Компьютерный практикум по статистическому анализу данных

Лабораторная работа № 7. Введение в работу с данными

Сунгурова Мариян Мухсиновна

Содержание

Список иллюстраций

1 Введение

Цель работы

Основной целью работы является освоение специализированных пакетов Julia для обработки данных.

Задачи

1. Используя Jupyter Lab, повторите примеры.
2. Выполните задания для самостоятельной работы.

2 Теоретическое введение

Julia — высокоуровневый свободный язык программирования с динамической типизацией, созданный для математических вычислений[1]. Эффективен также и для написания программ общего назначения. Синтаксис языка схож с синтаксисом других математических языков, однако имеет некоторые существенные отличия.

Для выполнения заданий была использована официальная документация Julia[2].

3 Выполнение лабораторной работы

Выполним примеры из лабораторной работы для изучения циклов и функций(рис. fig. 3.1 - fig. 3.15)

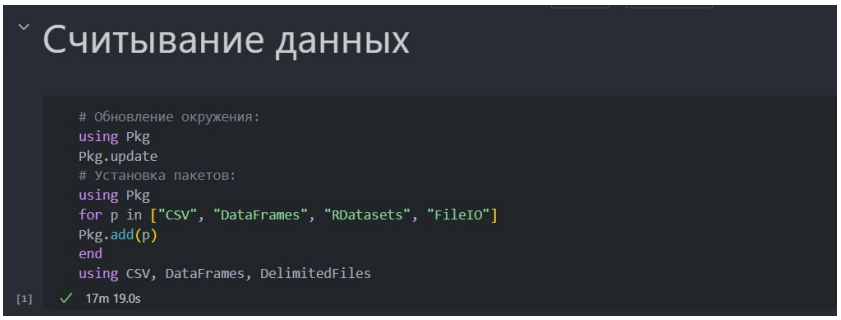

Рис. 3.1: Примеры Считывание данных

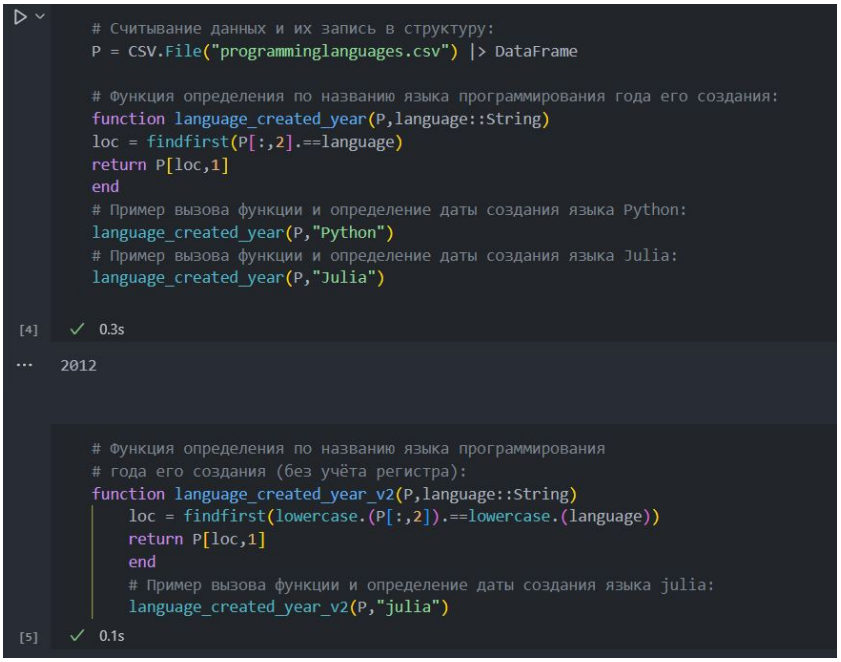

Рис. 3.2: Примеры Считывание данных

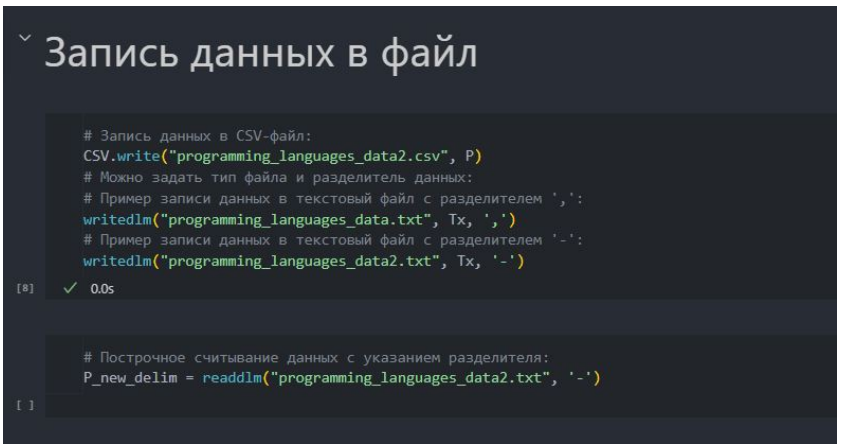

Рис. 3.3: Примеры Запись данных в файл

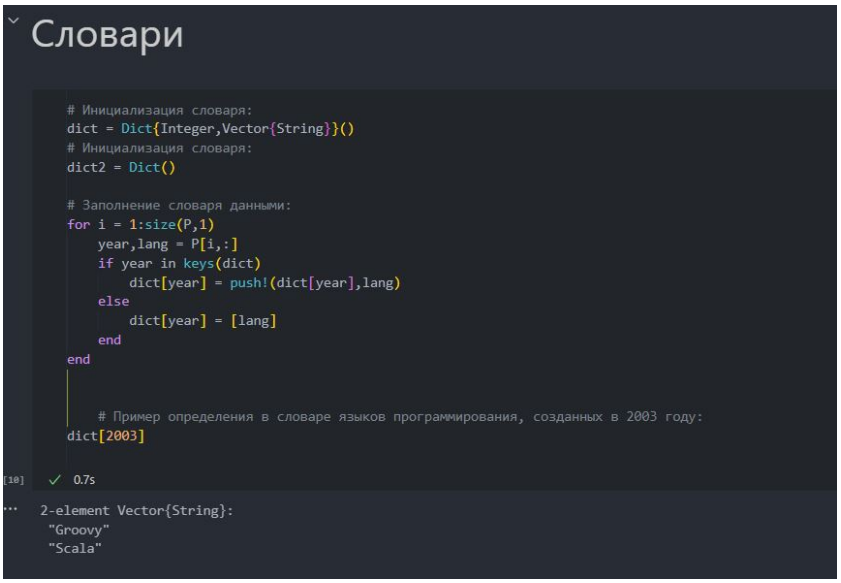

Рис. 3.4: Примеры Словари

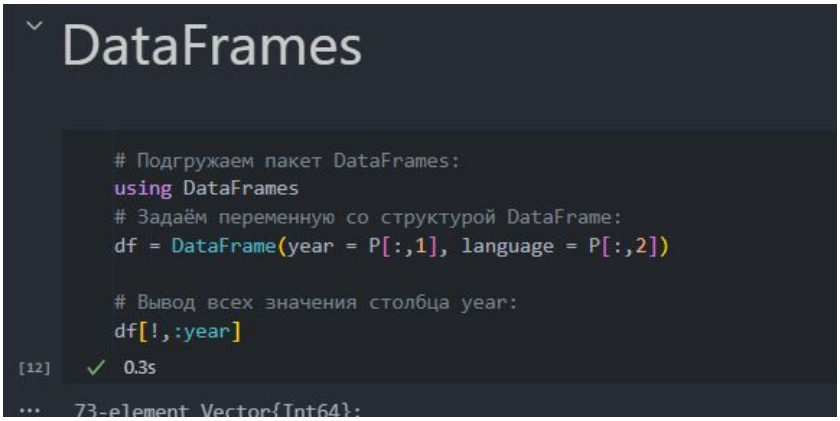

Рис. 3.5: Примеры DataFrames

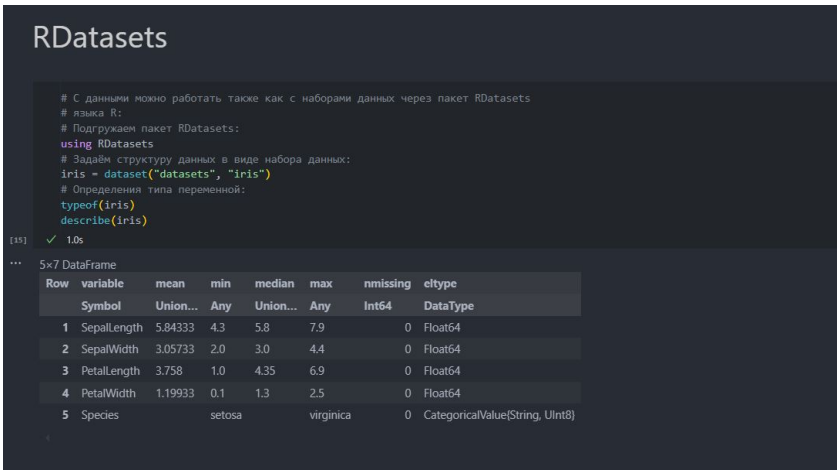

Рис. 3.6: Примеры RDatasets

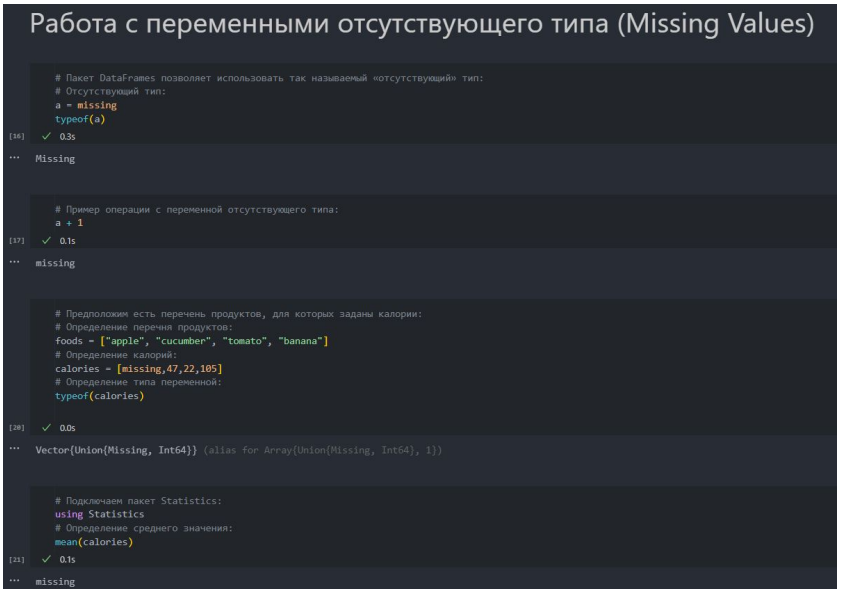

Рис. 3.7: Примеры Missing type

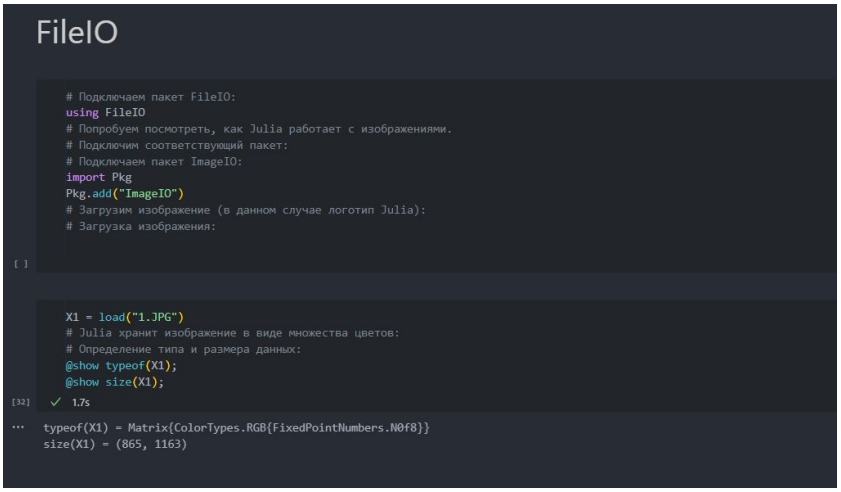

Рис. 3.8: Примеры Fileio

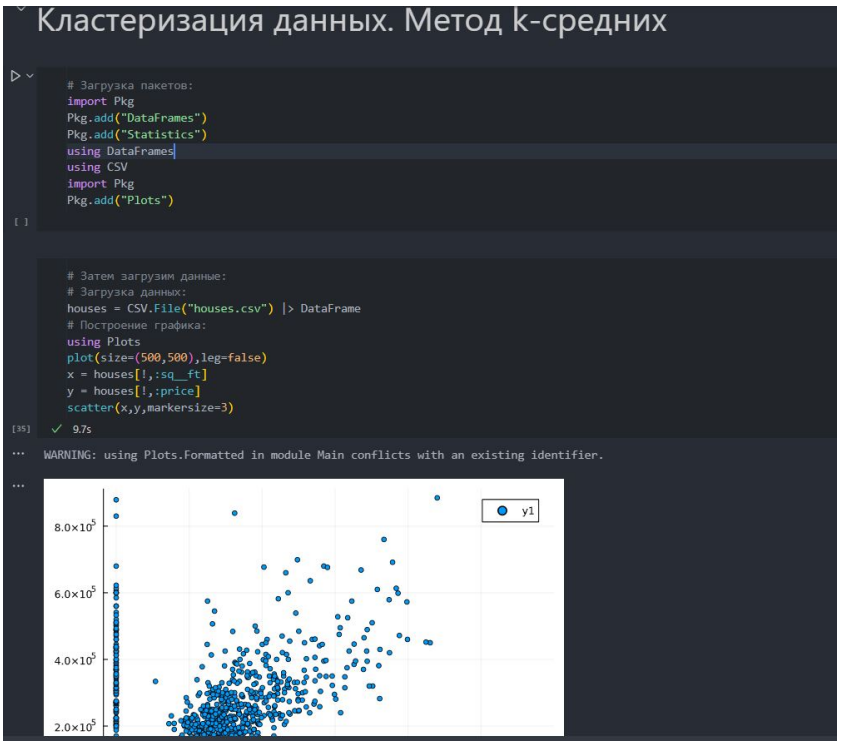

Рис. 3.9: Примеры Кластеризация

Рис. 3.10: Примеры Кластеризация

```
for uzip in unique_zips
subs = filter_houses[filter_houses[!,:zip].==uzip,:]
x = subs[!,:latitude]
y = subs[!,:longitude]
scatter!(zips_figure,x,y)
end
xlabel!("Latitude")
ylabel!("Longitude")
title!("Houses color-coded by zip code")
display(zips_figure)
```

[49] ⊗ 0.9s

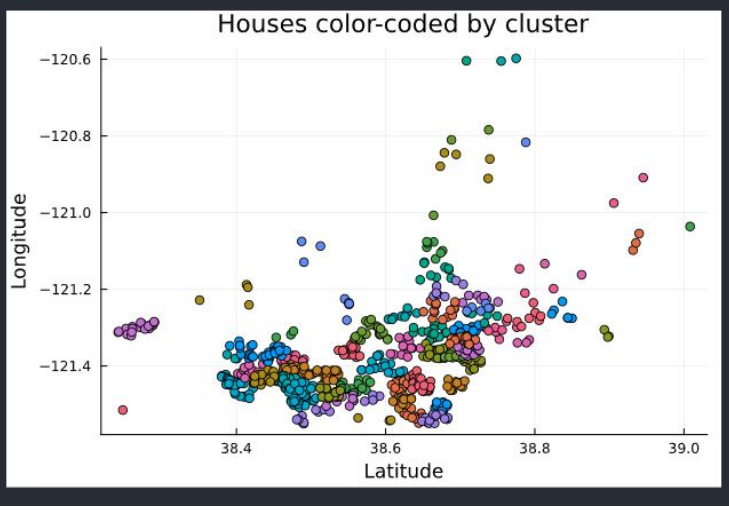

Рис. 3.11: Примеры Кластеризация

```
knearest = 10
id = 70
point = X[:,id]
# Поиск ближайших соседей:
kdtree = KDTree(X)
idxs, dists = knn(kdtree, point, knearest, true)
# Отобразим на графике соседей выбранного объекта недвижимости (рис. 7.5):
# Все объекты недвижимости:
x = filter_houses[!,:latitude];
y = filter_houses[!,:longitude];
scatter(x,y)
# Соседи:
x = filter_houses[idxs,:latitude];
y = filter_houses[idxs,:longitude];
scatter!(x,y)
```

