Основные понятия модуля

A/B-тестирование — это метод, который заключается в сравнении текущей версии продукта (версии *A*) с его изменённой версией (версией *B*) на основании данных, полученных до введения обновления в продукт и после него. Метод основан на проверке статистической значимости результатов эксперимента и позволяет заранее задать границу уверенности в результатах исследования (уровень надёжности).

Механизм A/B-тестирования

Контрольная версия — текущая версия продукта.
Тестовая версия — новая версия продукта.

Чтобы протестировать какую-либо гипотезу при помощи *A/B*-теста, аудиторию разделяют на две части:

- **Группа A** продолжает использовать (видеть) старую версию продукта.
- **Группа B** видит новую версию.
- В реальном времени собирают информация об обеих группах теста (*A* и *B*).
- Проводят замеры важных показателей.
- Проводят сравнение этих показателей.
- Принимают решение об эффективности влияния гипотезы на показатели продукта.

Принципы A/B-тестирования

- исключить влияние извне;
- использовать большой объём данных;
- применять правильные инструменты для анализа

Алгоритм A/B-тестирования

Конверсия — отношение числа посетителей сайта, выполнивших на нём какие-либо целевые действия, к общему числу посетителей сайта, выраженное в долях или процентах. Под целевым действием можно подразумевать покупку товара, лайк или репост поста в *Instagram*, просмотр фильма на Кинопоиске и многое другое.

Кумулятивные метрики

Кумулятивная метрика — это отображение целевой метрики, когда вы отслеживаете её поведение за каждый день — накопленным итогом по дням.

Принимать какие-либо решения стоит только после того, как метрика стабилизируется!

Кумулятивная метрика считается **стабилизированной**, когда на графике прекращаются резкие пики и спады показателя и линия постепенно выравнивается в прямую.

Пример вычисления кумулятивной метрики

```
daily_data_a.loc[:, 'cum_users_count'] =
daily_data_a['users_count'].cumsum()
```

Кумулятивную сумму можно записать в виде рекурсивной формулы:

$$S_t = S_{t-1} + x_t$$

- x_t — значение показателя в день t;
- S_t — значение суммы в день t.

Расчёт кумулятивной конверсии в процентах

```
daily_data['cum_conversion'] =
daily_data['cum_converted']/daily_data['cum_users_count'] * 100
```

Статистические тесты

Для проверки гипотезы равенства пропорций мы можем использовать:

- *Z*-тест для пропорций (**Z-критерий**). Самый популярный критерий для определения статистической значимости изменения конверсии.
- χ^2-тест (хи-квадрат, **χ^2-критерий**) для пропорций — это знакомый нам статистический тест для проверки независимости двух категориальных признаков. Однако его можно использовать и при проверке различия пропорций для сравнения наблюдаемых и ожидаемых частот в каждой группе, которые можно рассчитать на основе гипотетической модели независимости.
- Критерий Мак-Немара — ещё один знакомый нам статистический тест для проверки равенства пропорций в двух зависимых (связанных) между собой выборках. Как правило, при *A/B*-тестировании стараются использовать именно

независимые выборки, то есть показывать разным пользователям разные версии продукта. Однако если по каким-то причинам произвести независимое A/B тестирование невозможно и группы являются зависимыми, можно воспользоваться этим критерием.

Z-тест для пропорций

При использовании Z-теста выдвигают следующие нулевую (H_0) и альтернативную (H_1) гипотезы:

- Нулевая гипотеза: разница пропорций в группах A и B равна некоторому заранее заданному числу $value$.

$$H_0: p_a - p_b = value$$

- Альтернативная гипотеза: разница пропорций в группах A и B не равна некоторому заранее заданному числу $value$.

$$H_1: p_a - p_b \neq value$$

По умолчанию $value = 0$, соответственно, для случая двусторонних гипотез тестируется нулевая гипотеза о равенстве между истинными пропорциями p_a и p_b ($H_0: p_a = p_b$) против альтернативной об их неравенстве ($H_1: p_a \neq p_b$).
Альтернативная гипотеза также может быть и односторонней. Тогда гипотезы задаются в следующем виде:

- Правосторонняя альтернативная:

$$H_0: p_a - p_b \leq value$$
$$H_1: p_a - p_b > value$$

- Левосторонняя альтернативная:

$$H_0: p_a - p_b \geq value$$
$$H_1: p_a - p_b < value$$

При $value = 0$ мы получаем $H_0: p_a \leq p_b$ против $p_a > p_b$ для правосторонней альтернативной гипотезы и $H_0: p_a \geq p_b$ против $H_1: p_a < p_b$ — для левосторонней альтернативной гипотезы.

