|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА − Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИИТ)**

**Кафедра прикладной математики (ПМ)**

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №11**

по дисциплине «Языки программирования для статистической обработки данных»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент группы ИНБО-01-17 | *ИМБО-11-23, Журавлев Ф. А.* | (подпись) | |  |
| Преподаватель | *Трушин СМ* | (подпись) | |  |
|  |  | |  | |

Москва 2025 г.

## 1) ЦЕЛЬ И ЗАДАЧИ

**Цель практической работы:**

Освоить методы создания интерактивных графиков в Python, R.

**Задачи практической работы:**

1. Создать интерактивные графики:

• Python: использование библиотеки plotly для построения графиков с возможностью взаимодействия (зум, фильтры).

• R: построение интерактивных панелей с использованием пакета shiny.

2. Работа с большими данными:

• Использовать фильтры, агрегацию и другие инструменты для упрощения анализа.

3. Сравнить подходы работы с большими данными в Python (pandas, dask), R (data.table).

4. Оптимизировать обработку данных:

• Анализ производительности каж дого инструмента при работе с большим объёмом данных.

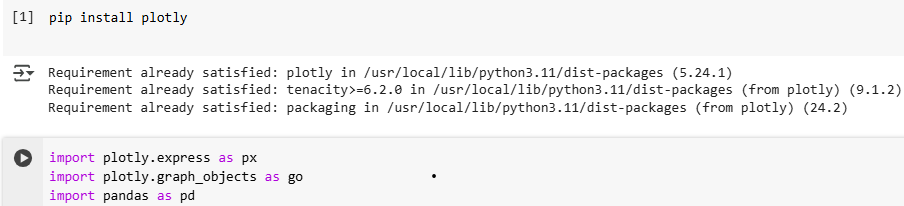
## 2) РЕЗУЛЬТАТЫ ПРАКТИКИ

**Шаг 1) Создание интерактивных графиков и работа с большими данными.**

### 1.1) Создание интерактивных графиков и работа с большими данными в Python.

Для начала работы необходимо установить все необходимые библиотеки, так как их достаточно много, то ниже написан код, который показывает все библиотеки, нужные для практической работы.

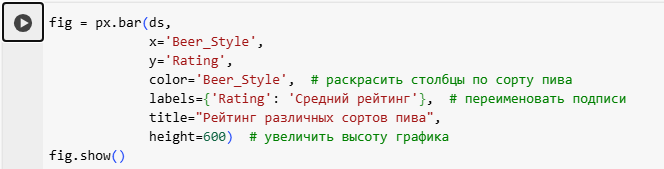
*Рисунок 1.1 — Необходимые библиотеки.*



Далее убедитесь, что у нас есть столбцы, из типа «value» и «category». Для дальнейшего построения интерактивных графиков.

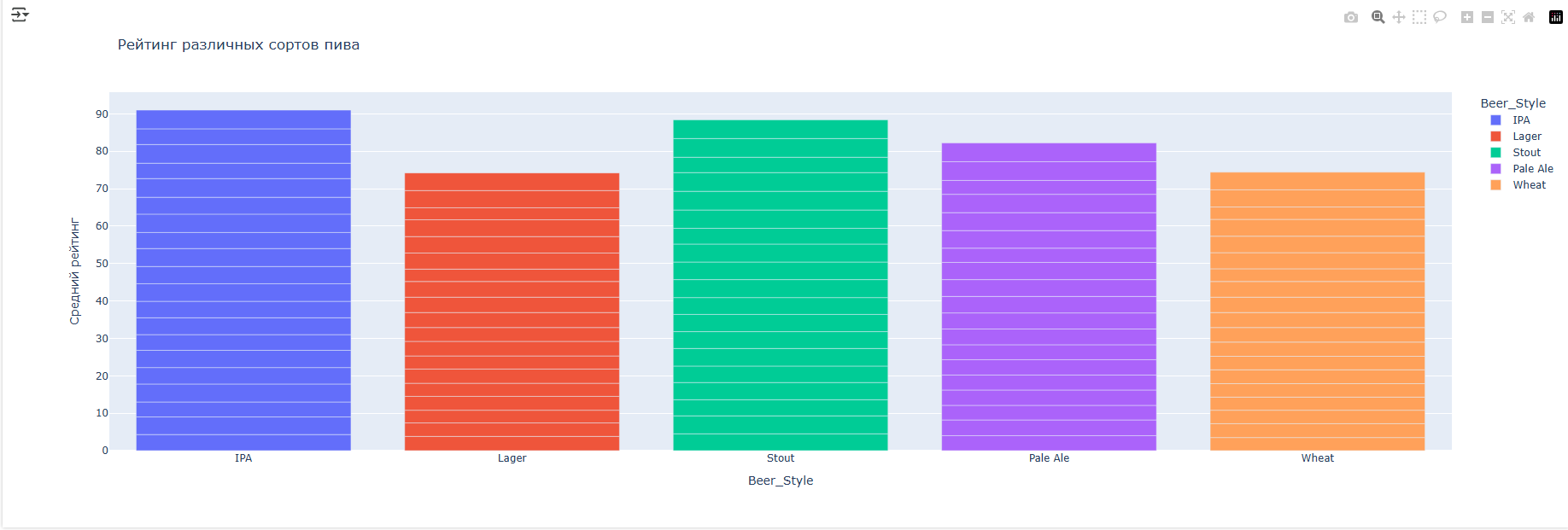
После напишем код, благодаря которому сможем увидеть пример подобного графика:

