Lab 4: House Elevation NPV Analysis

Jiayue Yin

Tue., Feb. 13

using CSV  
using DataFrames  
using DataFramesMeta  
using Distributions  
using Interpolations  
using Plots  
using StatsPlots  
using Unitful  
  
Plots.default(; margin=6Plots.mm)  
  
include("depthdamage.jl")

## pervious settings

# loading the data and the fuction before  
haz\_fl\_dept = CSV.read("data/haz\_fl\_dept.csv", DataFrame) # read in the file  
desc = "one story, Contents, fresh water, short duration"  
row = @rsubset(haz\_fl\_dept, :Description == desc)[1, :] # select the row I want  
dd = DepthDamageData(row) # extract the depth-damage data  
damage\_fn = get\_depth\_damage\_function(dd.depths, dd.damages) # get the depth-damage function  
elevation\_cost = get\_elevation\_cost\_function()

elevation\_cost (generic function with 1 method)

# set the origin value   
gauge\_dist = GeneralizedExtremeValue(5, 1, 0.1)   
offset = 7.5   
house\_dist = GeneralizedExtremeValue(gauge\_dist.μ - offset, gauge\_dist.σ, gauge\_dist.ξ)

GeneralizedExtremeValue{Float64}(μ=-2.5, σ=1.0, ξ=0.1)

# using the resteraunt data on lab3   
house\_area = 928u"ft^2"   
house\_value = 230000  
elevation = 1  
flood\_dist = GeneralizedExtremeValue(gauge\_dist.μ - (offset + elevation), gauge\_dist.σ, gauge\_dist.ξ)  
Δh = elevation \* 1u"ft"

1 ft

## NPV analysis

function single\_year\_cost\_benefit(flood\_dist, damage\_fn, elevation\_cost, house\_area, house\_value, Δh)   
 samples = rand(flood\_dist, 1000) .\* 1u"ft"  
 damages = damage\_fn.(samples)  
 expected\_damages\_pct = mean(damages)  
 c\_dmg = expected\_damages\_pct \* house\_value / 100  
 c\_constr = elevation\_cost.(Δh, house\_area)  
 return -c\_constr - c\_dmg  
end

single\_year\_cost\_benefit (generic function with 1 method)

function npv\_cost\_benefit(flood\_dist, damage\_fn, elevation\_cost, house\_area, house\_value, Δh, T, discount\_rate)  
 # calculate the costs and benefits for each year, and then discount  
 time = 0  
 npv = 0  
 for time in (1:T)  
 cost = single\_year\_cost\_benefit(flood\_dist, damage\_fn, elevation\_cost, house\_area, house\_value, Δh)  
 yearcost = cost \* (1-discount\_rate)^time  
 npv += yearcost  
 Δh=0u"ft"  
 end  
  
 return npv  
end

npv\_cost\_benefit (generic function with 1 method)

## One SOW, several actions

discount\_rate = 0.05  
T = 10  
# npvcost = npv\_cost\_benefit(flood\_dist, damage\_fn, elevation\_cost, house\_area, house\_value, Δh, T, discount\_rate)  
# print(npvcost)  
  
for elevation in range(0.,6.)  
 flood\_dist = GeneralizedExtremeValue(gauge\_dist.μ - (offset + elevation), gauge\_dist.σ, gauge\_dist.ξ)  
 Δh = elevation \* 1u"ft"  
 npvcost = npv\_cost\_benefit(flood\_dist, damage\_fn, elevation\_cost, house\_area, house\_value, Δh, T, discount\_rate)  
 println("While elevation = ",elevation," NPV = ", npvcost)  
end  
  
for Times in range(0,100,11)  
 flood\_dist = GeneralizedExtremeValue(gauge\_dist.μ - (offset + elevation), gauge\_dist.σ, gauge\_dist.ξ)  
 Δh = 5u"ft"  
 npvcost = npv\_cost\_benefit(flood\_dist, damage\_fn, elevation\_cost, house\_area, house\_value, Δh, Times, discount\_rate)  
 println("While Time = ",Times," NPV = ", npvcost)  
end

While elevation = 0.0 NPV = -70781.3745385032  
While elevation = 1.0 NPV = -123668.4312455626  
While elevation = 2.0 NPV = -110671.9091355255  
While elevation = 3.0 NPV = -100088.72184710945  
While elevation = 4.0 NPV = -97778.20671123911  
While elevation = 5.0 NPV = -95740.82494398886  
While elevation = 6.0 NPV = -95821.98928636005  
While Time = 0.0 NPV = 0  
While Time = 10.0 NPV = -129099.76817082023  
While Time = 20.0 NPV = -149528.85053468065  
While Time = 30.0 NPV = -163370.32762111427  
While Time = 40.0 NPV = -167051.38485564722  
While Time = 50.0 NPV = -171838.70202054476  
While Time = 60.0 NPV = -173080.75906432484  
While Time = 70.0 NPV = -176960.72523453407  
While Time = 80.0 NPV = -181697.96831393364  
While Time = 90.0 NPV = -182026.13592493755  
While Time = 100.0 NPV = -180950.97183792928

# Sensitivity Test

discount = Normal(0.5,2) /100  
rate\_samples = rand(discount, 200)  
elevation = 2  
Δh = elevation \* 1u"ft"  
npvs = npv\_cost\_benefit.(gauge\_dist, damage\_fn, elevation\_cost, house\_area, house\_value, Δh, T, rate\_samples)  
exp\_npv = mean(npvs)

-2.105280891764258e6

## Discussion

1. What do you notice about the NPV for different actions? Raising the height of a house has a significant impact. The NPV value has always remained negative, and the maximum NPV value appears when elevation is 5. As time increases, the value of NPV continues to decrease. There is no maximum NPV value appears while change time.
2. What do you notice about the sensitivity test? As the ratio fluctuates between 0.5, the average result is smaller than before.
3. What are some limitations of this analysis?
   * What things are missing from this analysis that you think are important?
   * How might they affect the results?
   * What are some ways you might address these limitations?

This result is a relatively crude result, and it does not explore specifically which types of costs have increased and which types of benefits have decreased? Is there a specific elevation threshold above which costs increase dramatically? There are many factors that are not taken into account, such as the specific type of climate disaster, the error of the loss function, the impact of inflation on the amount of loss, etc. I think a single model is not enough to solve this problem. Multiple models should be introduced and given weights to jointly solve this problem.