L1 Regularized Least Squares

ECE367 PS05 Problem 5.4 -- Aman Bhargava

Optimization Problem:

$$\min_{x \in \mathbb{R}^2} ||Ax - y||_2^2 + \gamma ||x||_1$$

For $0 < \gamma \in \mathbb{R}$. The minimizing x is denoted x_γ^*.

Part A

Consider

$$A = \begin{bmatrix} 0 & 2 \\ 1 & 0 \\ 2 & 1 \end{bmatrix}, \quad y = \begin{bmatrix} -2 \\ 5 \\ 9 \end{bmatrix}, \quad \gamma = 0.$$

Derive that $x_0^* = (5, -1)$.

Part A Solution

We re-derive using the formula derived in class for solution to unconstrained least-squares:

$$x^* = (A^T A)^{-1} A^T y$$

By the computation in cell 1:

$$\therefore x^* = (5, -1)$$

Part B

Figure 2 is a plot of x_γ^* as a function of γ. We observe that there is a range of γ where $x_{\gamma,2}^* = 0$. What is the smallest value of $\gamma = \gamma_{\min}$ such that $x_{\gamma,2}^* = 0$? Also, what is the value of $x_{\gamma_{\min},1}^*$?

Part B Solution

We let $p = ||Ax - y||_2^2 + \gamma ||x||_1$

$$\rightarrow p = (2x_2 + 2)^2 + (x_1 - 5)^2 + (2x_1 + x_2 - 9)^2 + \gamma(|x_1| + |x_2|)$$

$$\rightarrow p = 2x_1^2 + 4x_2 x_1 - 28x_1 + 8x_2^2 - 28x_2 + 110 + \gamma|x_1| + \gamma|x_2|$$

At the optimal value of $\vec{x}$ for a given γ, we would have $\nabla \vec{x} = \vec{0}$.

Therefore, we set $\nabla p = \vec{0}$ and $x_2 = 0$ and solve for x_1, γ.

$$\nabla p = \begin{bmatrix} -46 + 10x_1 + 4x_2 + (\gamma x_1)/|x_1| \\ -10 + 4x_1 + 10x_2 + (\gamma x_2)/|x_2| \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

By inspection, $x_1 > 0$ and $x_2 \to 0^-$.

$$\nabla p = \begin{bmatrix} -46 + 10x_1 + \gamma \\ -10 + 4x_1 - \gamma \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Solving as a system of linear equations (two constraints, two unknowns),

$$-56 + 14x_1 = 0; \gamma + 10x_1 - 46 = 0$$

$$\therefore (x_1, \gamma) = (4, 6)$$

Part C

It can be ovserved that $\gamma > \gamma_{\min}$ leads to x_γ^* of the form $(x_{\gamma 1}^*, 0)$. Explain why.

Part C Answer

- The second index of x_γ^* was 'eliminated' first because, of the two indexes, it explained less of the linear correlation between rows of A and indexes of y.
- As γ is increased, this does not change, so the index remains 0-valued.
- Geometrically, the cost associated with the regularization term has square cross sections with corners aligned with the x_1, x_2 axes.
- This axis alignment means that, once reached, the path of steepest descent is along an axis as the regularization parameter γ is increased.
- Therefore, $x_{\gamma,2}^* = 0$ for all $\gamma \geq \gamma_{\min}$.

Part D

Solve for x_γ^* for $\gamma \geq \gamma_{\min}$.

Part D: Solution

For $\gamma \geq \gamma_{\min}$, we know from C that $x_2 = 0$. We can re-use the same logic from A (i.e. that $\nabla p = \vec{0}$ at the optimal x_1, x_2, knowing that $x_2 = 0$:

$$\nabla p = \begin{bmatrix} -46 + 10x_1 + 4x_2 + (\gamma x_1)/|x_1| \\ -10 + 4x_1 + 10x_2 + (\gamma x_2)/|x_2| \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\nabla p = \begin{bmatrix} -46 + 10x_1 + \gamma \\ -10 + 4x_1 + \gamma \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Since the second index in ∇p corresponds to the derivative w.r.t. to x_2, we discard the constraint since we already showed in A that $x_2 = 0$.

$$\therefore x_1 = -0.1\gamma + 4.6$$

In [1]:

```
# IMPORT BOX #

using LinearAlgebra
using Plots
using JuMP, Ipopt
import GLPK

using Convex, SCS
```

In [2]:

```
# PART A #

A = [0 2; 1 0; 2 1];
y = [-2; 5; 9];

x_0_star = inv(transpose(A)*A)*transpose(A)*y
```