*Рисунок 1.2 – Код для построения графика.*



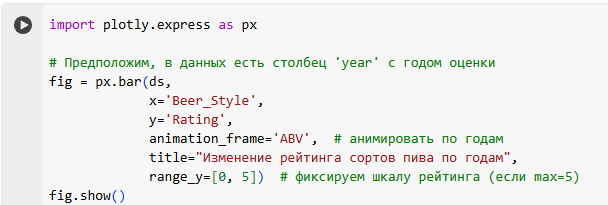
Построим какую красоту нам покажет этот код:

*Рисунок 1.3 — Интеративный график.*



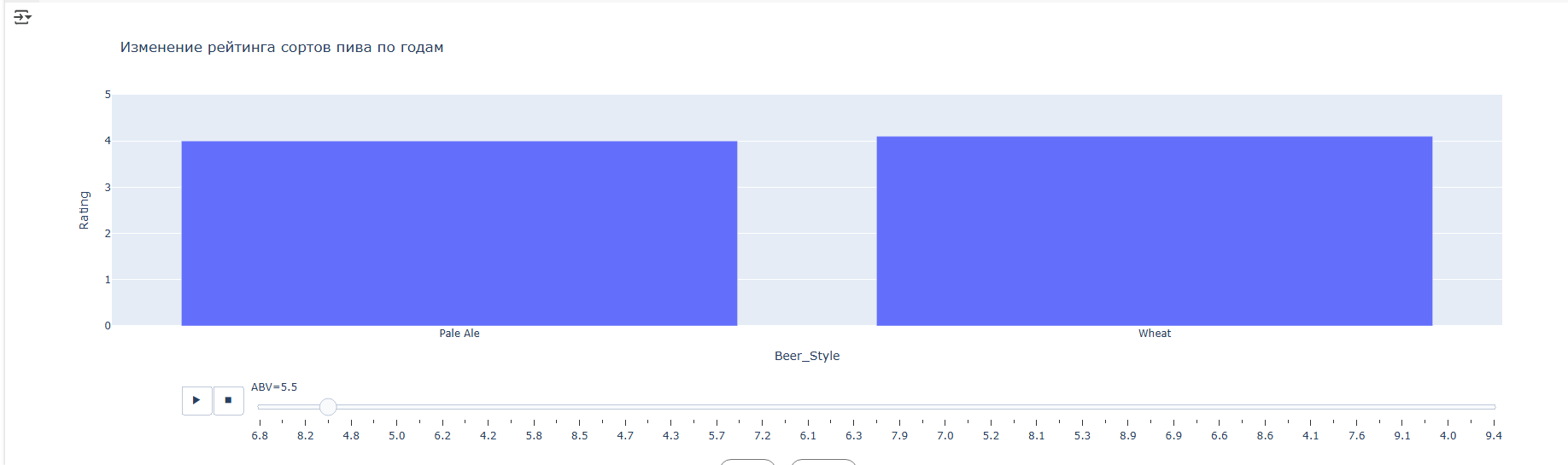
Далее можно написать код, благодаря которому мы сможем **анимировать** наши графики. Ниже представлен максимально простой способ:

*Рисунок 1.4 - Код для анимации.*

**

Далее посмотрим, какой результат выводит этот код:

