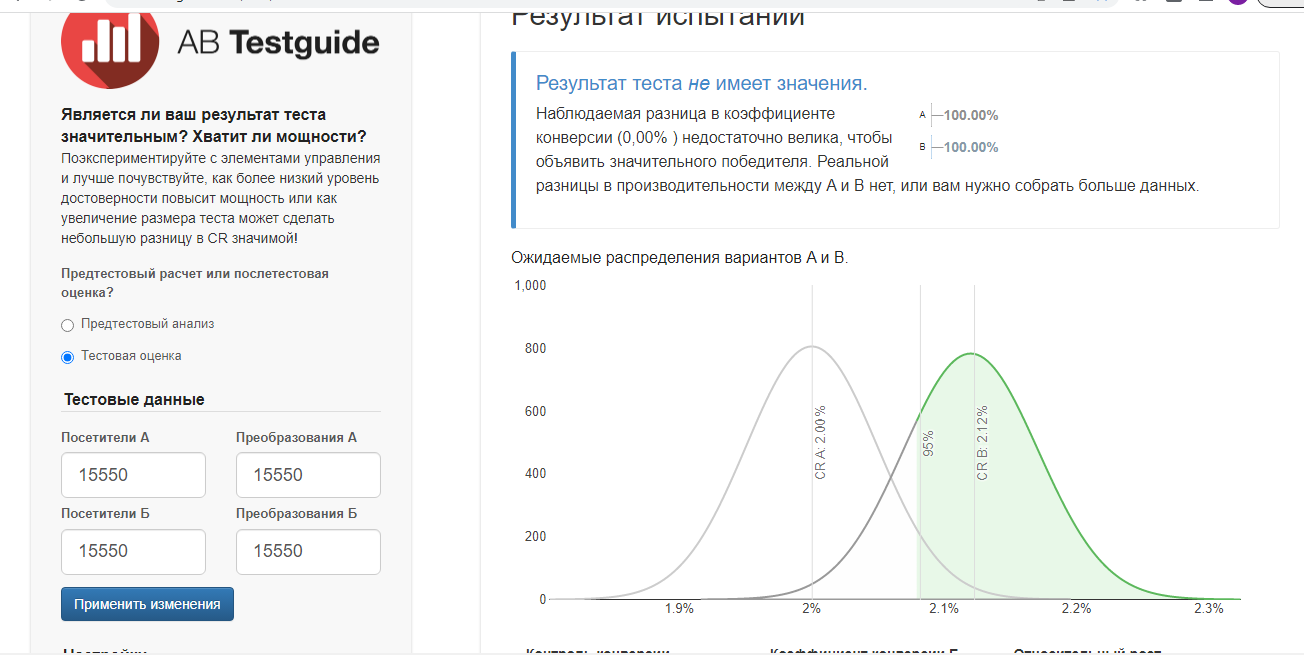
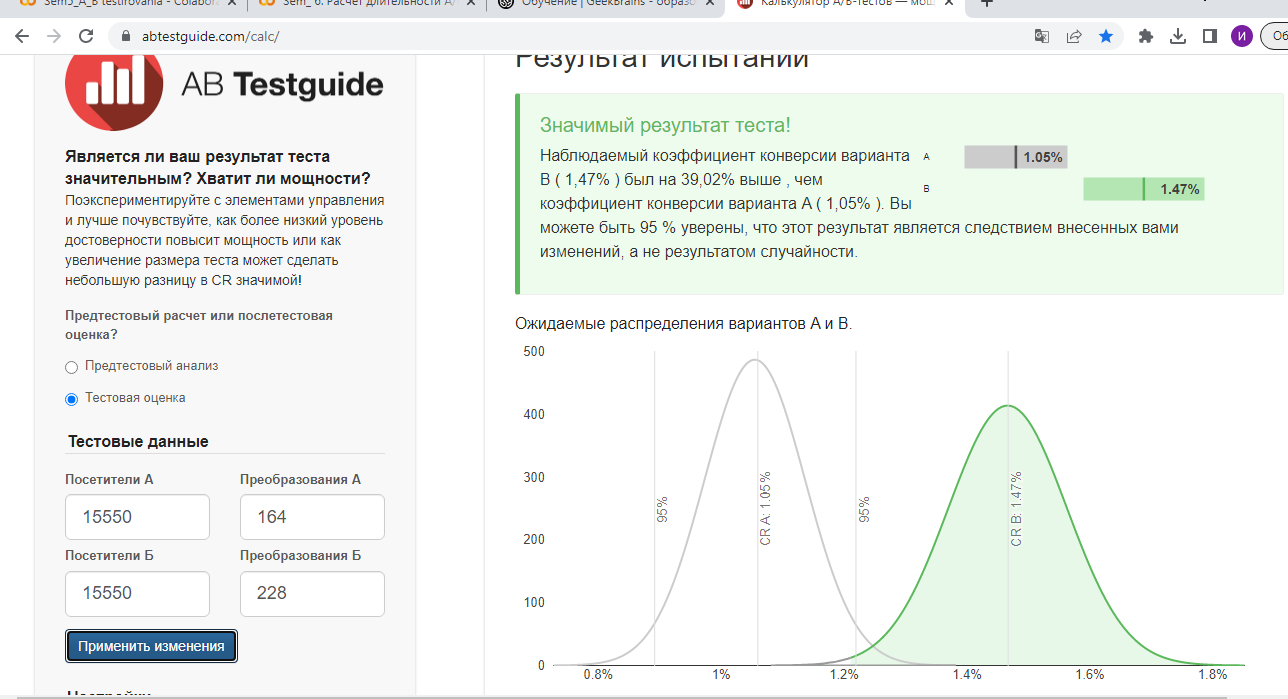
Урок 5. Применение математической статистики для проверки гипотез в реальной жизни для популярных метрик

