Университет ИТМО

Практическая работа №2

по дисциплине «Визуализация и моделирование»

Автор: Голуб А. Л.
Поток: ВИМ 1.2
Группа: К3243
Факультет: ИКТ
Преподаватель: Чернышева А.В.

Санкт-Петербург, 2021 г.

1 Описание данных

В ходе практической работы был выбран датасет Goodreads-books с информацией о книгах и их рейтингах на популярном ресурсе Goodreads. Информация о столбцах датасета представлена в таблице.

название столбца в датасете	**краткое описание данных**	**тип данных**	**шкала данных**
bookID	уникальный номер книги в датасете	натуральное число	интервальная
title	название книги, использованное при публикации	строка	номинальная
authors	список авторов книги	строка с перечислением имен авторов через /	номинальная
average_rating	средний рейтинг книги на сайте	число от 0.0 до 5.0	интервальная
isbn	ISBN-идентификатор книги	натуральное число (11 знаков)	интервальная
isbn13	ISBN-идентификатор книги (13 знаков)	натуральное число (13 знаков)	интервальная
language_code	язык оригинала книги	строка	номинальная
num_pages	число страниц в книге	натуральное число	относительная
ratings_count	сколько раз книге был поставлен рейтинг	натуральное число	относительная
text_reviews_count	число письменных отзывов на книгу	натуральное число	относительная
publication_date	дата первой публикации книги	дата в формате MM/DD/YYYY	интервальная
publisher	издательство, опубликовавшее книгу	строка	номинальная

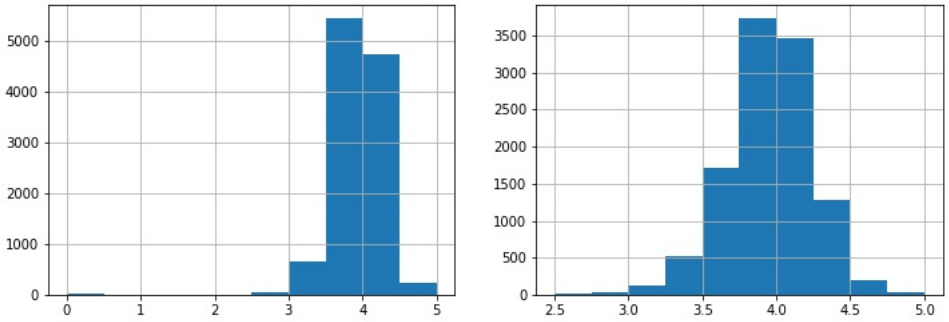
2 Описательная статистика

Построим диаграммы распределения данных в столбцах датасета, чтобы выявить основные закономерности в них.

1. Как статистически распределены **рейтинги книг** на сайте?
   Нарисуем гистограмму распределения и приблизим наиболее информативный участок.

```
fig, axes = plt.subplots(nrows=1, ncols=2)
df['average_rating'].hist(ax=axes[0], figsize=(12, 4));
df['average_rating'].hist(ax=axes[1],
  range=[2.5, 5], figsize=(12, 4));

```

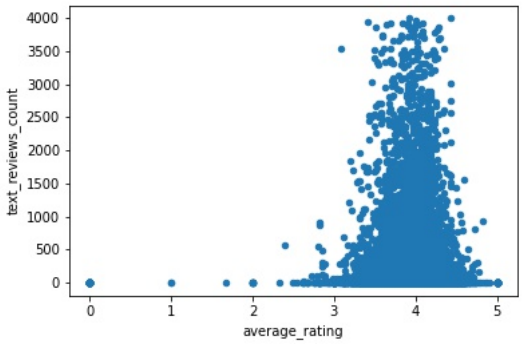


   Чаще всего книги получают оценки от **3.5** до **4.5**. Книги с рейтингом 0, скорее всего, пока не получили ни одной оценки. С помощью запроса к датасету можно посчитать, что таких книг 25.

