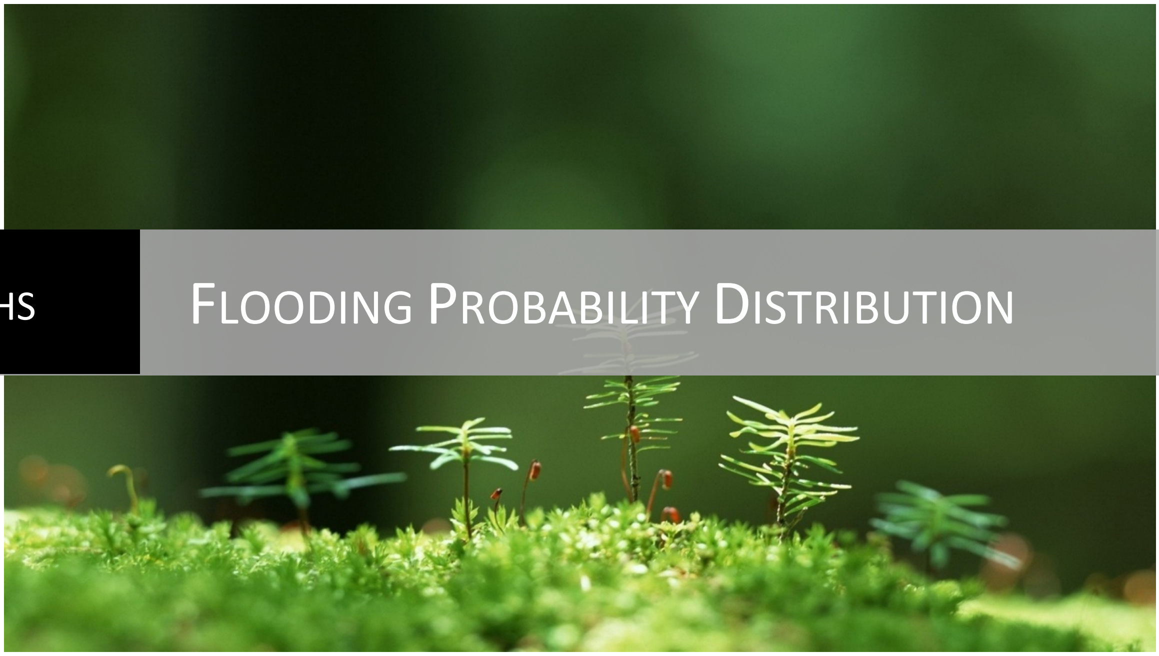
SCC Final Report



LCHS

Flooding Probability Distribution

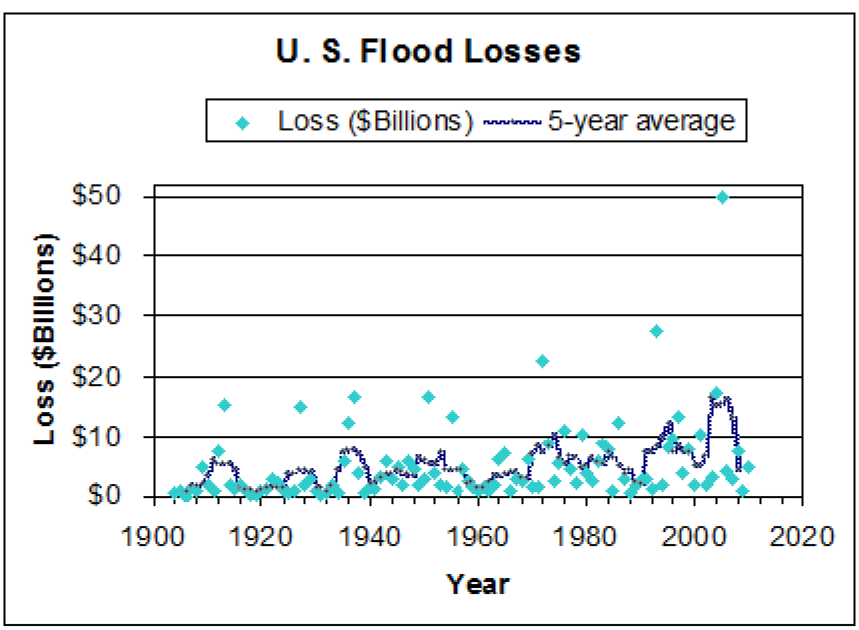
Team 59 | Kevin Lee, Andrew Lawendy, Dante Laroche

Table of Contents

1. Executive Summary

Currently, floods are the number one natural disaster in the United States. It has cost over 8 Billion USD in 2011, 4 Billion USD of damage in 2013 and it has claimed thousands of lives. Flooding is a growing issue all around the globe; last year Germany has spent 16 Billion USD alone on flooding. Floods usually occur when the ground cannot absorb water as quickly as it is falling from the sky, due to either heavy rain causing the ground to become saturated or from something preventing absorption, like concrete or asphalt.

The issue of flooding is not limited to coastal areas and locations with a source of water nearby; flooding may occur where ever it rains. Flooding can even occur in desert towns like El Paso, Texas which isn't too far away from home. In 2006, the desert town of El Paso, Texas received record rainfall which the city was not equipped to handle. The flood destroyed over 300 homes and cost the city over 100 USD million in damages. A visualization for the damage flooding has caused in the U.S. can be seen below.



The best protection against floods is avoidance. If we can conduct risk assessments in certain regions and bring awareness to a community about possible flooding in their area, they can be better prepared for it; thousands of lives and billions of dollars that would otherwise be lost may be saved. In our project, we have designed a model to give an approximate prediction of whether an area is at risk for flooding or not. This tool can help identify locations which may be affected by flooding. It also gives an estimate on the return period of the floods.

2. Problem Statement

Our project is based off of a mathematical model known as the Probability Distribution Theorem. We will use the equations related to probability distribution to help tackle flooding issues.

For the purposes of this project, we can only give a conservative estimation. We need a program that is easy enough to use and is not hard to customize in order to make it easy for the general population to use. This program also has a very user friendly interface allowing the user to manipulate the variables as the program is running which can then quickly return predictions.

We believe that we have accomplished this in a simple, and utilitarian fashion, with a program that allows the user to input the number of floods observed, the number of years observed, the scope of time they want to predict, and the number of floods that they want to predict. These features allow our model to be more useful than other similar online resources, which do not allow you to use your own data. This allows our simulation to be more modular and customizable as well.

As we found out first hand while looking for a program that worked similarly to what we were trying to accomplish we discovered that the interfaces were often very confusing, hard to use and/or they simply didn't allow you to input your own data. There is a need for a program that is vital for construction workers and business owners alike to know when floods are more likely to come otherwise they can be severely injured economically and physically.

3. Mathematical Model

We have determined that the most valuable information that the users can receive from the program are: the return interval for floods, the actual percent chance of flooding, and an estimated percentage of chance of floods from the inputted data. In order to give the user this information we used two different mathematical models. Our mathematical models converge to the Poisson Distribution, which is an expression used to find the percentage (probability) of a certain event happening, with a given number of observed events in a fixed interval of time. These mathematic models are the basis of our project:

3.1 Return Period Equation

The return period or “expected frequency” is a simple equation seen below:

$$\frac{1}{T} = \frac{M}{N}$$

$$where\ T\ is\ the\ return\ interval$$

$$where\ m\ is\ the\ number\ of\ recorded\ occurrecnces$$

$$where\ n\ is\ the\ number\ of\ years\ on\ record$$

"Return Period." Wikipedia. Wikimedia Foundation, 20 Dec. 2013. Web. 07 Feb. 2014.

This equation will give the user the “return period” or the amount of time between each flood. The return period equation is based off of empirical data. The return

Months	Precipitation/Did it flood?
January	29cm/no
February	31cm/no
March	32cm/no
April	29cm/no
May	33cm/yes
June	29cm/no
July	31cm/no
August	34cm/yes
September	32cm/no
October	30cm/no
November	31cm/no
December	34cm/yes

Lee, Soo Gyu. "Regression." 2014. Home, Las Cruces. 7 Feb. 2014. Lecture.

period gives us how often floods occur when a in an area. To further explain, here is a pre-defined data set shown below:

In the imaginary data set above whenever the rainfall exceeded 33cm, the area flooded. We can see how it only flooded 3 out of the 12 months in that particular year, so this data then gives us a return period of about once every 4 month. The probability of a flood occurring increases the closer you get to your return period and becomes even more likely after you pass your regular interval between floods.

3.2 Probability of Flooding

This next equation calculates the actual percentage chance of flooding. There are two parts to finding the percent chance of flood. The two equations are seen below.

$$p = \frac{1}{T} = \frac{M}{N+1}$$

$$= \binom{n}{r} \, x \, p^r \, x \, (1-p)^{(n-r)}$$

$$where\ p\ is\ the\ percent$$

$$where\ P\ is\ the\ probability\ of\ an\ event\ occuring$$

$$where\ T\ is\ the\ return\ interval$$

$$where\ m\ is\ the\ number\ of\ recorded\ occurrecnces$$

$$where\ N\ is\ the\ number\ of\ years\ on\ record$$

$$where\ r\ is\ the\ number\ of\ floods\ you\ want\ to\ predict$$

$$where\ n\ is\ the\ scope\ of\ years\ you\ want\ to\ predict$$

"Return Period." Wikipedia. Wikimedia Foundation, 20 Dec. 2013. Web. 07 Feb. 2014.

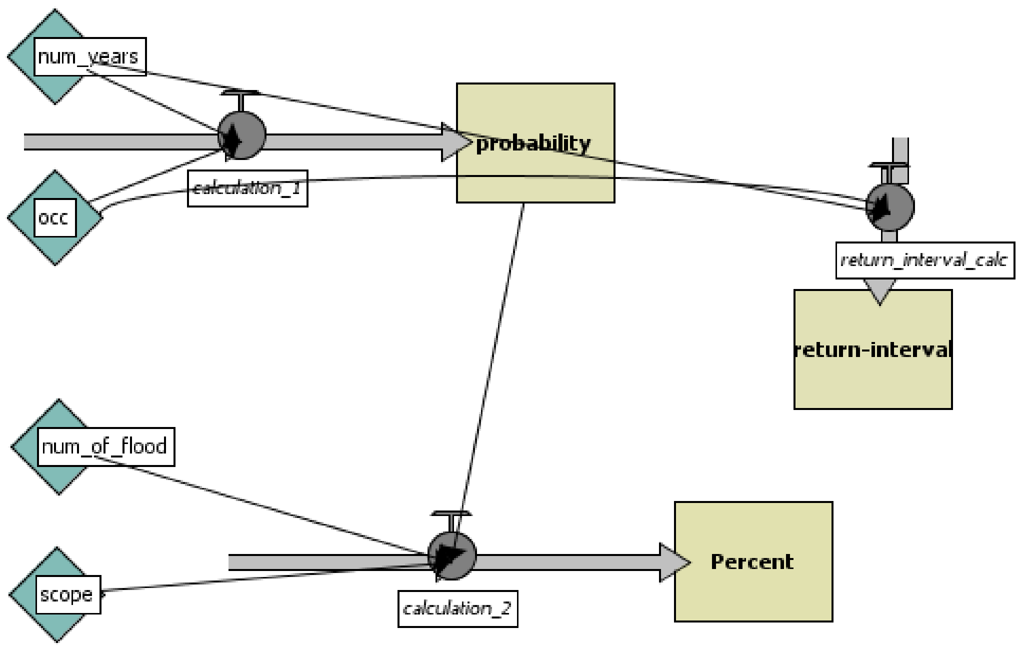
These equations form the backbone of our model, providing the outputs that our program needs. The outputs make sense of otherwise confusing data and give us relevant data that can be used to predict the likelihood of flooding in that particular area. This is the vital piece of information that people need when deciding to live in a certain area or whether or not they need flood insurance.

4. Computational Implementation

We have implemented the mathematical models into the agent based programming language; NetLogo. We have chosen this language because we feel it is more user friendly and therefore better fits the free software mentality. We also use specific features provided by NetLogo to make it easier to understand and run experiments with great ease. Specifically we utilized NetLogo's System Dynamics Modeler and Behavior Space.

System Dynamics Modeler is a comprehensive program inside NetLogo that allows you to connect VARIABLES visibly using links, which visually show what inputs are going into what operations clearly. You can then see the equation used by looking at the FLOW and you can see the final output in the stock. The

System Dynamics Modeler helps the user visually see what is going on in the program. An image of the System Dynamics Modeler is seen below.



