Hwk01

September 25, 2019

1 Homework 01

Congratulations! You've managed to open this Juypter notebook on either Github or on your local machine.

Help for Jupyter Notebooks can be found in the Jupyter Lab by going to `Help > Notebook Reference`. You can also go to the Notebook basics documentation.

The basics are that you can write *markdown* cells that have math, words, and other markdown (like Rmarkdown!)… and *code* cells that, well, have code and display the output below (as in Rmarkdown). Switch *mode* to change between the two (sidebar will change colors).

If you want to export your document to .pdf (you don't have to!), you an go to `File > Export Notebook As... > PDF`. To do this, I had to install the *Inkscape* application. (I did this with the Chocolatey package manager on Windows. You can probably do this with Homewbrew on a Mac)

2 Instructions

Your homework is to generate 3d plots of a quadratic function in $\mathbb{R}^2$ and to examine the relationship between eigenvalues of the Hessian matrices, shapes of the functions, and the (possible) existence of minima and maxima.

You can find the documentation for `Plots.jl` at `http://docs.juliaplots.org/latest/`

For the following functions

$$
\begin{aligned}
f^a(x,y) &= -x^2 - y^2 && (1)\\
f^b(x,y) &= -x^2 + xy - y^2 && (2)\\
f^c(x,y) &= -x^2 + 2xy - y^2 && (3)\\
f^d(x,y) &= -x^2 + 3xy - y^2 && (4)
\end{aligned}
$$

1. Write the Hessian matrix in LaTeX
2. Compute the determinants by hand. Are the Hessians PD, PSD, NSD, or ND? What does this imply about convexity / concavity of the function? What about the existence of a minimum or maximum over the domain $\mathbb{R}^2$?
3. `@assert` statements are wonderful to include in your functions because they make sure that the inputs meet certain assumptions… such as that the length of two vectors is the same. Using them regularly can help you avoid errors

Use an `@assert` statement to check that your determinants computed by hand are correct. See what Julia does when you put the wrong determinanmt in. See `LinearAlgebra.det` docs

```
@assert det(Ha) == ???
```

4. Compute the eigenvalues of your matrix using `LinearAlgebra.eigvals`
5. Create a function in Julia to compute f^a, f^b, f^c, f^d as above. Plot them!

To submit your homework, commit this notebook to your personal homework repo, push it, and issue a pull request to turn it in to me.

3 Top of your Julia Code

[1]:
```julia
# Load libraries first of all
using LinearAlgebra  # load LinearAlgebra standard library
using Plots          # loads the Plots module
```

[2]:
```julia
# tells Plots to use the GR() backend.
#    Note: for interactive 3d plots, you can also install
#    PyPlot or PlotlyJS and try using those. You might
#    need to use Atom or the REPL to get the interactivity
gr()
```

[2]:
```
Plots.GRBackend()
```

[3]:
```julia
# define a range we can iterate over
xrange = -3.0 : 0.1 : 3.0
```

[3]:
```
-3.0:0.1:3.0
```

4 Question 1

$$f^a(x, y) = -x^2 - y^2$$

4.1 Part 1a

Hessian is $H^a = \begin{bmatrix} ? & ? \\ ? & ? \end{bmatrix}$

4.2 Part 1b

Determinant is $|H^a| = ?$, so the matrix is ???, and it has a global minimum / maximum at ????

[4]:
```julia
# define the Hessian for H^a.
# Note:
#     Julia in Jupyter Notebooks and Atom can handle latexy characters
#     I got fancy by typing H\^a [tab] and getting a superscript
#     We could have also gotten greek letters with \beta [tab]
```

```
#     or (very important) approximately equals with \approx [tab]

H = [-2 4 ; 2 3]
```

[4]:
```
2×2 Array{Int64,2}:
 -2  4
  2  3
```

4.3 Part 1c

[5]:
```
@assert det(H ) == 3
```

```
AssertionError: det(H ) == 3



Stacktrace:

 [1] top-level scope at In[5]:1
```

4.4 Part 1d

[6]:
```
eigvals(H )
```

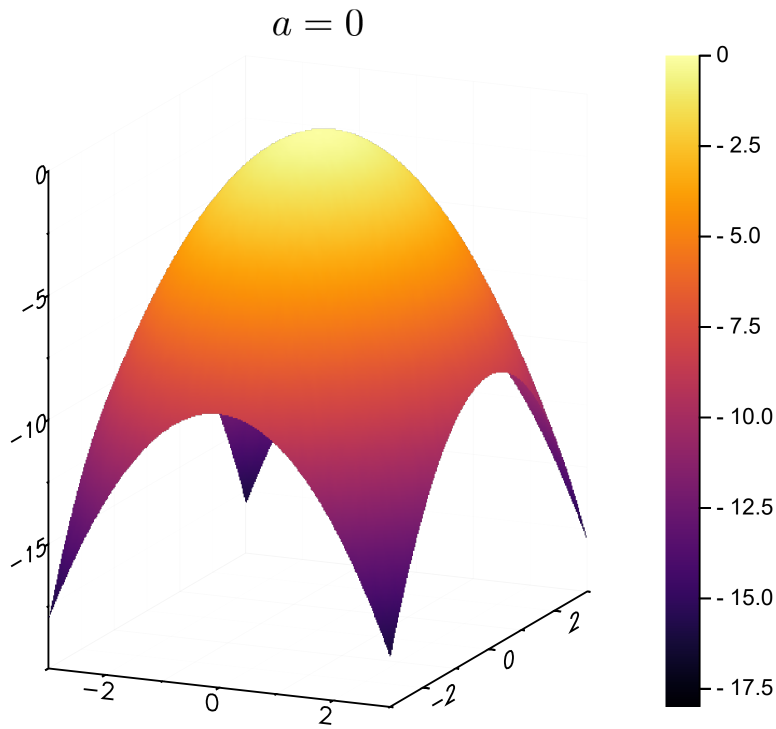
[6]:
```
2-element Array{Float64,1}:
 -3.274917217635375
  4.274917217635375
```

[7]:
```
# functions to plot
fa(x,y) = -x^2 - y^2
```

[7]:
```
fa (generic function with 1 method)
```

[8]:
```
plot(xrange, xrange, fa, st = :surface, title = "\$a=0\$")
```

[8]:



5 Question 2

6 Question 3

7 Question 4