2. Как **число текстовых отзывов** на книгу соотносится с ее **рейтингом**? Правда ли, что пользователи пишут отзывы на непонравившиеся книги чаще, чем на понравившиеся?
   Отфильтруем выбросы и построим график scatter plot.

```
df.query('text_reviews_count < 4000').plot. \
  scatter(x='average_rating', y='text_reviews_count');

```



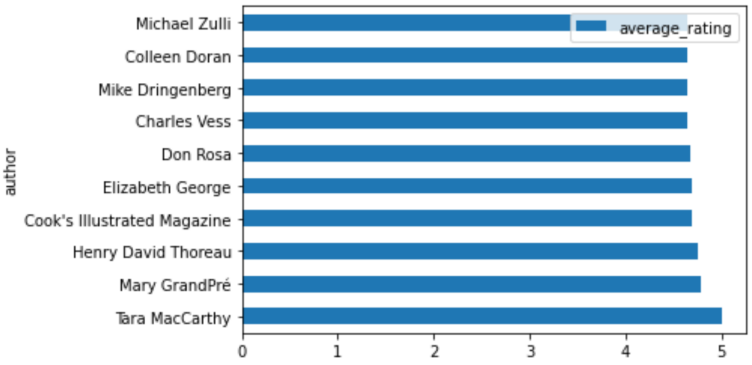
Судя по графику, предположение неверно. Число текстовых отзывов и количество поставленных рейтингов (см. гистограмму выше) примерно одинаково распределены относительно величины рейтинга.

3. У каких **авторов самый высокий средний рейтинг**?
   Создадим датафрейм со столбцами «автор», «средний рейтинг книг автора» и «общее число книг автора в датасете». Возьмем авторов, написавших не менее трех книг, чтобы сделать визуализацию более репрезентативной, и отсортируем их по рейтингу по убыванию. Топ-десять получившегося списка изобразим на диаграмме.

```
authors_ratings = dict()
books_count = dict()
for _, row in df[['authors', 'average_rating']].iterrows():
  authors = row['authors'].split('/')
  for a in authors:
    authors_ratings[len(authors_ratings)] = [a, row['average_rating']]
    if a in books_count:
      books_count[a] += 1
    else:
      books_count[a] = 1

authors_ratings = pd.DataFrame.from_dict(data=authors_ratings,
  orient='index', columns=['author', 'average_rating'])
authors_ratings['books_number'] = pd.Series([books_count[a] for a in
 authors_ratings['author']])
authors_ratings[(authors_ratings['books_number'] >= 3) \
  & (authors_ratings['author'] != 'NOT A BOOK')] \
  .groupby('author')['average_rating'].apply(pd.Series.mode) \
  .sort_values(ascending=False).reset_index() \
  .iloc[:10].plot.barh(x='author', y='average_rating');

```



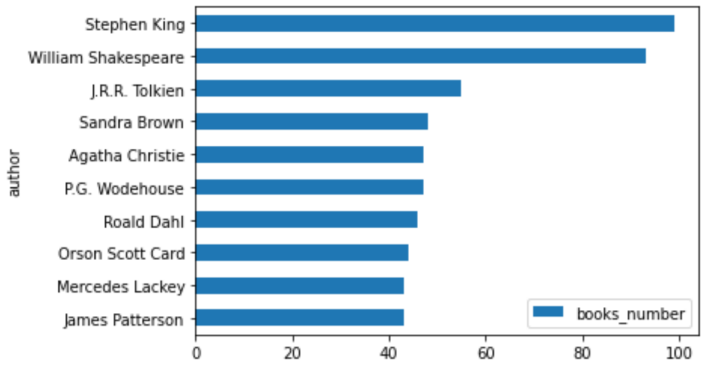
Как видно на диаграмме, средний рейтинг книг наиболее высоко оцененных авторов лежит в диапазоне от 4.5 до 5.

4. Какие **авторы** написали **больше всех книг**?
   Воспользуемся только что созданным датафреймом и найдем топ-десять авторов по

числу написанных книг. Результат также изобразим на диаграмме.

```
authors_ratings = authors_ratings.drop_duplicates(subset=['author'])
fig = authors_ratings.sort_values(ascending=False, by='books_number')
  .iloc[:10].plot.barh(y='books_number', x='author');
fig.invert_yaxis()

```



Наиболее плодовитыми писателями оказались Стивен Кинг, Уильям Шекспир и Дж. Р. Р. Толкин.

5. Книг на каком **языке** в датасете больше всего?