Behavior space allows the user to run many experiments in a single run and it allows for greater orgainzation in experiments. The Behavior space also outputs the data needed in a nicely formatted data sheet.

4.1 How it Works

First the model will take in the user's input of:

Empirical values/data:

Observed occurrences (How many times has it flooded in your area?).

The number of years observed for those occurrences (The range of time you have observed those floods).

These numbers can be inputted via the first two sliders.

Hypothetical values/data:

The scope of the years you want to predict (i.e. how many floods will there be in x amount of years?).

How many floods you want to predict (If you want to calculate for just 1 flood then set the slider to 1, if you want to see the chance of there being 4 floods, change the slider accordingly).

In the System Dynamics Modeler we can see that first the VARIABLES num_years and occ take in the users inputs from the slider then they are divided to give you the return interval (the time between each flood). Then the FLOW calculation_1 takes the number of observed occurrences and divides it by the numbers of years observed plus one, which is then divided by 1 to give us the probability (frequency). Once we have the probability percentage we can find the actual probability of future floods. The VARIABLES num_of_flood and scope take in the user's hypothetical inputs from the sliders. Then the FLOW calculation_2 calculates the final percent chance of x amounts of flood in y amount of years. This is done by using the probability distribution theorem which is:

= (n / r) * p^r * (1-p)^(n-r)

where:

n is hypothetical number of years

r is hypothetical number of floods

p is probability

The simulation takes the four slider inputs and runs them through our equations for the percent chance and the return period of floods.

Once the simulation has finished, the monitors give a quick numeric reporter for the experiment and are accompanied by a color specific to the danger level. The colors that appear are blue, green, yellow, orange, and red in that order. These colors were picked because they resemble the forest service's fire danger signs, and work on the same principle. Green is low, blue is moderate, yellow is high, orange is very high, and red is extreme. The monitors are the direct return values of the equations from our model.

"US Forest Service - Caring for the Land and Serving People." *US Forest Service - Caring for the Land and Serving People.* N.p., n.d. Web. 01 Apr. 2014.

5. Simulations

We have run several simulations to test for the accuracy of the program. One experiment we ran was for Pleasant Hill Missouri. From the nation weather service website we obtained data that shows that the area has been flooded 11 times in the past 200 years. The probability of a flood in the coming year is 5.4726%. Our simulation matches the national weather service which rates the area at almost no flooding, so we know that our program is accurate.

6. Results

The program had a few bugs that we ran into, one of which was that the monitor would give us “0” for an output we then figured out that this error was occurring because the value that would have been in the monitor was a negative number which would haven’t have been useful anyway. We also found that our program doesn’t run properly when Occurrences to number of years exceeded 100% also it won’t work properly when the scope to number of floods is less than .001. When we let hypothetical floods get to large we got back an overflow error and the monitors printed out zero we discovered this happened because probability was raised to the number of hypothetical floods.

7. Conclusion

Our project was to implement a conservative mathematical model of a region's flooding chance and to create a prediction for the return period floods. The results of this model are consistent and accurate to real world data, which allows us to extrapolate the real world data forward in time to make predictions. It is also user friendly and allows us to change the data set to simulate any part of the world as long as we have data.

An effective, elegant and easy to use program that is still accurate is what we aimed to create. As you can see above in the Simulation Section you can see that the accuracy of our program is relatively high making its results reliable and trustworthy.

By using only three major equations we were able to give the user an abstracted understanding of what he is to expect in the upcoming month and the percent chance that a flood will occur in that month. Our program is only limited by the amount of data that is provided so you can look up the chance of flooding in an entire country or a small town like Mesilla, no other program that we have come across does this making our program unique in its own way.

8. Appendix/References

"Return Period." Wikipedia. Wikimedia Foundation, 15 Mar. 2014. Web. 01 Apr. 2014.