Out[2]:

```
2-element Array{Float64,1}:
  5.0
 -0.9999999999999998
```

In [6]:

```
# PART B: Solving for Optimal Point given λ #


#ARG:
γ = 0

function get_optimal_x(γ)
    n = 2

    # Create a (column vector) variable of size n x 1.
    x = Variable(n)

    # The problem is to minimize ||Ax - b||^2 subject to x >= 0
    # This can be done by: minimize(objective, constraints)
    problem = minimize(sumsquares(A * x - y) + γ*norm(x, 1))

    # Solve the problem by calling solve!
    solve!(problem, SCS.Optimizer)

    # Check the status of the problem
    problem.status # :Optimal, :Infeasible, :Unbounded etc.

    # Print the value for x
    println(round.(evaluate(x), digits=2))

    # Get the optimal value
    problem.optval
    return(evaluate(x))
end
```

Out[6]:

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

In [ ]:

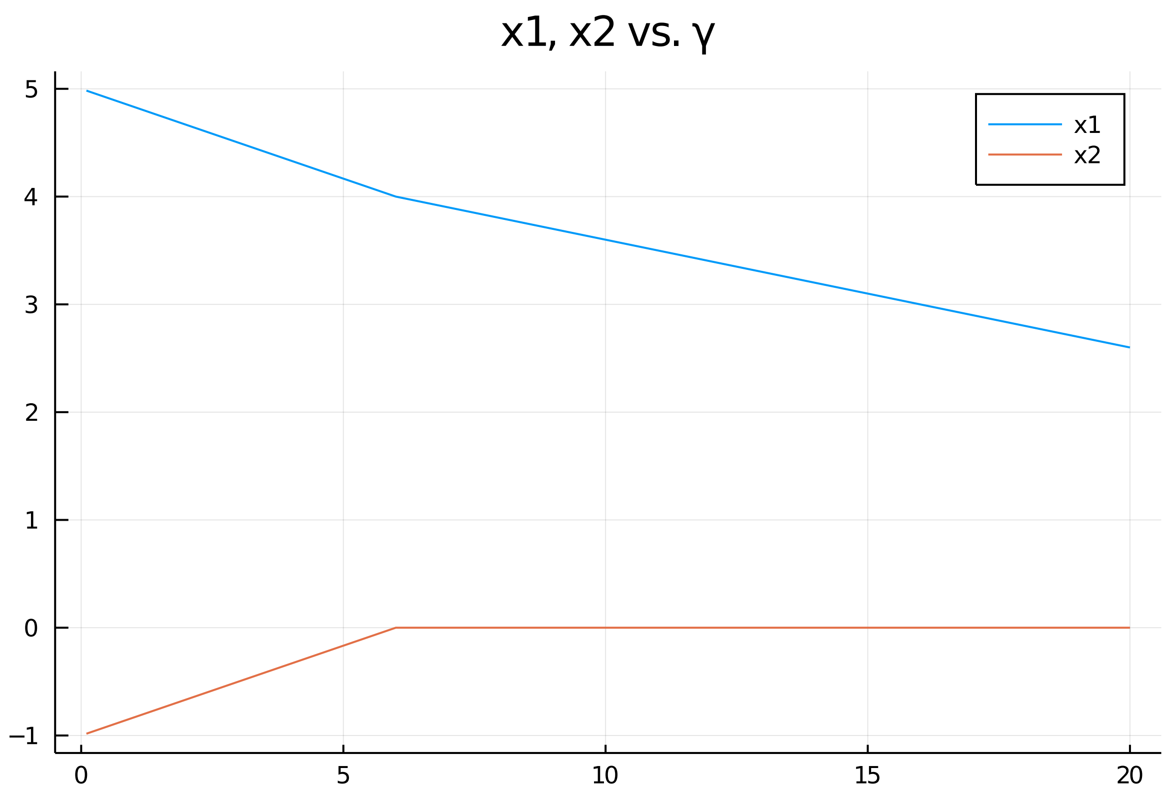
```
range_gamma = 20

max_i = 20*10

xmat = zeros(max_i, 2) # matrix to keep optimal x for various γ values.
```

```
gs = zeros(max_i)

for i = 1:max_i
    γ = i/10;
    gs[i] = γ;

    x = get_optimal_x(γ)

    xmat[i,:] = x
end
```

In [5]:

```
plot((1:max_i)/10, xmat, label = ["x1" "x2"])
title!("x1, x2 vs. γ")
```

Out[5]:



In [52]:

```
# PART C: See justification above #
```

In [53]:

```
# PART D: See justification above #
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: