

Лабораторная работа №5.

Построение графиков

Тазаева А. А.

Российский университет дружбы народов, Москва, Россия

Цели работы

Основная цель работы — освоить синтаксис языка Julia для построения графиков.

Задание

1. Используя Jupyter Lab, повторите примеры из раздела 5.2.
2. Выполните задания для самостоятельной работы (раздел 5.4).

```
(@v1.10) pkg> add Plots.jl
  Resolving package versions...
  No Changes to `C:\Users\noname\.julia\environments\v1.10\Project.toml`
  No Changes to `C:\Users\noname\.julia\environments\v1.10\Manifest.toml`
Precompiling project...
 29 dependencies successfully precompiled in 16 seconds. 162 already precompiled.
```

Рис. 1: Установка пакетов для julia

Основные пакеты для работы с графиками в Julia

Основные пакеты для работы с графиками в Julia

```
using Pkg
Pkg.add("Plots")
Pkg.add("PyPlot")
Pkg.add("Plotly")
Pkg.add("UnicodePlots")
# подключаем для использования Plots:
using Plots
```

```
⌕[32m⌕[1m Resolving⌕[22m⌕[39m package versions... ●●●
```

Рассмотрим построение графика функции $f(x) = (3x^2 + 6x - 9)e^{-0.3x}$ разными способами. Фактически для построения графика функции требуется иметь массив соответствующих значений x и y .

```
# задание функции:
f(x) = (3x.^2 + 6x .- 9).*exp.(-0.3x)
```

```
f (generic function with 1 method)
```

```
# генерирование массива значений x в диапазоне от -5 до 10 с шагом 0,1
# (шаг задан через указание длины массива):
x = collect(range(-5,10,length=151))
```

```
151-element Vector{Float64}: ●●●
```

```
# генерирование массива значений y:
y = f(x)
```

```
151-element Vector{Float64}: ●●●
```

Рис. 2: Примеры. Основные пакеты для работы с графиками в Julia. ч.1

Задание 1

```
##### задает массив для x
x = collect(range(0.2, pi, length = 50))
# задает функцию sin(x)
f(x) = sin(x)
# графики
plotly()
serialtypes = [:line, :scatter, :step, :bar, :sticks, :hline, :vline, :path]
titles = ["line" "scatter" "step" "bar" "sticks" "hline" "vline" "path"]
plot(x,f,
     st=serialtypes,
     title = titles,
     layout=(4,4),
     size=(1200,600)
)
```

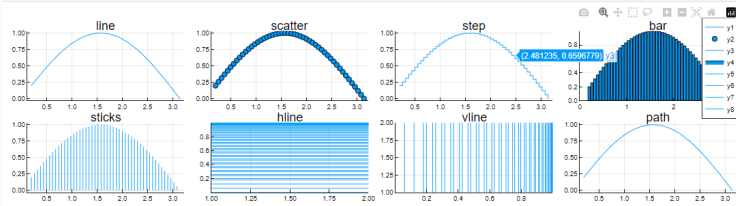
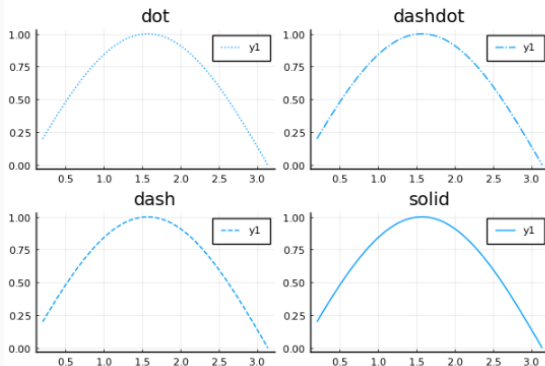


Рис. 3: Задание 1

Задание 2

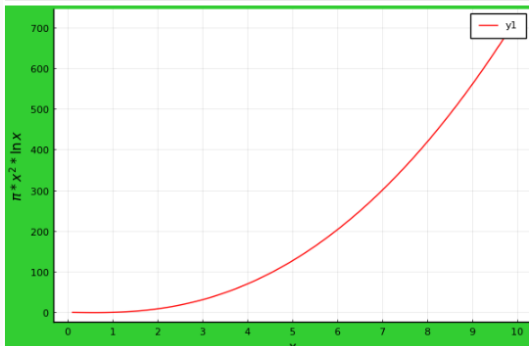
```
titles = ["dot" "dashdot" "dash" "solid"]
```

```
p1=plot(x,f, linestyle=:dot)  
p2=plot(x,f, linestyle=:dashdot)  
p3=plot(x,f, linestyle=:dash)  
p4=plot(x,f, linestyle=:solid)  
plot(p1,p2,p3,p4,  
     title = titles,  
     layout=(2,2)  
)
```



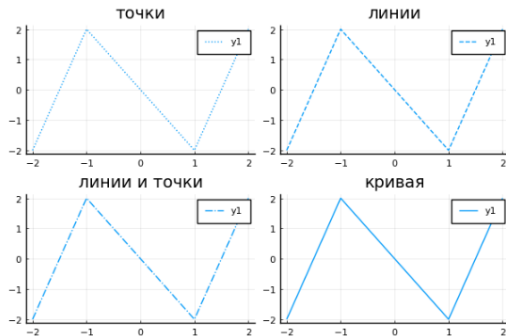
Задание 3

```
#задаем шрифт надписей
using LaTeXStrings
#задаем функцию
x=collect(range(0,10,100))
f(x) = pi*x^2*log(x)
plot(x,f,
    xlabel="x",
    ylabel=L"\pi*x^2*\ln{x}",
    background_color_outside = :limegreen,
    linecolor = :red,
    xticks=(0:1:10),
    yticks=(0:100:1000)
)
```



Задание 4

```
x = [-2, -1, 0, 1, 2]
f(x) = x**3-3*x
titles = ["точки" "линии" "линии и точки" "кривая"]
p1=plot(x,f, line=:dot)
p2=plot(x,f, line=:dash)
p3=plot(x,f, line=:dashdot)
p4=plot(x,f, line=:solid)
plot(p1,p2,p3,p4,
     title = titles,
     layout=(2,2)
)
savefig("figure_tazaeva.png")
```



"C:\\Users\\noname\\figure_tazaeva.png"

Задание 5

```
x=collect(3:0.1:6)
y1(x)=pi*x
y2(x)=exp(x)*cos(x)
plot(x,[y1,y2],legend=true,grid=true, layout=(1,2), size=(1000,400))
```

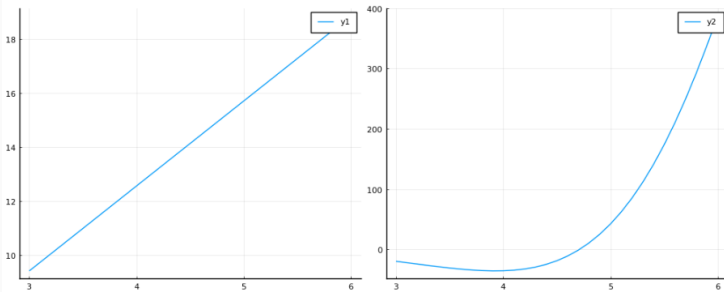
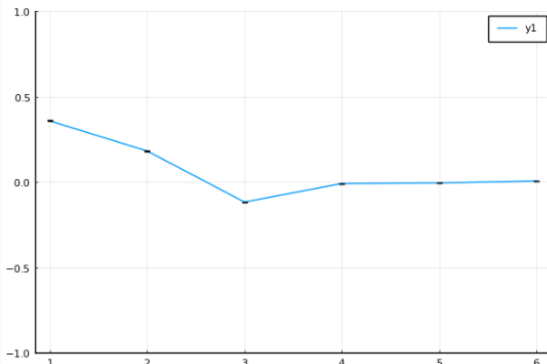


Рис. 7: Задание 5

Задание 6

```
sds = [1, 1/2, 1/4, 1/8, 1/16, 1/32]
n = 10
y = [mean(sd*randn(n)) for sd in sds]
errs = 2.45 * sds / n*0.01
plot(y,
      ylims = (-1,1),
)
plot(y,
      ylims = (-1,1),
      err = errs
)
```



Задание 7

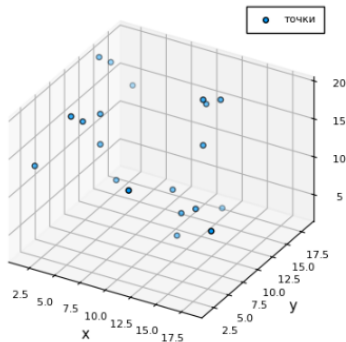
```
n = 50
x = rand(1:20, 20)
y = rand(1:20, 20)
scatter(x,y,
        legend=true,
        label = "точки",
        xlabel="x",
        ylabel="y",
        title="Задание 7. Точечный график случайных чисел")
```



Задание 8

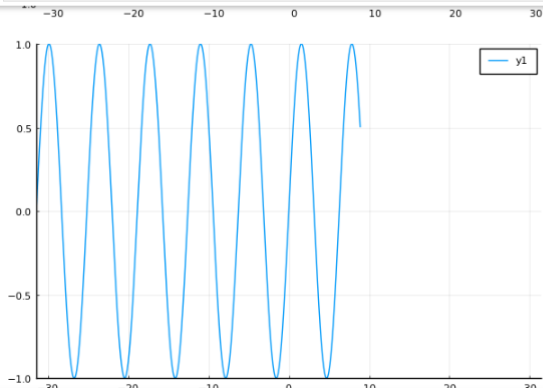
```
n = 50
x = rand(1:20, 20)
y = rand(1:20, 20)
z = rand(1:20, 20)
scatter(x,y,z,
        legend=true,
        label = "точки",
        xlabel="x",
        ylabel="y",
        title="Задание 8. 3D - Точечный график случайных чисел")
```

