy.stats import chi2_contingency

# Grouping data by demographics (e.g., Age Group, Enrollment Type) and Email Variant
demographic_groups = ['Age_Group', 'Enrollment_Type']
results = []

for group in demographic_groups:
    print(f"\nAnalyzing response rates by '{group}' and Email Variant")

    # Group by demographic attribute and email variant
    demographic_summary = df.groupby([group, 'Email_Variant']).agg(
        total_clicks=('Click_Through_Rate', 'sum'),
        total_users=('User_ID', 'count'),
        total_conversions=('Conversion_Rate', 'sum')
    ).reset_index()

    # Calculate CTR and Conversion Rate within each group and email variant
    demographic_summary['CTR'] = demographic_summary['total_clicks'] / demographic_summary['total_users
    demographic_summary['Conversion_Rate'] = demographic_summary['total_conversions'] / demographic_sum
    print(demographic_summary[['Email_Variant', group, 'CTR', 'Conversion_Rate']])

    # Pivot the data to create a contingency table for CTR
    for val in df[group].unique():
        click_contingency = pd.crosstab(df[df[group] == val]['Email_Variant'],
                                        df[df[group] == val]['Click_Through_Rate'])
        conversion_contingency = pd.crosstab(df[df[group] == val]['Email_Variant'],
                                             df[df[group] == val]['Conversion_Rate'])

        # Chi-square tests
        chi2_click, p_click, _, _ = chi2_contingency(click_contingency)
        chi2_conversion, p_conversion, _, _ = chi2_contingency(conversion_contingency)

        # Collect and display results
        results.append({
            'Demographic Group': group,
            'Demographic Value': val,
            'CTR P-Value': p_click,
            'Conversion Rate P-Value': p_conversion
        })

        print(f"\n'{group}' = {val}")
        print(f"CTR p-value: {p_click}, Conversion Rate p-value: {p_conversion}")

# Display the summarized test results
results_df = pd.DataFrame(results)
print("\nSummary of p-values for each demographic group and variant:")
print(results_df)
```

```
Analyzing response rates by 'Age_Group' and Email Variant
  Email_Variant Age_Group       CTR  Conversion_Rate
0             A     18-25  0.311828         0.064516
1             B     18-25  0.444444         0.153846
2             A     26-35  0.360000         0.070000
3             B     26-35  0.433333         0.088889
4             A     36-45  0.372340         0.085106
5             B     36-45  0.333333         0.104167
6             A     46-55  0.402062         0.092784
7             B     46-55  0.422018         0.045872
8             A       56+  0.439252         0.149533
9             B       56+  0.350515         0.072165

'Age_Group' = 46-55
CTR p-value: 0.8818321970598999, Conversion Rate p-value: 0.2900257388380034

'Age_Group' = 56+
CTR p-value: 0.25000392430640644, Conversion Rate p-value: 0.12771287863391959

'Age_Group' = 36-45
CTR p-value: 0.6812261376078861, Conversion Rate p-value: 0.8408535898242047

'Age_Group' = 26-35
CTR p-value: 0.37672678475838584, Conversion Rate p-value: 0.8315668249756706

'Age_Group' = 18-25
CTR p-value: 0.06900009702079542, Conversion Rate p-value: 0.07142961041063417

Analyzing response rates by 'Enrollment_Type' and Email Variant
  Email_Variant    Enrollment_Type       CTR  Conversion_Rate
0             A         Free Trial  0.405498         0.099656
1             B         Free Trial  0.380471         0.127946
2             A  Paid Subscription  0.340000         0.085000
3             B  Paid Subscription  0.424528         0.047170

'Enrollment_Type' = Paid Subscription
CTR p-value: 0.09647525369802354, Conversion Rate p-value: 0.17649093739948726

'Enrollment_Type' = Free Trial
CTR p-value: 0.5913881472247076, Conversion Rate p-value: 0.3423073301227318

Summary of p-values for each demographic group and variant:
  Demographic Group  Demographic Value  CTR P-Value  Conversion Rate P-Value
0         Age_Group              46-55     0.881832                 0.290026
1         Age_Group                56+     0.250004                 0.127713
2         Age_Group              36-45     0.681226                 0.840854
3         Age_Group              26-35     0.376727                 0.831567
4         Age_Group              18-25     0.069000                 0.071430
5   Enrollment_Type  Paid Subscription     0.096475                 0.176491
6   Enrollment_Type         Free Trial     0.591388                 0.342307
```

Created by: Felice Benita