

Лабораторная работа №5

Леснухин Даниил Дмитриевич Российский университет
дружбы народов Москва

Цель работы

Освоение синтаксиса языка Julia для построения графиков.

Задание

- 1 Используя JupyterLab, повторить примеры, дополнив графики обозначениями осей, легендами, названиями.
- 2 Выполнить задания для самостоятельной работы.

Теоретическое введение

Julia — высокоуровневый язык программирования с динамической типизацией для математических вычислений.

Эффективен для программ общего назначения. Синтаксис близок к другим математическим языкам, но имеет отличия.

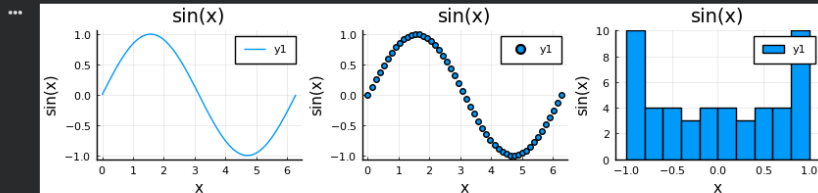
Использовалась официальная документация Julia.

Задания для самостоятельной работы

Задание №1

Все графики отображены в одном окне.

```
x = collect(range(0, 2*pi, length =50))  
f(x) = sin(x)  
p1 = Plots.plot(x, f, title = "sin(x)", xlabel = "x", ylabel= "sin(x)")  
p2 = Plots.plot(x, f, title = "sin(x)", xlabel = "x", ylabel= "sin(x)", seriestype = :scatter)  
p3 = Plots.histogram(x, f, bins = 10, title= "sin(x)", xlabel = "x", ylabel = "sin(x)")  
Plots.plot(p1, p2, p3, layout=(1,3), size=(800, 200))
```



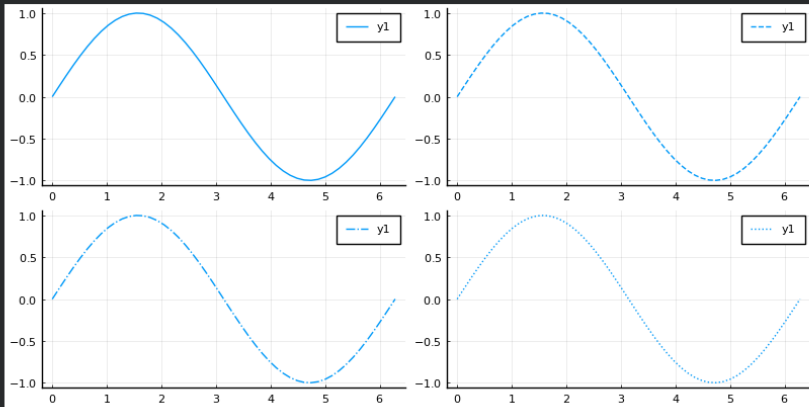
sys:1: UserWarning: No data for colormapping provided via 'c'. Parameters 'vmin', 'vmax' will be ignored

Задание №2



```
titles = ["solid", "dash", "dot", "dashdot"]  
p1 = Plots.plot(x, f, linestyle=:solid)  
p2 = Plots.plot(x, f, linestyle=:dash)  
p3 = Plots.plot(x, f, linestyle=:dashdot)  
p4 = Plots.plot(x, f, linestyle=:dot)  
Plots.plot(p1, p2, p3, p4, layout=(2,2), size=(800, 400))
```

...



Задание №3

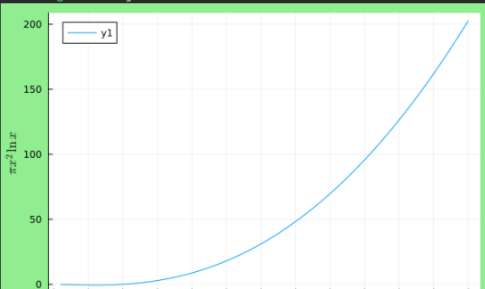
```
using Pkg
Pkg.add("LaTeXStrings")
using LaTeXStrings

using Plots
gr()

f(x) = π * x^2 * log(x)
x = 0.1:0.01:6

plot(x, f,
      xlabel = L"x",
      ylabel = L"\pi x^2 \ln x",
      background_color_outside = :lightgreen,
      xticks = 0:0.5:6
    )
```

```
*** Resolving package versions...
      Updating `~/julia/environments/v1.11/Project.toml`
[b964fa9f] + LaTeXStrings v1.4.0
No Changes to `~/julia/environments/v1.11/Manifest.toml`
```

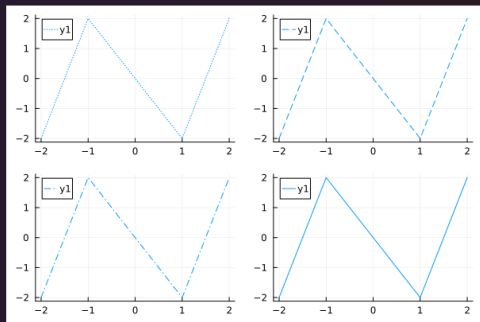


Задание №4

```
x = [-2, -1, 0, 1, 2]
f(x) = x^3 - 3 * x
p1 = Plots.plot(x, f, lines=:dot)
p2 = Plots.plot(x, f, lines=:dash)
p3 = Plots.plot(x, f, lines=:dashdot)
p4 = Plots.plot(x, f, lines=:solid)
Plots.plot(p1, p2, p3, p4, layout = (2, 2))
Plots.savefig("figure_lesnukhin.png")
```

*** "/content/figure_lesnukhin.png"

Изменить

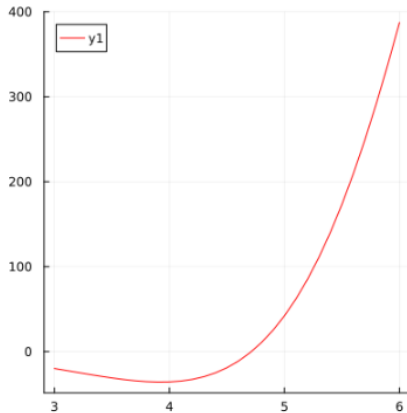
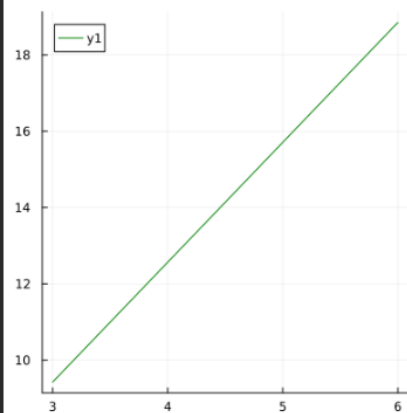


Задание №5

```
x = collect(3:0.1:6)
f(x) = pi*x

g2(x) = exp(x)*cos(x)

p1 = Plots.plot(x, f, c="green")
p2 = Plots.plot(x, g2, c="red")
Plots.plot(p1, p2, layout=(1,2), size=(800, 400), legend = true, grid = true)
```



Задание №6



using Statistics

```
sds = [1, 0.8, 0.6, 0.4, 0.2, 0.1]
```

```
n = 10
```

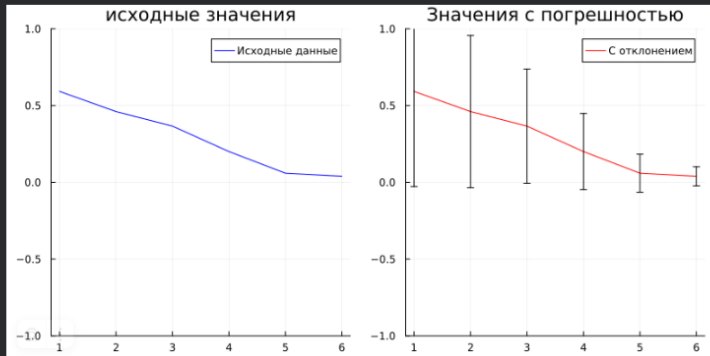
```
y = [mean(sd * rand(n)) for sd in sds]
```

```
errs = 1.96 .* sds ./ sqrt(n)
```

```
p1 = Plots.plot(y, ylims=(-1, 1), label = "Исходные данные", title = "исходные значения", color = :blue)
```

```
p2 = Plots.plot(y, ylims=(-1, 1), err = errs, label = "С отклонением", title = "Значения с погрешностью", color = :red)
```

```
Plots.plot(p1, p2, layout=(1, 2), size = (800, 400))
```



