Прикладная статистика и анализ данных ¶

Задание 6

Правила:

- Дедлайн **23 марта 16:30**. После дедлайна работы не принимаются кроме случаев наличия уважительной причины.
- Выполненную работу нужно отправить на почту `mipt.stats@yandex.ru` , указав тему письма `"[asda] Фамилия Имя - задание 6"` . Квадратные скобки обязательны. Если письмо дошло, придет ответ от автоответчика.
- Прислать нужно ноутбук и его pdf-версию (без архивов). Названия файлов должны быть такими: `6.N.ipynb` и `6.N.pdf` , где `N` - ваш номер из таблицы с оценками.
- Задачу 3 необходимо оформить в tex'е и прислать `pdf` или же прислать фотку в правильной ориентации рукописного решения, где **все четко видно**.
- Решения, размещенные на каких-либо интернет-ресурсах не принимаются. Кроме того, публикация решения в открытом доступе может быть приравнена к предоставлении возможности списать.
- Для выполнения задания используйте этот ноутбук в качестве основы, ничего не удаляя из него.
- Никакой код из данного задания при проверке запускаться не будет.
- В каждой задаче не забывайте делать **пояснения и выводы**.

Баллы за задание:

- Задача 1 - 5 баллов
- Задача 2 - 2 балла
- Задача 3 - 4 балла
- Задача 4 - 8 баллов

In [ ]:

```
import numpy as np
import pandas as pd
import scipy.stats as sps
import warnings
warnings.filterwarnings("ignore")

import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline
```

Задача 1

Рассмотрим таблицу с оценками за осенний семестр. Предполагая нормальность распределений проверьте следующие гипотезы:

- средний балл за первое и последнее практические задания не отличается;
- средний балл по решению домашних заданий в группах Димы и Ромы;
- средний балл за вторую контрольную в группах Димы и Ромы.

В каждом случае:

- для каждой выборки на одном графике постройте ядерные оценки плотности;
- постройте боксплоты;
- постройте доверительный интервал для разности средних.

Сделайте выводы.

Задача 2

Сравните вероятности получения конфет на экзамене в зависимости от семинарской группы в осеннем семестре. В каких случаях имеется статистически значимое отличие?

Сделайте выводы.

Задача 3

Пусть $X = (X_1, \ldots, X_n)$ и $Y = (Y_1, \ldots, Y_m)$ --- нормальные выборки с одинаковой неизвестной дисперсией. Докажите что статистика T-критерия о равенстве средних имеет распределение Стьюдента с $n - m - 2$ степенями свободы.

Указание. Воспользуйтесь теоремой об ортогональном разложении гауссовского вектора.

Задача 4

Для анализа будем использоваться датасет экспресии генов (https://ru.wikipedia.org/wiki/Экспрессия_генов) в нормальных тканях и в карциномах (https://ru.wikipedia.org/wiki/Карцинома), полученные с помощью нуклеотидных микрочипов. Данные опубликованы в работе `Notterman, et al, Cancer Research vol. 61: 2001` . Всего доступна информация о 18 опухолевых образцах и о соответсвующих им здоровых тканях.

Для лучшего понимания задачи можно почитать следующие статьи:

- https://fb.ru/article/256575/ekspressiya-genov---eto-chto-takoe-opredelenie-ponyatiya (https://fb.ru/article/256575/ekspressiya-genov---eto-chto-takoe-opredelenie-ponyatiya)
- https://ru.qwe.wiki/wiki/Gene_expression (https://ru.qwe.wiki/wiki/Gene_expression)

Шаг 1. Загрузка и подготовка датасета

Загрузим данные

In [ ]:

```
! wget http://genomics-pubs.princeton.edu/oncology/Data/CarcinomaNormalDataset
! unzip CarcinomaNormalDatasetCancerResearchText.zip
```

Загрузим данные в `pandas` и посмотрим на них:

In [ ]:

```
data = pd.read_table(
    "CarcinomaNormalDatasetCancerResearch.txt",
    skiprows=range(1,8), index_col=0, usecols=range(39)
)
data = data.drop(['Sample'], axis=1)

data.head()
```

Посмотрим также на хвост данных

In [ ]:

```
data.tail()
```

В конце прочитались две пустых строки. Удалим их и убедимся, что пропусков в данных нет

In [ ]:

```
data = data.iloc[:-2]
data.isna().sum()
```

Каждый ряд соответствует какому-то из интересующих нас генов, а колонка соответствует данным об экспресии каждого гена в опухолевых (`Tumor`) и контрольных (`Normal`) клеток.

Явно укажем вещественный тип данных

In [ ]:

```
data.iloc[:, 2:] = data.iloc[:, 2:].astype('float')
```

Проверим, уникальны ли все образцы в датасете, для этого сравним количество уникальных ID генов с количеством строк:

In [ ]:

```
len(np.unique(data.index)), len(data.index)
```

Некоторые эксперименты повторялись более одного раза. Оставим те, где средний уровень экспрессии выше.

In [ ]:

```
data['mean_expr'] = data.iloc[:, 1:-1].mean(axis=1)
data.sort_values(by=['mean_expr'], ascending=False)
data = data.groupby('Accession Number').first()
```

Сохраним описания генов и данные по их экспрессии отдельно. Значение средней экспрессии нам не нужны, поэтому избавимся от них

In [ ]:

```
expr_data, descr = data.drop(
    ['Description', 'mean_expr'], axis=1
), data.Description
```

In [ ]:

```
expr_data.head()
```

Для удобства работы транспонируем матрицу данных об экспрессии, и разметим для каждой строки, является ли образец опухолевым или нормой

In [ ]:

```
expr_data = expr_data.T
```

Осуществите разметку данных (`-1` -- опухолевые, `1` -- здоровые)

In [ ]:

```
<...>
```

In [ ]:

```
expr_data.head()
```

In [ ]:

```
expr_data.info()
```

Значения дифференциальной эскпрессии гена в образцах могут быть крайне различны. Для каждого гена можно характеризовать разброс экспрессии как среднее, максимум и минимум (отдельно в опухолевых и в контролях). Постройте гистограммы этих значений в опухолевых образцах и в контролях.

In [ ]:

```
<...>
```

1. Предварительная визуализация

Посмотрим на значения экспрессии случайного гена

In [ ]:

```
plt.figure(figsize=(12, 6))

plt.subplot(121)
sns.distplot(expr_data[expr_data.Label==1].iloc[:, 6], kde=True)
plt.title('Normal')

plt.subplot(122)
sns.distplot(expr_data[expr_data.Label==-1].iloc[:, 6], kde=True)
plt.title('Tumor');
```

Сравним плотности этих распределений с помощью `kdeplot` и разброс значений с помощью `boxplot` :

In [ ]:

```
1  <...>
```

На графиках видно, что для одного случайного гена профили экспрессии возможно различаются.
Но насколько это статистически достоверно и такова ли эта картина в целом?

Для продолжения анализа необходимо понять, как именно были получены данные. Для этого обратимся к статье:
Gene intensity information was converted to a mean intensity for each gene by proprietary software (Affymetrix), which includes routines for filtering and centering the data (in these experiments, to 50 intensity units). Expression of genes related to smooth muscle and connective tissue was consistently greater in the normal than the tumor samples, probably because of the greater heterogeneity of tissue type in the normal samples

Видим, что нормализация данных уже выполнена. Во многих случаях для визуализации удобно переходить к логарифмическому формату данных.

2. Анализ распределений

Предварительный зрительный анализ может сказать очень многое о том, как устроены наши данные. Тем не менее, для получения полной картины простого взгляда на данные недостаточно. Первое на что нужно обрать внимание -- это параметры распределения. Все распределения в первую очередь характеризуются медианой и средним. Для гена `D00137` вычислите медиану и средний уровень экспрессии в опухолевых и нормальных тканей.

In [ ]:

```
1  <...>
```

3. Проверка статистических гипотез

Для того, чтобы утверждать, что "ген `X` овер-экспрессирован в опухолевых образцах", недостаточно просто посмотреть на боксплоты, необходимо провести статистический анализ.

Для анализ одного гена проверяются гипотезы о сравнении профилей экспрессий между опухолевыми и контрольными образцами:
H_0: для гена `X` не наблюдается разницы средних экспрессий;
H_1: для гена `X` наблюдается разница средних экспрессий.

Какой критерий стоит выбрать для проверки в предположении нормальности распределений?
Воспользуйтесь этим критерием для проверки гипотезы для гена `Human class I alcohol dehydrogenase beta-1 subunit, allele 1 mRNA, complete cds` .

In [ ]:

```
1  <...>
```

Проинтерпретируйте результаты:

Type *Markdown* and LaTeX: α^2

Давайте проверим, что наши данные действительно разделяются на два разных распределения. Для этого проверим, как будет работать этот же метод, если случайным образом перемешать метки подгрупп. Реализуйте случайное сэмплирование с помощью функции `random.choice` из библиотеки `numpy` и посмотрите, как изменится p-value (и изменится ли).

In [ ]:

```
<...>
```

Сделайте вывод

<...>

И, наконец, самое интересное. Посчитайте, в каком проценте генов, для которых с уровнем значимости 0.05 наблюдается овер-экспрессия в опухолевых тканях. При этом важно выполнить поправку на множественное тестирование.

In [ ]:

```
<...>
```

Постройте гистограмму полученных значений p-value и скорректированных.

In [ ]:

```
<...>
```

Напечатайте число генов, для которых можно отвергнуть нулевую гипотезу, а также их долю среди всех генов

In [ ]:

```
<...>
```

4. Немного поближе взглянем на результаты

Посмотрим, какие же гены оверэкспрессированы. В коде ниже `p_vals_adjusted` -- скорректированные значения p-value.

In [ ]:

```
for name, function in zip(
    expr_data.columns[np.where([p_vals_adjusted < 0.05])[1]],
    descr[np.where([p_vals_adjusted < 0.05])[1]]
):
    print(name + ": " + function)
```