Mobile App A/B Test

Introduction

There is a mobile app with two variations for an enrollment button. A says 'Secure Free Trial', and B says 'Enroll Now'. The goal is to see if changing to B will result in more clicks and boost the company's sales.

In this experiment a one-tailed z-test for comparing two proportions will be used.

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import norm
```

$$H_0: \mu_B \leq \mu_A$$
$$H_1: \mu_B > \mu_A$$

Data

In [2]:
```
def measure_click(ctr):
    return 1 if np.random.uniform(0,1) < ctr else 0

def measure_a():
    return measure_click(ctr=0.005)

def measure_b():
    return measure_click(ctr=0.007)
```

Pilot Study

- This pilot study is set up for a power of 80% and a false positive rate of 5%. The practical significance determined by the company is 0.1%.

In [3]:
```
def design_ab_test():
    def pilot_study(num_pilot_measurements):
        clicked_pre_a = np.array([measure_a() for _ in range(num_pilot_measurements)])
        clicked_pre_b = np.array([measure_b() for _ in range(num_pilot_measurements)])
        sd_1 = np.sqrt(clicked_pre_a.std()**2 + clicked_pre_b.std()**2)
        return sd_1
    sd_1 = pilot_study(1000)
    prac_sig = 0.001
    num_ind = (2.48*sd_1/prac_sig)**2
    return int(num_ind)
```

In [4]:
```
np.random.seed(17)
num_ind = design_ab_test()
num_ind
```

Out[4]:
```
91561
```

- So 91,561 individual measurements are needed to confidently detect a meaningful difference between the two variants.

Trace of the CTR as the A/B Test Runs

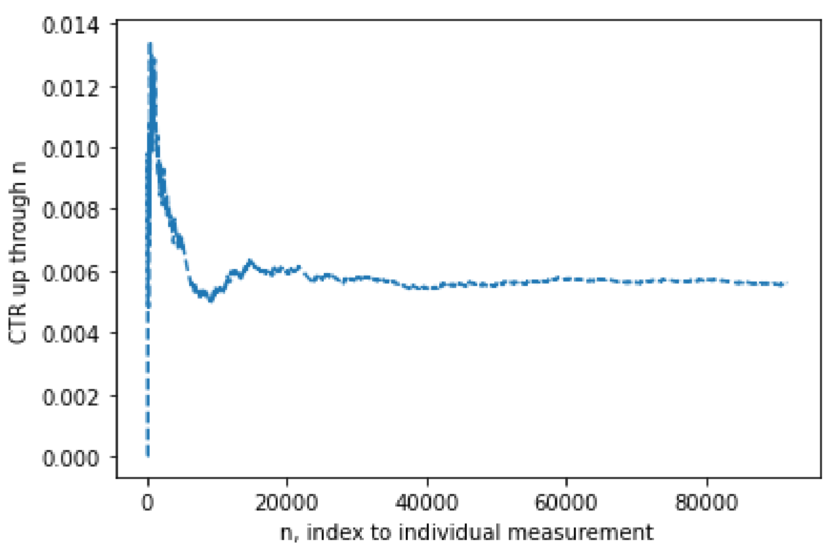
In [5]:
```
def ab_test(num_ind):
    sum_clicks = 0.0
    num_ads = 0.0
    sum_a = 0.0
    num_a = 0
    sum_b = 0.0
    num_b = 0

    ctr_vs_n = []
    ctr_a = []
    ctr_b = []
    for n in range(num_ind):
```

```
        if np.random.uniform(0,1)<0.5:
            clicked=measure_a()
            sum_a+=clicked
            num_a+=1
        else:
            clicked=measure_b()
            sum_b+=clicked
            num_b+=1
        sum_clicks+=clicked
        num_ads+=1
        if num_a>0 and num_b>0:
            ctr_a.append(sum_a/num_a)
            ctr_b.append(sum_b/num_b)
            ctr_vs_n.append(sum_clicks/num_ads)
    return ctr_vs_n,ctr_a,ctr_b
```

In [6]:
```
np.random.seed(17)
ctr_vs_n, ctr_a, ctr_b = ab_test(num_ind)
```

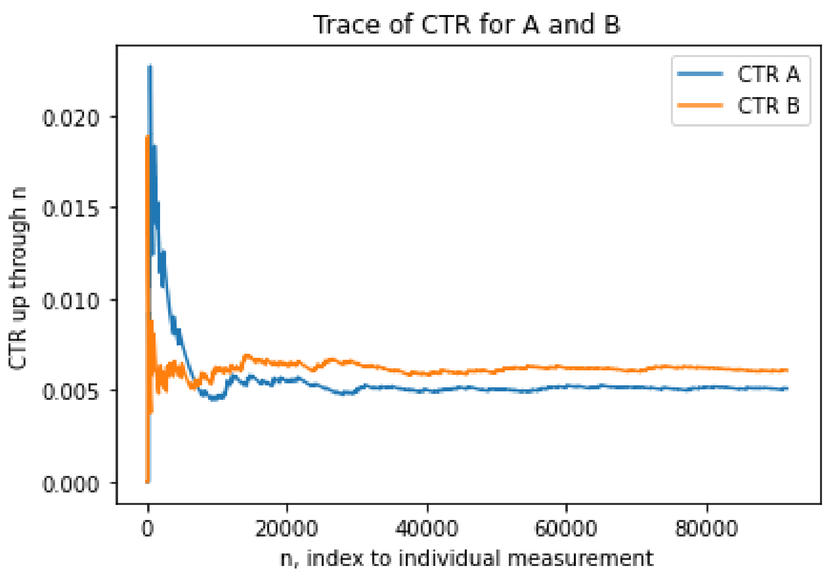
In [7]:
```
#
plt.plot(ctr_vs_n, '--')
plt.xlabel('n, index to individual measurement')
plt.ylabel('CTR up through n');
```



- Since A/B tests randomly choose between variants we can see our ctr is roughly halfway between the true ctr of A and B most of the time.

In [8]:

```
#
plt.plot(ctr_a)
plt.plot(ctr_b)
plt.title('Trace of CTR for A and B')
plt.xlabel('n, index to individual measurement')
plt.ylabel('CTR up through n')
plt.legend(['CTR A', 'CTR B']);
```



- Early on, the ctr of A is higher, but eventually falls below and stays there

---

Z-Test Statistic and P-Value

In [9]:

```
def run_ab_test(num_ind):
    clicked_a = []
    clicked_b = []
    for n in range(num_ind):
        if np.random.uniform(0,1)<0.5:
```

```
            clicked=measure_a()
            clicked_a.append(clicked)
        else:
            clicked = measure_b()
            clicked_b.append(clicked)

    clicked_a = np.array(clicked_a)
    clicked_b = np.array(clicked_b)

    return clicked_a, clicked_b
```

In [10]:
```
np.random.seed(17)
clicked_a, clicked_b = run_ab_test(num_ind)
```

In [11]:
```
def analyze_a_b_test(clicked_a, clicked_b):
    mean_a = clicked_a.mean()
    mean_b = clicked_b.mean()
    std_a = clicked_a.std()
    std_b = clicked_b.std()
    m = mean_b - mean_a
    se = np.sqrt((std_a**2+std_b**2)/num_ind)
    z = m/se

    return z, mean_a, mean_b, std_a, std_b
```

In [12]:
```
z, mean_a, mean_b, std_a, std_b = analyze_a_b_test(clicked_a, clicked_b)
```

In [13]:
```
np.random.seed(17)
clicked_a, clicked_b = run_ab_test(num_ind)
print(z.round(2))
```

```
2.83
```

- The value of the test statistic being greater than the critical of 1.64 indicates strong evidence against the null hypothesis.

In [14]:
```
alpha=0.05
p_value = 1 - norm.cdf(z)
if p_value < alpha:
    print('Reject the null hypothesis')
```