*Рисунок 1.5 - Анимированный график.*

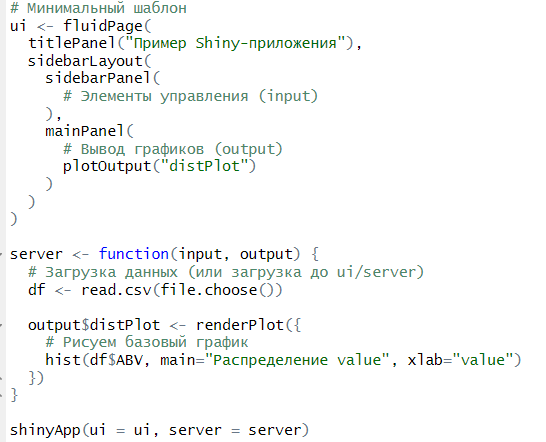


**Шаг 2) Создание интерактивных графиков и работа с большими данными.**

**2.1) Создание интерактивных графиков и работа с большими данными в RStudio.**

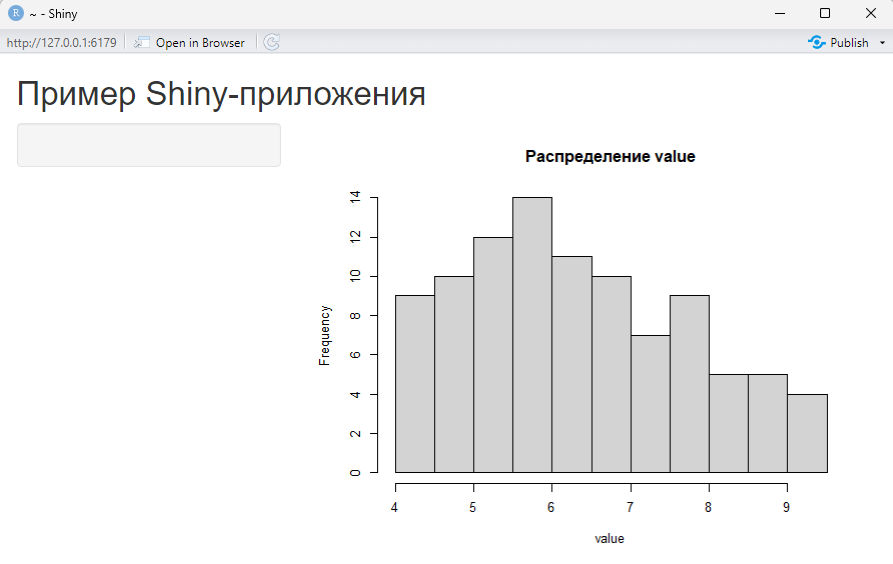
Далее воспользуемся специальным сервером и библиотекой Shiny, для визуализации интерактивых графиков. Ниже представлен максимально простой способ:

*Рисунок 2.1 — код в R.*

****

Посмотрим, что он нам выводит:

*Рисунок 2.2 — Пример Shiny-приложения.*

****

**Д**алее добавим интерактивных элементов:

*Рисунок 2.3 — График #1.*



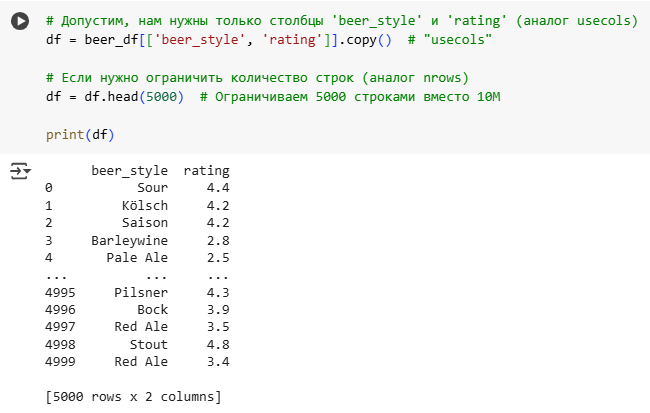
Дело в том, что мы не ограничены только стилем “Stout”, мы можем выбирать все категории пива, которые присутствуют в нашем наборе данных в этом приложении. Например вот пример для “Lager”

*Рисунок 2.4 — График #2.*

**3.1) Работа с большими данными в Python.**

Далее немного поработаем с большими данными, загрузим дата сет и отфильтруем только нужные столбцы, которые гипотетические будут нужны нам для работы, а также воспользуемся библиотекой dask, которая помогает работать ну прямо с огромными данными, которые разделенные на специальные чанки, ниже представлены различные коды:

*Рисунок 3.1 — Выборка нужного из дата сета.*

****

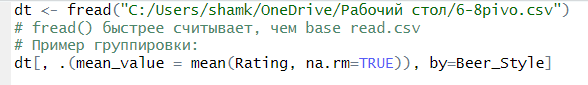
*Рисунок 3.2 — Выборка с помощью dask.*

****

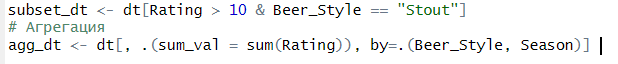
**3.2) Работа с большими данными в RStudio.**

Для начала установим библиотеку data.table, для того, чтобы быстрее считывать большие данные. Далее с помощью нее отфильтруем нужные нам значения:

*Рисунок 3.3 — Фильтрация.*

****

А затем произведем агрегацию, чисто протестировав код для примера:

*Рисунок 3.4 — Агрегация.*

В нашем дата сете не оказалось таких значений, но это и не страшно, главное что сам код работает верно и подойдет для различных практических заданий.

*Рисунок 3.5 — Результат агрегации.*

**3 ВЫВОДЫ**

В результате 11 практической работы мы научились строить интегративные графики в python и r, научились строить shiny-приложение, хоть и самое простое, а также чуть-чуть поработали с большими данными, для наглядности реализовав функции data.table из r, и dask из python.