Пример кода для проверки гипотезы о равенстве конверсий в группах:

```
from statsmodels.stats.proportion import proportions_ztest
alpha = 0.05 # уровень значимости
# вычисляем значение p-value для Z-теста для пропорций
_, p_value = proportions_ztest(
    count=converted_piv['sum'], # число "успехов"
    nobs=converted_piv['count'], # общее число наблюдений
    alternative='two-sided',
)
# выводим результат на экран
print('p-value: ', round(p_value, 3))
# сравниваем полученное p-value с уровнем значимости
if (p_value < alpha):
    print("Отвергаем нулевую гипотезу в пользу альтернативной")
else:
    print("У нас нет оснований отвергнуть нулевую гипотезу")
```

χ^2-тест для пропорций

Этот тест применяется только для двусторонних гипотез:

- Нулевая гипотеза: разница пропорций в группах *A* и *B* равна некоторому заранее заданному числу $value$.

$$H_0: p_a - p_b = value$$

- Альтернативная гипотеза: разница пропорций в группах *A* и *B* отлична от числа $value$.

$$H_1: p_a - p_b \neq value$$

По умолчанию $value = 0$, соответственно, проверяется нулевая гипотеза равенства между истинными пропорциями p_a и p_b $(H_0: p_a = p_b)$ против альтернативной гипотезы об их неравенстве $(H_1: p_a \neq p_b)$.

Пример кода для проверки гипотезы о равенстве конверсий в группах:

```
from statsmodels.stats.proportion import proportions_chisquare

alpha = 0.05 # уровень значимости
# вычисляем значение p-value для Z-теста для пропорций
_, p_value, c = proportions_chisquare(
    count=converted_piv['sum'], # число «успехов»
    nobs=converted_piv['count'] # общее число наблюдений
)
# выводим результат на экран
print('p-value: ', round(p_value, 3))
# сравниваем полученное p-value с уровнем значимости
if (p_value < alpha):
    print("Отвергаем нулевую гипотезу в пользу альтернативной")
else:
    print("У нас нет оснований отвергнуть нулевую гипотезу")
```

Помимо конверсии нужно рассматривать и другие метрики, например **средний чек**. Важно, помнить о том, что любой статистический метод имеет свою область применения, которая зависит от задачи и распределения данных. Перед тем как проводить статистический тест, важно узнать распределение метрики, по которой вы будете сравнивать.

Параметрические тесты

Например, время, проведенное пользователем на сайте, часто бывает распределённым нормально. Тогда мы можем использовать двухвыборочный *T*-критерий для средних.

- Нулевая гипотеза (об отсутствии эффекта): среднее время, которое проводят на сайте пользователи из группы *A*, равно среднему времени, которое проводят на сайте пользователи из группы *B*.

$$H_0: \mu_a = \mu_b$$

- Альтернативная гипотеза (о наличии эффекта): среднее время, которое проводят на сайте пользователи из группы *A*, отличается от среднего времени, которое проводят на сайте пользователи из группы *B*.

$$H_1: \mu_a \neq \mu_b$$

Пример кода для проверки гипотезы о равенстве среднего времени на сайте в группах:

```
from scipy.stats import ttest_ind
alpha = 0.05 #уровень значимости
# вычисляем результат T-теста для выборок
results = ttest_ind(
    a=time_data['time(A)'],
    b=time_data['time(B)'],
    alternative='two-sided'
)
print('p-value:', round(results.pvalue, 2))

# сравниваем полученное p-value с уровнем значимости
if results.pvalue < alpha:
    print("Отвергаем нулевую гипотезу в пользу альтернативной")
else:
    print("У нас нет оснований отвергнуть нулевую гипотезу")
```

Непараметрические тесты

Денежные метрики, такие как средний чек, часто (но не всегда) напоминают логнормальное распределение: большинство наблюдений сосредоточены около нуля, и частота постепенно падает. Для исследования метрик необходимо воспользоваться непараметрическими тестами, например, критерием Манна — Уитни, *ANOVA*-тестом и др.

Например, для критерия Манна — Уитни гипотезы будут выглядеть следующим образом:

- Нулевая гипотеза (об отсутствии эффекта): распределение, лежащее в основе среднего чека в группе *A*, идентично распределению, лежащему в основе среднего чека в группе *B*.

$$H_0: F_a(u) = F_b(u)$$

- Альтернативная гипотеза (о наличии эффекта): распределение, лежащее в основе среднего чека в группе *A*, отлично от распределения, лежащего в основе среднего чека в группе *B*.

$$H_1: F_a(u) \neq F_b(u)$$

Пример кода для проверки гипотезы о равенстве распределений средних чеков в группах:

```
from scipy.stats import mannwhitneyu
alpha = 0.05 # уровень значимости

# вычисляем результат теста Манна – Уитини для выборок
results = mannwhitneyu(
    x=check_data['mean_check_a'],
    y=check_data['mean_check_b'],
    alternative='two-sided'
)
print('p-value:', round(results.pvalue, 2))

# сравниваем полученное p-value с уровнем значимости
if results.pvalue < alpha:
    print("Отвергаем нулевую гипотезу в пользу альтернативной")
else:
    print("У нас нет оснований отвергнуть нулевую гипотезу")
```

Доверительные интервалы

Интервальные оценки — это ещё один способ оценки параметров генеральной совокупности, при использовании которого ответ даётся не в виде одного числа, а в виде интервала.