```
Reject the null hypothesis
```

- Because the p value is less than our chosen alpha of 0.05 we reject the null hypothesis and conclude that B performs significantly better than A.

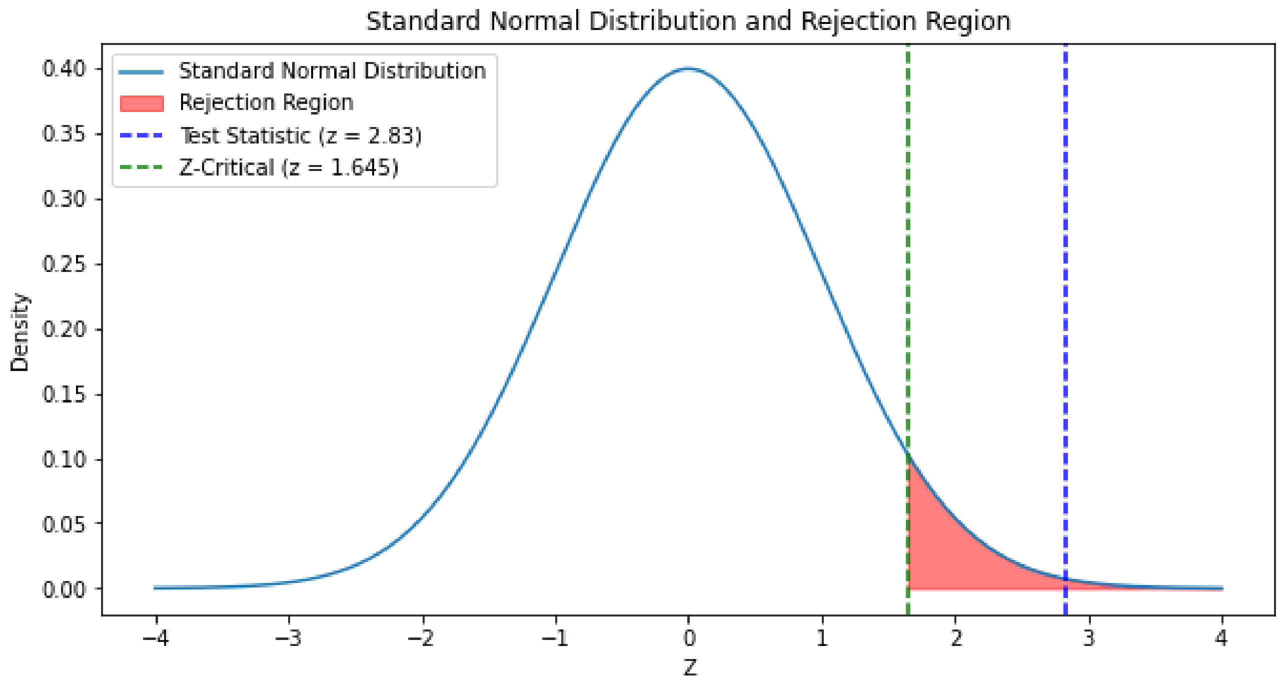
Standard Normal Distribution with Rejection Region

In [15]:

```
x = np.linspace(-4, 4, 1000)
y = norm.pdf(x)

z_critical = 1.645

plt.figure(figsize=(10, 5))
plt.plot(x, y, label="Standard Normal Distribution")
plt.fill_between(x, y, where=(x >= z_critical), color="red", alpha=0.5, label="Rejection Region")
plt.axvline(z, color="blue", linestyle="--", label=f"Test Statistic (z = {z:.2f})")
plt.axvline(z_critical, color="green", linestyle="--", label=f"Z-Critical (z = {z_critical})")
plt.title("Standard Normal Distribution and Rejection Region")
plt.xlabel("Z")
plt.ylabel("Density")
plt.legend()
plt.show()
```



- This graph shows our test statistic, z, being well within the rejecion region.

95% Confidence Interval

In [16]:

```
mean_diff = mean_b - mean_a
std_err = np.sqrt((std_a**2 + std_b**2) / num_ind)

lower_bound = mean_diff - 1.96 * std_err
upper_bound = mean_diff + 1.96 * std_err

print(f"95% Confidence Interval: ({lower_bound:.6f}, {upper_bound:.6f})")
```

```
95% Confidence Interval: (0.000302, 0.001668)
```

- The calculated 95% confidence interval for the difference in click-through rates (CTR) between Variant B and Variant A is (0.000302, 0.001668). While this range is statistically significant (entirely above 0), it partially overlaps with values below the practical significance threshold of 0.001 set by the company.

Conclusion

While the results suggest that Variant B is statistically better than Variant A, the overlap with the region below the practical significance threshold introduces some uncertainty about whether the observed improvement justifies the effort and resources required to implement the change.

Given this, some options might be to:

a. Proceed cautiously with implementing Variant B, acknowledging that the improvement might not always reach practical significance.

b. Collect more data to narrow the confidence interval and reduce uncertainty.

c. Use additional business considerations, such as cost-benefit analysis or the ease of implementing Variant B, to make the final decision.