Florida disaster history (n.d.). In *Florida disaster*. Retrieved April 2, 2014, from http://floridadisaster.com/history.html

Historic flood events in the Missouri River basin (2010, August 11). In *National Weather Service*. Retrieved April 2, 2014, from http://www.crh.noaa.gov/mbrfc/?n=flood

"로지스틱 회귀분석 이렇게 하는게 맞나요 : 지식." 로지스틱 회귀분석 이렇게 하는게 맞나요 ?: 지식. Naver, n.d. Web. 07 Feb. 2014.

Lee, Soo Gyu. "Regression." 2014. Home, Las Cruces. 7 Feb. 2014. Lecture.

"Return Period." Wikipedia. Wikimedia Foundation, 20 Dec. 2013. Web. 07 Feb. 2014.

Redd, Nola Taylor. "Flood Facts, Types of Flooding, Floods in History." LiveScience. TechMedia Network, 11 Oct. 2012. Web. 08 Feb. 2014.

"AHPS Precipitation Analysis." AHPS Precipitation Analysis. U.S Government, 22 Oct. 2013. Web. 06 Feb. 2014.

Tai, Amos. Personal interview. 18 Nov. 2013

"US Forest Service - Caring for the Land and Serving People." *US Forest Service - Caring for the Land and Serving People.* N.p., n.d. Web. 01 Apr. 2014.

9. Source Code

```
to Reset
clear-all
end


to Calculate
system-dynamics-setup
visual
end


to visual
  ask patches
  [
  if probability <= .20
    [
      set pcolor  green
    ]
  if probability <= .40 and (probability > .20)
    [
      set pcolor  blue
    ]
  if probability <= .60 and (probability > .40)
    [
      set pcolor  yellow
```

```
    ]
  if probability <= .80 and (probability > .60)
    [
      set pcolor  orange
    ]
  if probability <= 1.00 and (probability > .80)
    [
      set pcolor red
    ]
  ]
  End
;; System dynamics model globals
globals [
  ;; stock values
  Percent
  probability
  return-interval
  ;; size of each step, see SYSTEM-DYNAMICS-GO
  dt
]


;; Initializes the system dynamics model.
;; Call this in your model's SETUP procedure.
to system-dynamics-setup
```

```
  reset-ticks
  set dt 1.0
  ;; initialize stock values
  set Percent calculation_2
  set probability calculation_1
  set return-interval return_interval_calc
end


;; Step through the system dynamics model by performing next iteration of Euler's method.
;; Call this in your model's GO procedure.
to system-dynamics-go


  ;; compute variable and flow values once per step
  let local-num_years num_years
  let local-occ occ
  let local-scope scope
  let local-num_of_flood num_of_flood
  let local-calculation_2 calculation_2
  let local-calculation_1 calculation_1
  let local-return_interval_calc return_interval_calc


  ;; update stock values
  ;; use temporary variables so order of computation doesn't affect result.
```

```
  let new-Percent ( Percent + local-calculation_2 )
  let new-probability ( probability + local-calculation_1 )
  let new-return-interval ( return-interval + local-return_interval_calc )
  set Percent new-Percent
  set probability new-probability
  set return-interval new-return-interval


  tick-advance dt
end


;; Report value of flow
to-report calculation_2
  report ( 100 * ((scope / num_of_flood) *  (probability)^(num_of_flood) * (1 -
probability)^(scope - num_of_flood))
  ) * dt
end


;; Report value of flow
to-report calculation_1
  report ( (occ / (1 + num_years))
  ) * dt
end


;; Report value of flow
```

```
to-report return_interval_calc
  report ( num_years / occ
  ) * dt
end

;; Report value of variable
to-report num_years
  report num-years-obs-slider
end

;; Report value of variable
to-report occ
  report observed-occurances-slider
end

;; Report value of variable
to-report scope
  report scope-slider
end

;; Report value of variable
to-report num_of_flood
  report number-flood-slider
end
```

```
;; Plot the current state of the system dynamics model's stocks
;; Call this procedure in your plot's update commands.
to system-dynamics-do-plot
  if plot-pen-exists? "Percent" [
    set-current-plot-pen "Percent"
    plotxy ticks Percent
  ]
  if plot-pen-exists? "probability" [
    set-current-plot-pen "probability"
    plotxy ticks probability
  ]
  if plot-pen-exists? "return-interval" [
    set-current-plot-pen "return-interval"
    plotxy ticks return-interval
  ]
end
```