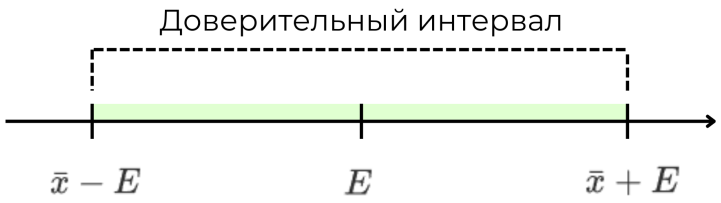
Доверительный интервал — интервал, который с заданной надёжностью покрывает значение неизвестного параметра.

Виды доверительных интервалов:

- двусторонние;
- левосторонние;
- правосторонние.

Любой **двусторонний** доверительный интервал обладает следующей структурой:

$$\text{Параметр} = \text{Выборочная оценка} \pm \text{Предел погрешности}$$



Доверительный интервал для **истинного среднего при известном стандартном отклонении**:

$$\mu = X_{mean} \pm z_{\alpha/2} \times \frac{\sigma}{\sqrt{n}}$$

- X_{mean} – выборочное среднее;
- σ – истинное стандартное отклонение;
- n – выборочное среднее
- $z_{крит} = z_{(1-\gamma)/2} = z_{\alpha/2}$ – значение, которое отсекает критическую область нормального распределения при надёжности, равной γ.

Под **уровнем надёжности** γ понимается вероятность того, что истинное значение параметра окажется в построенном интервале. А под **уровнем значимости** α — вероятность того, что построенный доверительный интервал «промахнётся» и не захватит истинное значение параметра.

Пример кода для построения доверительного интервала при известном стандартном отклонении:

```
from scipy.stats import norm

def z_mean_conf_interval(x_mean, sigma, n, gamma=0.95):
    alpha = 1 - gamma # уровень значимости
    z_crit = -norm.ppf(alpha/2) # z-критическое
    eps = z_crit * sigma/(n ** 0.5) #погрешность
    lower_bound = x_mean - eps # левая (нижняя) граница
    upper_bound = x_mean + eps # правая (верхняя) граница
    # возвращаем кортеж из границ интервала
    return lower_bound, upper_bound

# строим доверительный интервал для среднего при известном СКО
lower_bound, upper_bound = z_mean_conf_interval(x_mean, sigma, n)

# выводим результат
print('Доверительный интервал: {}'.format(lower_bound, upper_bound))
```

Доверительный интервал для **истинного среднего при неизвестном стандартном отклонении**:

$$\mu = X_{mean} \pm t_{крит} \times \frac{X_{std}}{\sqrt{n}}$$

- X_{std} — выборочное стандартное отклонение
- $t_{крит}(k) = t_{(1-\gamma)/2}(k) = t_{\alpha/2}(k)$ — значение, которое отсекает критическую область распределения Стьюдента при надёжности, равной γ.
- k — число степеней свободы.

Пример кода для построения доверительного интервала при неизвестном стандартном отклонении:

```
from scipy.stats import t
def t_mean_conf_interval(x_mean, x_std, n, gamma=0.95):
    alpha = 1 - gamma # уровень значимости
    t_crit = -t.ppf(alpha/2, k) # t-критическое
    eps = t_crit * x_std/(n ** 0.5) # погрешность
    lower_bound = x_mean - eps # левая (нижняя) граница
    upper_bound = x_mean + eps # правая (верхняя) граница
    # возвращаем кортеж из границ интервала
    return lower_bound, upper_bound

# строим доверительный интервал для среднего при неизвестном СКО
lower_bound, upper_bound = t_mean_conf_interval(x_mean, x_std, n)

# выводим результат
print('Доверительный интервал: {}'.format(lower_bound, upper_bound))
```

Доверительный интервал для **истинной пропорции**:

$$p = \mu = X_p \pm z_{крит} \times \sqrt{\frac{X_p(1-X_p)}{n}}$$

- X_p — выборочная пропорция

Пример кода для построения доверительного интервала для пропорции:

```
from scipy.stats import norm
def proportion_conf_interval(x_p, n, gamma=0.95):
    alpha = 1 - gamma # уровень значимости
    z_crit = -norm.ppf(alpha/2) # z-критическое
    eps = z_crit * (x_p * (1 - x_p) / n) ** 0.5 #погрешность
    lower_bound = x_p - eps # левая (нижняя) граница
    upper_bound = x_p + eps # правая (верхняя) граница
    # возвращаем кортеж из границ интервала
    return lower_bound, upper_bound


# строим доверительный интервал для пропорций
lower_bound, upper_bound  = proportion_conf_interval(
    x_p=x_p, # выборочная пропорция
    n=n, # размер выборки
    gamma=gamma # уровень надёжности
)

#выводим результат
print('Доверительный интервал: {}'.format(lower_bound, upper_bound))
```

Доверительный интервал **разницы пропорций:**

$$\Delta p = \Delta X_p \pm z_{\text{крит}} \times \sqrt{\frac{X_{p_a}(1-X_{p_a})}{n_a} + \frac{X_{p_b}(1-X_{p_b})}{n_b}}$$

Индексы *a* и *b* обозначают принадлежность параметра группе *A* или *B* соответственно.

- $\Delta p = p_b - p_a$ — истинная разница конверсий групп *B* и *A*;
- $\Delta X_p = X_{p_b} - X_{p_a}$ — выборочная разница конверсий групп *B* и *A*

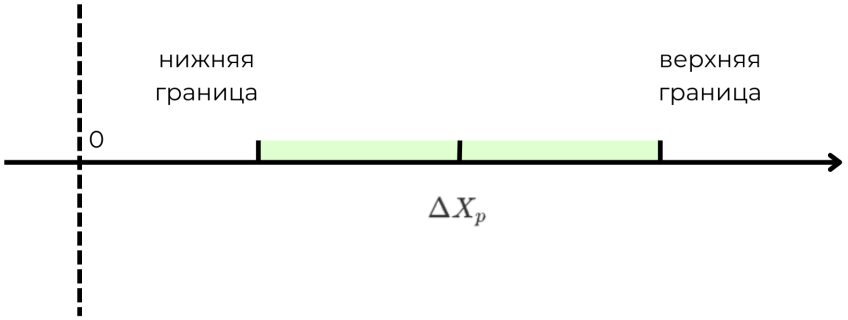
Пример кода для построения доверительного интервала для разности пропорций:

```
def diff_proportion_conf_interval(x_p, n, gamma=0.95):
    alpha = 1 - gamma # уровень значимости
    diff = x_p[1] - x_p[0] # выборочная разница конверсий групп B и A
    z_crit = -norm.ppf(alpha/2) # z-критическое
    eps = z_crit * (x_p[0] * (1 - x_p[0])/n[0] + x_p[1] * (1 - x_p[1])/n[1])
** 0.5 # погрешность
    lower_bound = diff - eps # левая (нижняя) граница
    upper_bound = diff + eps # правая (верхняя) граница
```

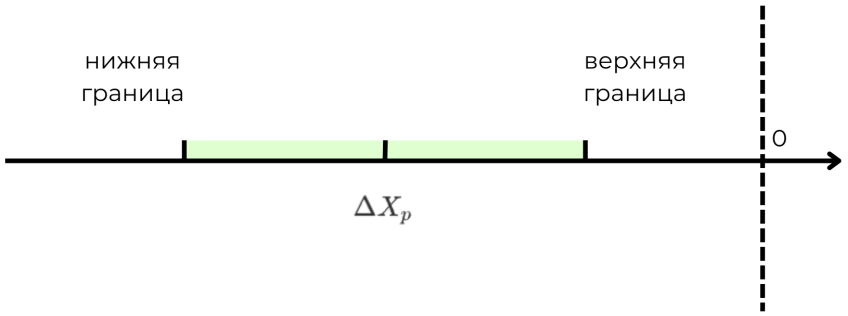
```
    # возвращаем кортеж из  границ интервала
    return lower_bound, upper_bound

# строим доверительный интервал для разности пропорций
lower_bound, upper_bound = diff_proportion_conf_interval(
    x_p=[xp_a, xp_b], # выборочные пропорции в группах
    n=[n_a, n_b] # размеры выборок
)

# выводим результат
print('Доверительный интервал для разности конверсий: {}'.format(lower_bound,
upper_bound))
```

Три случая доверительного интервала для разницы пропорций

Обе границы доверительного интервала являются **положительными** (больше 0). То есть истинная разница в пропорциях $\Delta p = p - p$ положительная. Пропорция *A* < пропорции *B*.



Обе границы доверительного интервала являются **отрицательными** (меньше 0). То есть истинная разница в пропорциях $\Delta p = p \quad - p$ отрицательна. Пропорция *A* > пропорции *B*.



Интервал охватывает точку 0. Левая граница доверительного интервала отрицательная, а правая — положительная. То есть истинная разница в пропорциях $\Delta p = p - p$ может быть как положительной, так и отрицательной. Тогда это будет значить, что пропорция *A* равна пропорции *B*.