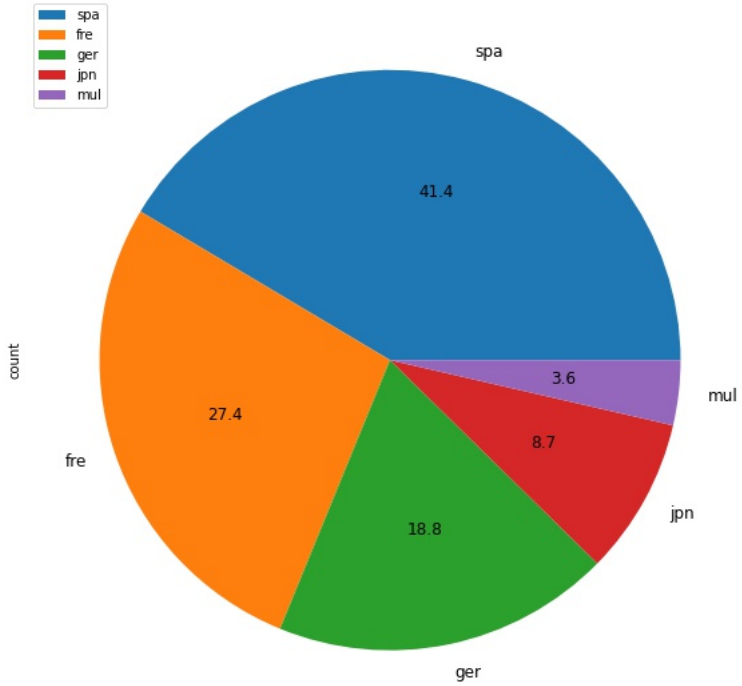
   Сгруппируем датасет по языкам, посчитаем, сколько книг написано на каждом из них, и сохраним результат в отдельный датафрейм. Подавляющее большинство книг в датасете - 1629 - написано на **английском** языке; исключим его из рассмотрения, чтобы получить более репрезентативное изображение. Визуализируем процентное соотношение книг среди пяти самых частых языков в виде круговой диаграммы.

```
languages_count = df.groupby('language_code')['language_code']. \
  count().reset_index(name='count')
eng_sum = languages_count[languages_count['language_code'].str. \
 contains('en-')]['count'].sum()
languages_count[languages_count['language_code'] == 'eng']['count']
  += eng_sum
languages_count.drop(languages_count.loc[languages_count[
  'language_code'].str.contains('en-')].index, inplace=True)

languages_count = languages_count.sort_values(ascending=False,
  by='count').iloc[1:6]
languages_count.plot.pie(y='count', labels=languages_count[
  'language_code'], autopct="%.1f", fontsize=12, figsize=(10, 10));

```

После английского языка больше всего книг в датасете написано на **испанском**, **французском** и **немецком**.

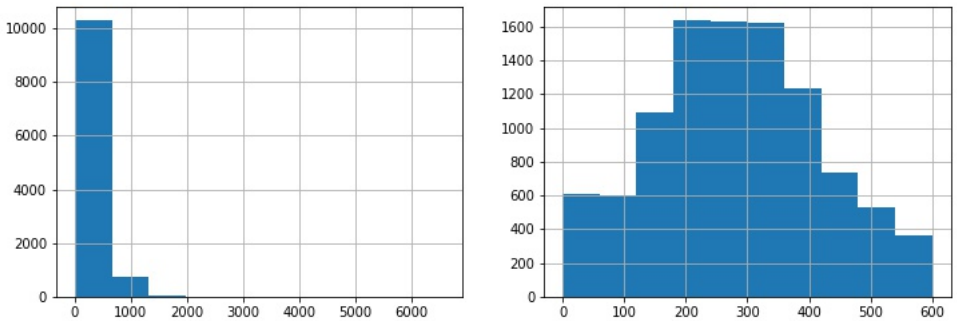


6. Распределение **числа страниц**

   Построим распределение с помощью pandas-метода hist() и приблизим его наиболее информативный фрагмент.

```
fig, axes = plt.subplots(nrows=1, ncols=2)
df['num_pages'].hist(ax=axes[0], figsize=(12, 4));
df['num_pages'].hist(ax=axes[1], range=[0, 600], figsize=(12, 4));

```



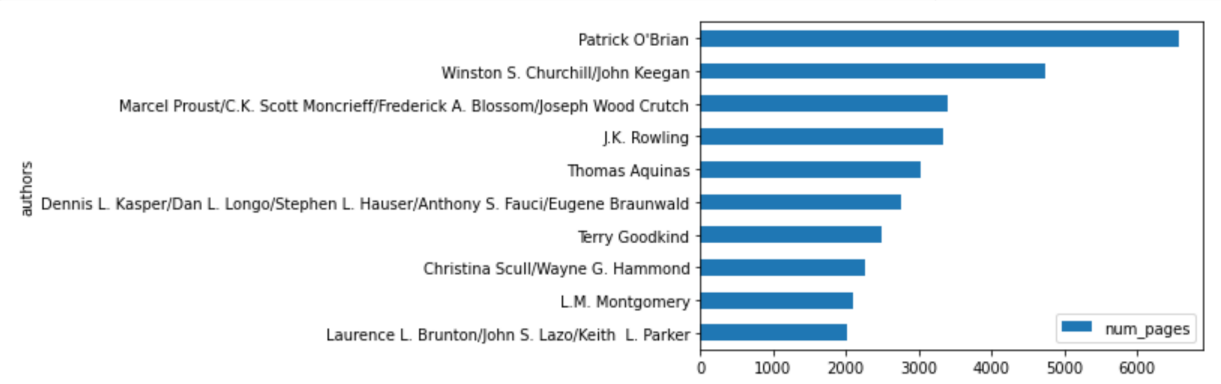
   В датасете представлены книги длиной от 0 до 6576 страниц, тогда как медианное значен числа страниц - 299. Как видно на диаграмме, большая часть книг в датасете длиной от **200** до **400** страниц.

7. Какие **авторы** написали **самые длинные книги**?

   Отсортируем строки по убыванию числа страниц. Отобразим топ-десять авторов на диаграмме, исключив из рассмотрения анонимных авторов и J.K. Rowling/Mary GrandPré. (J.K. Rowling уже входит в этот список, к тому же, Mary GrandPré иллюстрировала произведения Роулинг, но не была их автором).

```
fig = df[['authors', 'num_pages']].sort_values(
  ascending=False, by='num_pages') \
  .query('authors not in ["Anonymous",
  "J.K. Rowling/Mary GrandPr "]').iloc[:10] \
  .plot.barh(x='authors', y='num_pages');
fig.invert_yaxis()

```

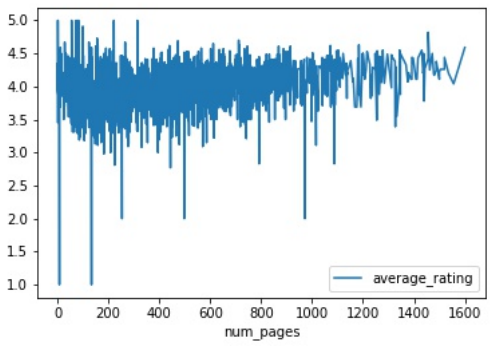


8. Как **рейтинг** книги зависит от **числа страниц**?

   Отфильтровав записи о книгах, построим график зависимости среднего рейтинга книги от числа страниц в ней.

```
pages_rating = df[['num_pages', 'average_rating']]
  .query('0 < num_pages <= 1600 & average_rating != 0')
pages_rating.groupby('num_pages')['average_rating']. \
  apply(pd.Series.mode).reset_index(). \
  plot(x='num_pages', y='average_rating');

```



9. Как количество **публикуемых** книг менялось **со временем**?

   Создадим отдельный датафрейм, где посчитаем, сколько книг было издано в какой год.

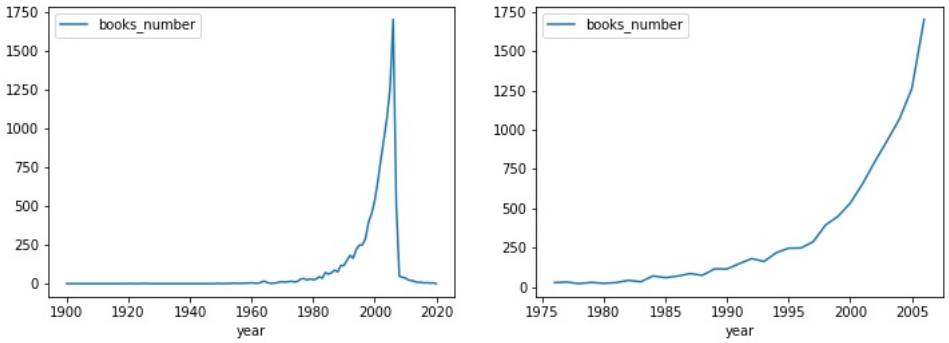
```
published_over_time = df[['bookID', 'publication_date']]
years = []
for v in published_over_time['publication_date']:
  years.append(int(v.split('/')[2]))
published_over_time['year'] = pd.Series(years)
published_over_time = published_over_time.groupby('year')['bookID'].
  count().reset_index(name='books_number')