[51] ✓ 0.3s

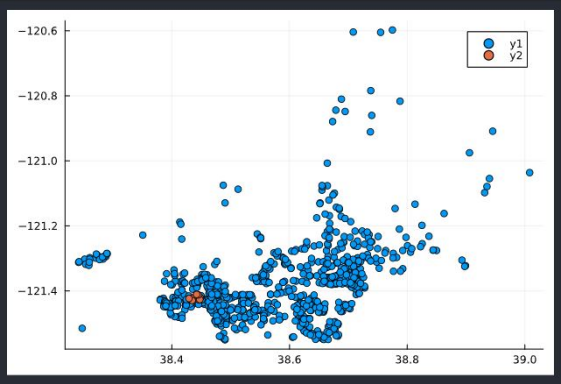

Рис. 3.12: Примеры Кластеризация

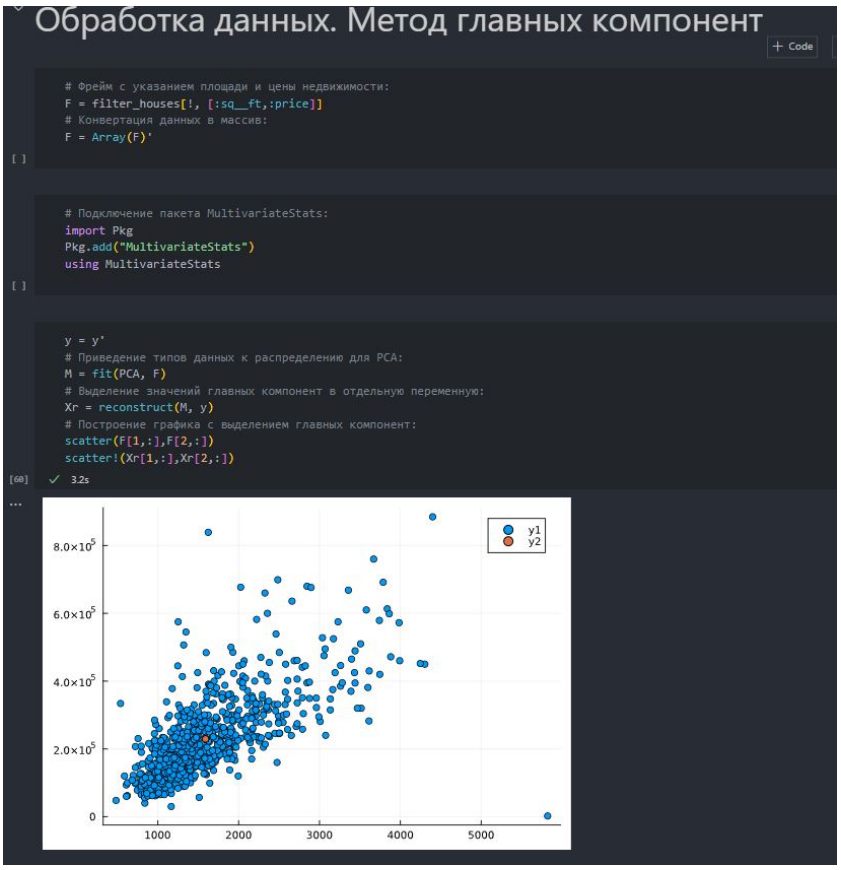

Рис. 3.13: Примеры Метод главных компанент

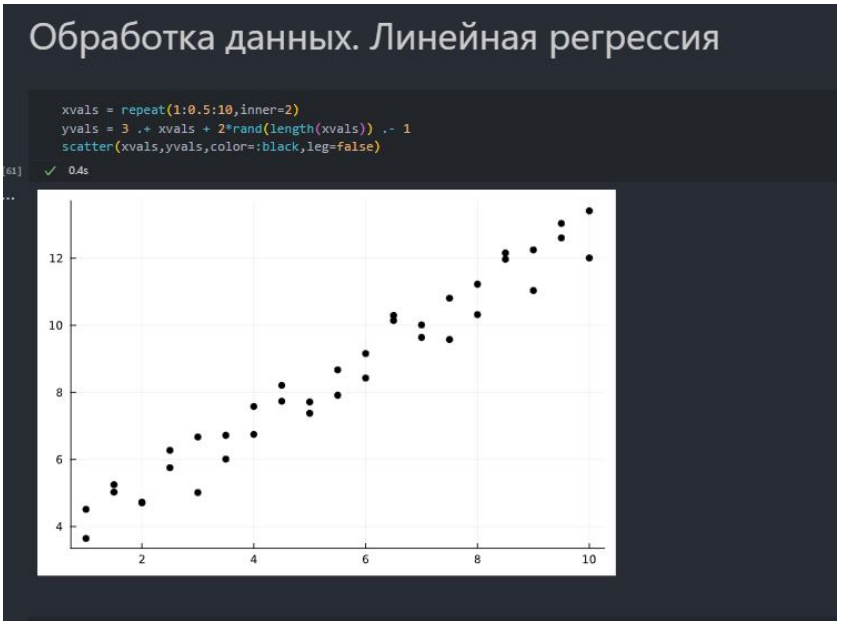

Рис. 3.14: Примеры Линейная регрессия

```
# Определим функцию линейной регрессии:
function find_best_fit(xvals,yvals)
meanx = mean(xvals)
meany = mean(yvals)
stdx = std(xvals)
stdy = std(yvals)
r = cor(xvals,yvals)
a = r*stdy/stdx
b = meany - a*meanx
return a,b
end


# Применим функцию линейной регрессии для построения соответствующего графика
# значений (рис. 7.8):
a,b = find_best_fit(xvals,yvals)
ynew = a * xvals .+ b
plot!(xvals,ynew)
```

[63] ✓ 0.1s

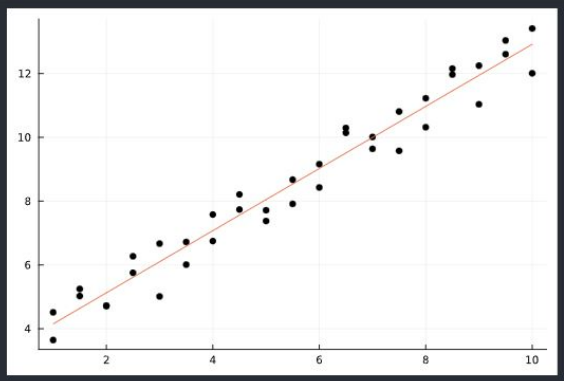

Рис. 3.15: Примеры Линейная регрессия

Затем выполним задания(рис. fig. 3.16 - fig. 3.15)

```julia
using RDatasets
iris = dataset("datasets", "iris") |> DataFrame
iris_spec = Dict("setosa"=>1, "versicolor"=>2, "virginica"=>3)
iris[!,:5] = [iris_spec[item] for item in iris[!,:5]]

X = Matrix(iris)'

#Зададим число кластеров
k = length(unique(iris[!,:5]))
# Определение k-среденго
C = kmeans(X,k)

# DataFrame
df = DataFrame(cluster = C.assignments, SepalLength = iris[!, :SepalLength],
SepalWidth = iris[!,:SepalWidth], PetalLength = iris[!,:PetalLength],
PetalWidth = iris[!,:PetalWidth], Species = iris[!,:Species]);

cluster_fig = plot(legend = false)
for i=1:k
    clustered_houses = df[df[!,:cluster].==i, :]
    xvals = clustered_houses[!, :SepalWidth]
    yvals = clustered_houses[!,:SepalLength]
    scatter!(cluster_fig, xvals,yvals,markersize=4)
end
xlabel!("SepalWidth")
ylabel!("SepalLength")
title!("Color-Coded")
display(cluster_fig)
```

✓ 0.1s

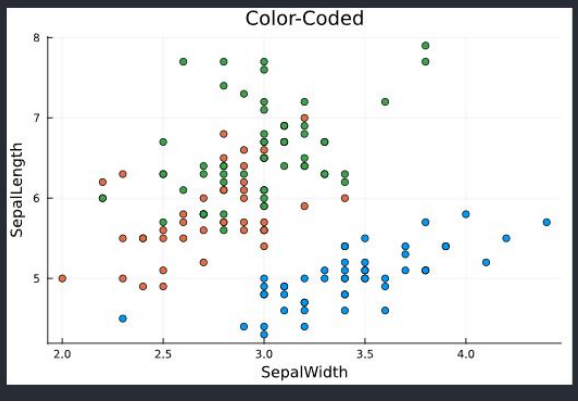

Рис. 3.16: Задания 1

Рис. 3.17: Задания 1

Рис. 3.18: Задания 2

4 Выводы

В результате выполнения данной лабораторной работы было освоено использование специализированных пакетов Julia для обработки данных.

Список литературы

1. JuliaLang [Электронный ресурс]. 2024 JuliaLang.org contributors. URL: https://julialang.org/ (дата обращения: 11.10.2024).
2. Julia 1.11 Documentation [Электронный ресурс]. 2024 JuliaLang.org contributors. URL: https://docs.julialang.org/en/v1/ (дата обращения: 11.10.2024).