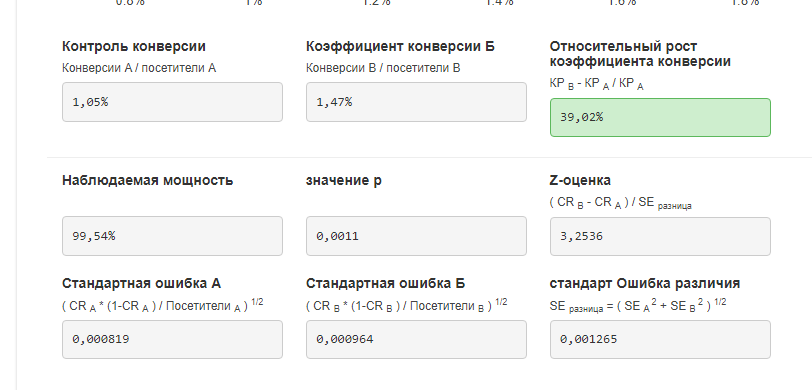
1)Вы провели эксперимент c упрощением формы заказа в магазине Утконос и получили результаты по метрике конверсий в покупку. Выберите метод оценки и оцените есть ли стат.значимые различия между конверсиями в двух группах при alpha = 5%. Дайте краткие рекомендации команде .Результаты: 1)Число юзеров в группах , которые заходили на сайт в период эксперимента: n1 = 15550 и n2 = 15550 . 2) Число юзеров в группах , которые совершили хотя бы одну покупку за период эксперимента: n1 = 164 и n2 = 228 3) Конверсии : conv1 = 1.05% conv2 = 1.47% .

1. Разницы в группах нет, статистически значимых различий нет



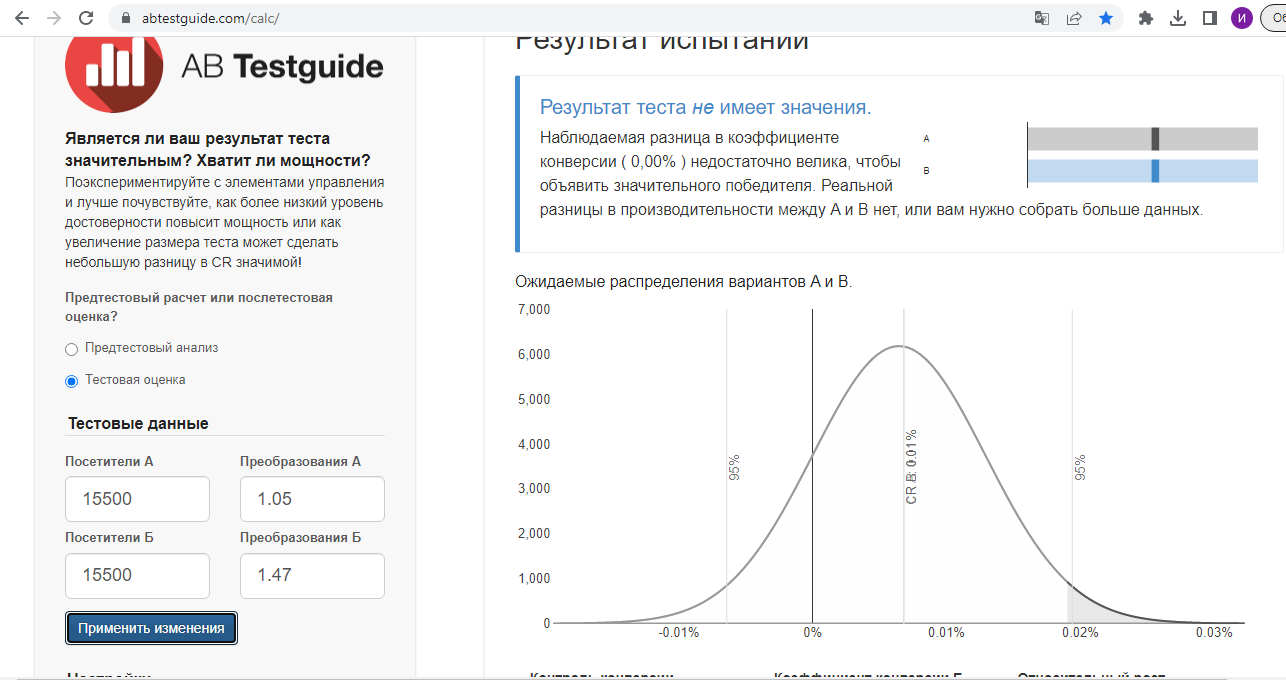
2)

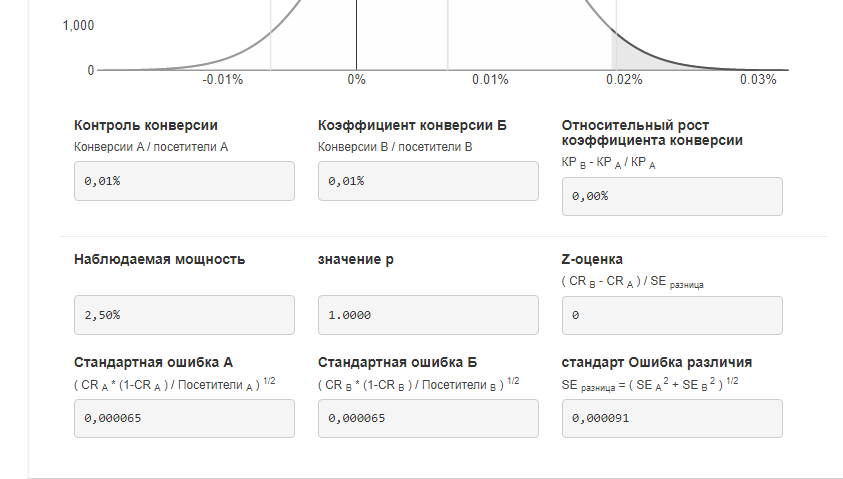




Данные изменения внесли изменения в эксперимент, они являются не случайными. Выявлены статистически значимые различия. Наблюдается высокая мощность теста 99%, p-value ниже a=5%, принимаем нулевую гипотезу.

3)





0 не входит в интервал, значит на основе собранных данных, обнаружены статистически значимые значения между группами, принимаем альтернативную гипотезу.

2) Сравниваем метрику конверсия в покупку. Размер выборки - 10000 элементов в каждой группе . Какой статистический критерий тут лучше всего подойдёт и почему ?

Метрика конверсия в покупку - Пропорционные или качественные метрика

наилучший метод чтобы проверить статистическую значимость изменения для пропорций это метод Хи‑Квадрат. Но перед тем, как его проводить, стоит проверить размер тестовой группы для желаемого минимального статистически значимого изменения. Это можно сделать, используя один из инструментов в интернете, например,  с помощью библиотеки statsmodels.stats через python, вот пример кода:

1 сек.

from statsmodels.stats.proportion import proportion\_effectsize

from statsmodels.stats.power import TTestIndPower

baseline\_cr = 0.2 # базовый уровень конверсии

min\_effect = 0.05 # минимальный значимый результат

effect\_size = proportion\_effectsize(baseline\_cr, baseline\_cr + min\_effect)

alpha = 0.05 # уровень значимости

power = 0.8  #уровень мощности

power\_analysis = TTestIndPower()

sample\_size = power\_analysis.solve\_power(effect\_size, power=power, alpha=alpha, alternative='two-sided')

print(f"Необходимый размер выборки: {sample\_size:.0f}")



Необходимый размер выборки: 1093

/usr/local/lib/python3.9/dist-packages/scipy/stats/\_continuous\_distns.py:6832: RuntimeWarning: invalid value encountered in \_nct\_sf

return np.clip(\_boost.\_nct\_sf(x, df, nc), 0, 1)

/usr/local/lib/python3.9/dist-packages/scipy/stats/\_continuous\_distns.py:6826: RuntimeWarning: invalid value encountered in \_nct\_cdf

return np.clip(\_boost.\_nct\_cdf(x, df, nc), 0, 1)

Выбираем Хи-квадрат p-value: 0.5637905872744118

Доверительный интервал изменения: (-0.40081583523469, 4.40041587523069)

Разницы нет.

import numpy as np

import scipy.stats as stats

# Загрузите данные в переменные

group\_A = [1, 10000]

group\_B = [2, 10001]

# Запустите тест

chi2, p, dof, ex = stats.chi2\_contingency([group\_A, group\_B], correction=False)

# Рассчитайте доверительный интервал для изменения

lift = (group\_B[0]/group\_B[1])/(group\_A[0]/group\_A[1])

std\_error = np.sqrt(1/group\_B[0] + 1/group\_B[1] + 1/group\_A[0] + 1/group\_A[1])

ci = stats.norm.interval(0.95, loc=lift, scale=std\_error)

# Выводим результаты

print("Хи-квадрат p-value: ", p)

print("Доверительный интервал изменения: ", ci)

# Проверяем есть ли изменение

if p < 0.05 and ci[0] > 1:

    print("Вариант лучше.")

else:

    print("Разницы нет.")