```

   Построим график зависимости числа книг от года их издания. Очевидно, в датафрейме в основном представлены книги, изданные в промежуток с **1975** по **2007** год. Построим дополнительный график, чтобы рассмотреть этот участок кривой.

```
fig, axes = plt.subplots(nrows=1, ncols=2)
published_over_time.plot(ax=axes[0], x='year',
  y='books_number', figsize=(12, 4));
published_over_time.query('1975 < year < 2007')
  .plot(ax=axes[1], x='year', y='books_number', figsize=(12, 4));

```

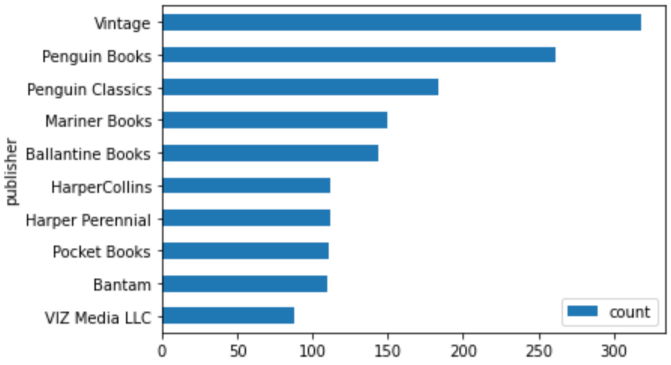


10. Какое **издательство** печатает **больше всех книг**?

    Создадим датафрейм со столбцами «издательство» и «число изданных им книг». Отсортируем список по числу книг и отобразим топ-десять на диаграмме.

```
publishers =  df.groupby('publisher')['publisher'].count() \
  .reset_index(name='count')
fig = publishers.sort_values(ascending=False, by='count') \
  .iloc[:10].plot.barh(x='publisher', y='count');
fig.invert_yaxis()

```

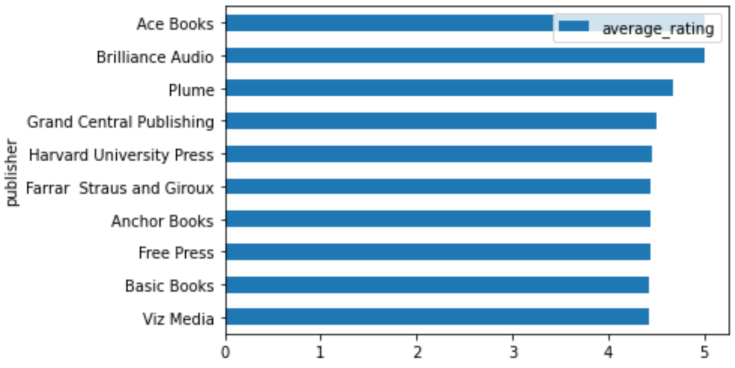


Из всех книг в датасете более 300 были изданы издательством «Vintage». Более 250 книг вышли в издательстве «Penguin Books» и чуть менее 200 - в издательстве «Penguin Classics». Около 150 книг напечатали «Ballantine Books» и «Mariner Books» каждое.

11. У какого **издательства самый высокий средний рейтинг**?
Дополним датафрейм publishers столбцом со средним рейтингом книг, изданных издательством. Среди издательств, напечатавших хотя бы 20 книг, выявим топ-десять по рейтингу и изобразим результат на диаграмме.

```
publishers['average_rating'] = df.groupby('publisher')[
  'average_rating'].apply(pd.Series.mode). \
reset_index()['average_rating']
fig = publishers.query('count >= 20').sort_values(
  ascending=False, by='average_rating').iloc[:10]
  .plot.barh(x='publisher', y='average_rating');
fig.invert_yaxis()

```



Самый высокий средний рейтинг у книг издательств «Ace Books», «Brilliance Audio» и «Plume».