Задание №7

7

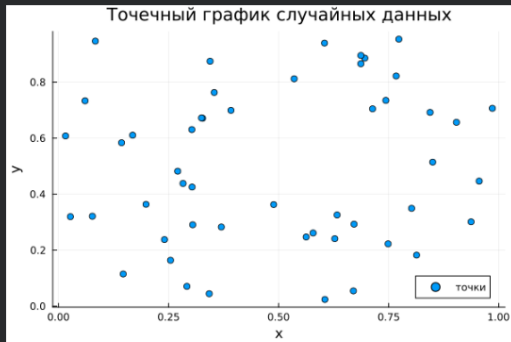
112]

0
СВК.



```
n= 50  
x = rand(n)  
y = rand(n)  
Plots.scatter(x, y, title = "Точечный график случайных данных", legend = true, label = "точки", xlabel="x", ylabel= "y" )
```

▼



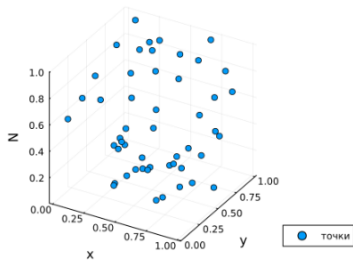
Задание №8



```
n = 50  
x = rand(n)  
y = rand(n)  
z = rand(n)  
Plots.scatter(x, y, z, title = "Точечный график случайных данных", legend = true, label = "точки", xlabel="x", ylabel="y", zlabel="z")
```

...

Точечный график случайных данных



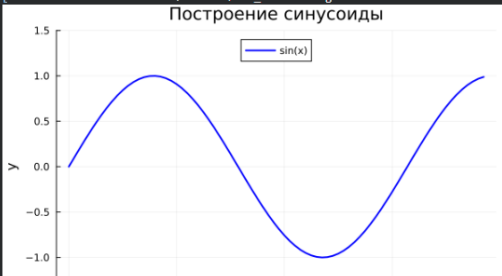
Задание №9

```
using Plots
gr()

anim = @animate for i in 1:50
    x = 0:0.1:(2π + 0.1 * i)
    y = sin.(x)
    plot(x, y,
        label = "sin(x)",
        color = :blue,
        lw = 2,
        legend = :top,
        ylim = (-1.5, 1.5),
        xlabel = "x",
        ylabel = "y",
        title = "Построение синусоиды"
    )
end

gif(anim, "sin_animation.gif", fps = 20)
```

*** [Info: Saved animation to /content/sin_animation.gif



Задание №10

```
let
using Plots
gr()

# Параметры
r = 1
ks = [2, 3, 3//2, 5//2] # два целых и два рациональных
n_frames = 100
θ = range(0, stop=2π, length=n_frames+1) # параметры угла

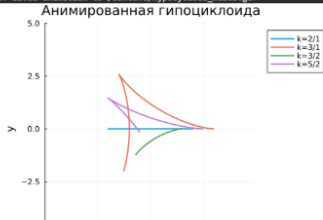
# Создаём анимацию
anim = @animate for i in 1:n_frames
    plt = plot(xlim=(-5,5), ylim=(-5,5),
        aspect_ratio=1,
        legend=:outertopright,
        title="Анимированная гипоциклоида",
        xlabel="x", ylabel="y")

    for k in ks
        t = θ[i:i]
        x = r*(k-1)*cos.(t) + r*cos.((k-1).*t)
        y = r*(k-1)*sin.(t) - r*sin.((k-1).*t)

        label_str = isa(k, Rational) ? "k=$(Int(numerator(k)))/$(Int(denominator(k)))" : "k=$k"
        plot!(plt, x, y, lw=2, label=label_str)
    end
end

# Сохраняем в gif
gif(anim, "hypocycloid_multi.gif", fps=20)
end
```

*** [Info: Saved animation to /content/hypocycloid_multi.gif



Выводы

В ходе выполнения лабораторной работы:

- Освоен синтаксис Julia для построения графиков
- Получены навыки оформления графиков с легендами, подписями осей и названиями