Analytics for a Better World

Recitation 1

Schedule

1. A geometric Linear Optimization example.
2. Installing Julia, JuMP, Gurobi, and Jupyter workshop.

Draw the problems:

1.
$$\begin{aligned} \min_{x,y} \quad & 2x + y \\ \text{s.t.} \quad & -x + 2y \leq 2 \\ & -3x + 2y \geq -6 \\ & 4x + 3y \leq 12 \end{aligned}$$

2.
$$\begin{aligned} \min_{x,y} \quad & 2x + y \\ \text{s.t.} \quad & -x + 2y \leq 2 \\ & -3x + 2y \geq -6 \\ & 4x + 3y \geq 12 \end{aligned}$$

On the same graph:

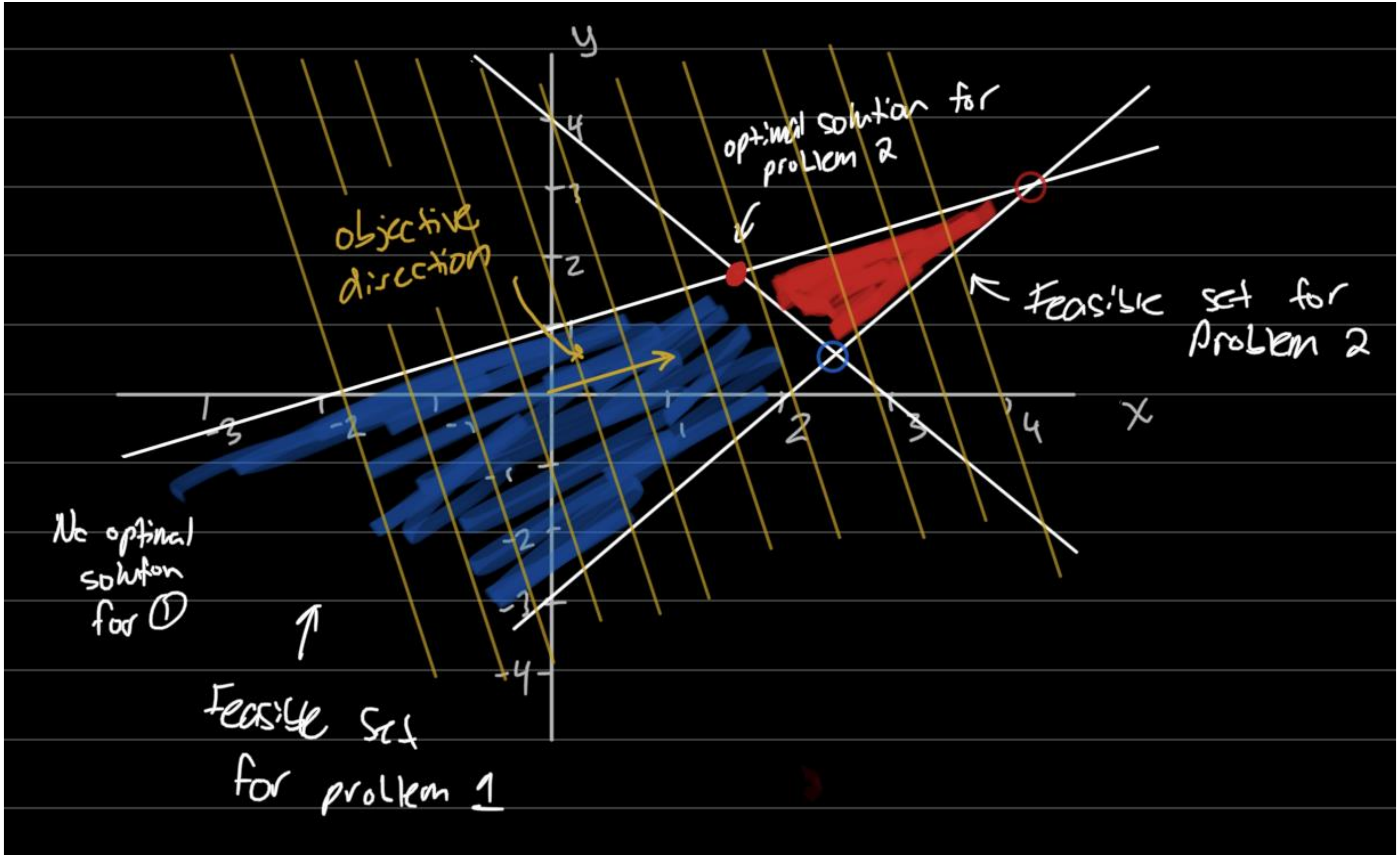
- Graph a line showing the boundary of each of the constraints.
- Shade the feasible region
- Draw the direction of the gradient and a contour line of the objective function
- Finally, identify what the optimal solution and associated objective value of the problem is.

Follow-up Questions

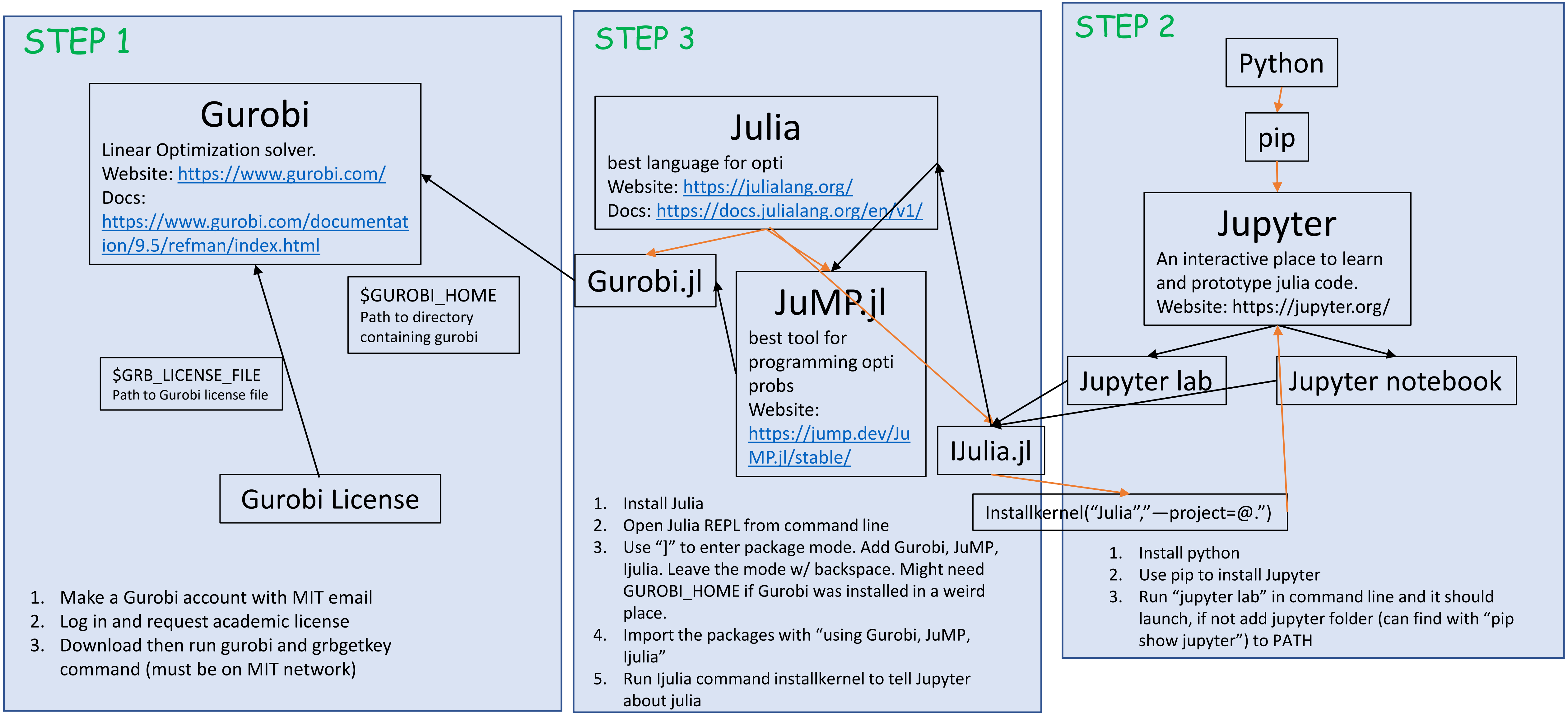
1. How can I change the inequality directions to make no feasible solution exist?
2. Where are the optimal solutions if I maximize instead of minimize?
3. If I scale the objective function, how does the optimal solution change?

Answers

1. How can I change the inequality directions to make no feasible solution exist? Flip inequalities 1 and 2 in problem 1, for instance.
2. Where are the optimal solutions if I maximize instead of minimize? See picture below
3. If I scale the objective function, how does the optimal solution change? If scaled by a positive, the optimal solution (x*,y*) doesn’t change because the direction hasn’t changed. However, the optimal cost does get scaled! If scaled by a negative, the optimal solutions become the optimal solutions for the maximization problem with the original objective function.


y
x
4
3
2
-1
-2
-3
-4
-3
-2
-1
1
2
3
4
objective direction
optimal solution for problem 2
Feasible set for Problem 2
No optimal solution for ①
Feasible set for problem 1

Software Installation



STEP 1

Gurobi

Linear Optimization solver.
Website: https://www.gurobi.com/
Docs: https://www.gurobi.com/documentation/9.5/refman/index.html

$GUROBI_HOME
Path to directory containing gurobi

$GRB_LICENSE_FILE
Path to Gurobi license file

Gurobi License

1. Make a Gurobi account with MIT email
2. Log in and request academic license
3. Download then run gurobi and grbgetkey command (must be on MIT network)

STEP 3

Julia

best language for opti
Website: https://julialang.org/
Docs: https://docs.julialang.org/en/v1/

Gurobi.jl

JuMP.jl

best tool for programming opti probs
Website: https://jump.dev/JuMP.jl/stable/

IJulia.jl

1. Install Julia
2. Open Julia REPL from command line
3. Use “]” to enter package mode. Add Gurobi, JuMP, Ijulia. Leave the mode w/ backspace. Might need GUROBI_HOME if Gurobi was installed in a weird place.
4. Import the packages with “using Gurobi, JuMP, Ijulia”
5. Run Ijulia command installkernel to tell Jupyter about julia

STEP 2

Python

pip

Jupyter

An interactive place to learn and prototype julia code.
Website: https://jupyter.org/

Jupyter lab

Jupyter notebook

Installkernel(“Julia”,”—project=@.”)

1. Install python
2. Use pip to install Jupyter
3. Run “jupyter lab” in command line and it should launch, if not add jupyter folder (can find with “pip show jupyter”) to PATH

*Orange means installation related.
*Black means usage related.

Other good options for writing julia code…

Windows Linux Subsystem

Windows Terminal

vim

emacs

VS Code

Julia Extension