Задание 8. 3D - Точечный график случайных чисел



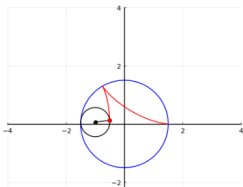
Задание 9

```
n=1000
f(x)=sin(x)
c = collect(range(-10*pi, 10*pi, n))
a=@animate for i in 1:n
    plot(xlim=(-10*pi,10*pi), ylim=(-1,1))
    x=c[1:i]
    y=map(f,x)
    plot!(x,y)
end
gif(a, "task9.gif")
```



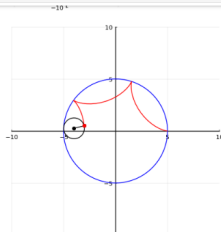
Самостоятельная работа

```
# радиус малой окружности:
rr = 0.5
# коэффициент для построения большой окружности:
k = 3
# число отсчётов:
n = 100
# массив значений угла  $\theta$ :
# theta from 0 to 2pi ( + a little extra)
 $\theta = \text{collect}(0:2*\pi/100:2*\pi+2*\pi/100)$ 
# массивы значений координат:
X = rr*k*cos.( $\theta$ )
Y = rr*k*sin.( $\theta$ )
a=@animate for i in 1:n
    # задаём оси координат:
    plt=plot(5,xlim=(-4,4),ylim=(-4,4), c=:red, aspect_ratio=1, legend=false, framestyle=:origin)
    # большая окружность:
    plot!(plt, X,Y, c=:blue, legend=false)
    i = 50
    t =  $\theta[1:i]$ 
    # гипотенуза:
    x = rr*(k-1)*cos.(t) + rr*cos.((k-1)*t)
    y = rr*(k-1)*sin.(t) - rr*sin.((k-1)*t)
    plot!(x,y, c=:red)
    # малая окружность:
    xc = rr*(k-1)*cos(t[end]) .+ rr*cos.( $\theta$ )
    yc = rr*(k-1)*sin(t[end]) .+ rr*sin.( $\theta$ )
    plot!(xc,yc,c=:black)
    # радиус малой окружности:
    x1 = transpose([rr*(k-1)*cos(t[end]) x[end]])
    y1 = transpose([rr*(k-1)*sin(t[end]) y[end]])
    plot!(x1,y1,marker=:circle,marker=:circle,markerstroke=:black,markerstrokecolor=:red)
    scatter!([x[end]], [y[end]], c=:red, markerstrokecolor=:red)
end
gif(a,"task10.gif")
```



Задание 11

```
# радиус малой окружности:
rr = 1
# коэффициенты для построения большой окружности:
k = 5
# число отсчётов:
n = 100
# массив значений угла  $\theta$ :
# theta from 0 to  $2\pi$  ( + a little extra)
 $\theta = \text{collect}(\theta:2*\pi/100:2*\pi*2/100)$ 
# массивы значений координат:
X = rr*cos.( $\theta$ )
Y = rr*sin.( $\theta$ )
 $\text{animate} \text{ for } i \text{ in } 1:n$ 
    # задаём оси координат:
    plt=plot(5,xlim=(-10,10),ylim=(-10,10), c=:red, aspect_ratio=1, legend=false, framestyle=:origin)
    # большая окружность:
    plot!(plt, X,Y, c=:blue, legend=false)
    i = 50
    t =  $\theta[i:i]$ 
    # гипотенуза:
    x = rr*(k-1)*cos.(t) + rr*cos.((k-1)*t)
    y = rr*(k-1)*sin.(t) - rr*sin.((k-1)*t)
    plot!(x,y, c=:red)
    # малая окружность:
    xc = rr*(k-1)*cos(t[end]) + rr*cos.( $\theta$ )
    yc = rr*(k-1)*sin(t[end]) + rr*sin.( $\theta$ )
    plot!(xc,yc,c=:black)
    # радиус малой окружности:
    x1 = transpose([rr*(k-1)*cos(t[end]) x[end]])
    y1 = transpose([rr*(k-1)*sin(t[end]) y[end]])
    plot!(x1,y1,markershape=:circle,markersize=4,c=:black)
    scatter!([x[end]],y[end],c=:red, markerstrokecolor=:red)
end
gif(n,"task11.gif")
```



Выводы по проделанной работе

В ходе лабораторной работы мною был освоен синтаксис языка Julia для построения графиков.