The Tree of Well-Being

AnQi Wang, Shaowen Ren

Abstract

This project is a simplified simulation of an application to predict the future of personal well-being. In order to make the prediction, the project uses l-system to draws a tree of well-being. The system is evaluated, altered, and mutated with genetic algorithms that simulate the health behavior of a human being. The application predicts well-being of an individual. The prediction becomes better with each generation of genetic algorithm. The application's returns will be more accurate and similar to natural human life as more data and research are incorporated and calculated. The input for the application to calculate will be given by the user. Since this is an application that requires a mass amount of data. This project will prompt user for two simplified input category, social, and body. A tree will be drawn after input is taken with left tree being physical (body) health, and right tree being social health.

Key Words: l- system, genetic algorithm, evolving genotype, trees, life prediction using genetic algorithm,

Group Contributions

At first the group members had different main ideas, one wanted to work with cellular automata and genetic algorithms. But later found L-systems work smoothly with genetic algorithms. And Shaowen wanted to work with L-systems and genetic algorithms to evolve a particular image.

Main Idea

After discussions and analyzing the group came up with a focus on evolving L-system with genetic algorithms to simulate a real life procedure with the l-system working as trees like what we have done in class. Questions arose on what to evolve, what is simulate, and how to evolve the L-system in order to produce meaningful returns.

Coding

ShaoWen worked on the coding to get user input and calculate the input into the genetic algorithm. And to get the tree to draw (20 hrs.) AnQi coded part of the genetic algorithm and the evolving l-system functions. (> 15hrs) Both partners coded on the procedure to make calculation and prediction from the evolved system.

Analysis

In order for meaningful predictions AnQi made research on how much habits and performance effects the health of an individual, and ShaoWen incorporated these data into the code. We realized that gathering data for the well-being of an individual is quiet extensive, and will not be accurate without extensive and detailed data. So we group focused on several well-being effectors that are less controllable by an individual.

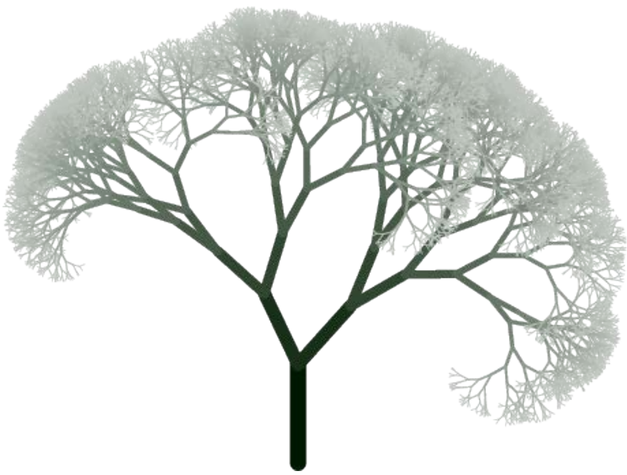
Report Writing

The abstract, what to evolve, how it is well-being is represented, calculation and predictions data and research, were written my AnQi

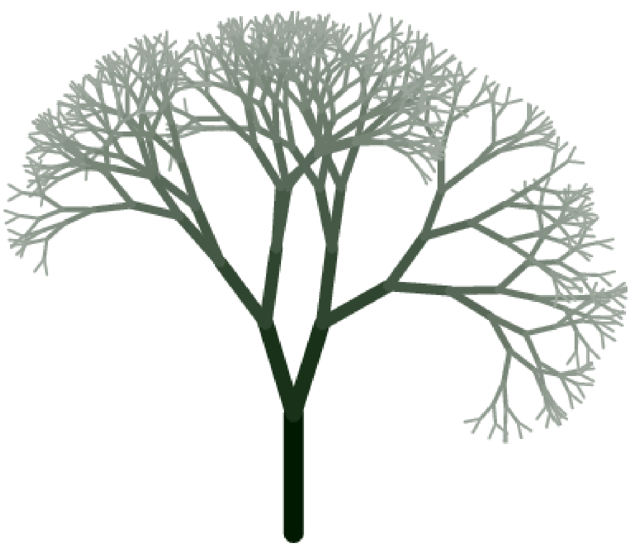
The evolving procedures and inputs, user interaction and application purpose and further development were written by ShaoWen

What to Evolve

For this project, l-systems that simulate trees are evolved. Since trees are forms of life that grows with a rule, we thought it best to use it to simulate the health behaviors of human beings. The trees provide a graphic representation of the predictions and returns for the application. In theory, if an individuals predicted future well-being is less optimistic the tree will be presented as a less lively, with less leafs, and branches. If an individual's predicted future well-being is more optimistic, the trees will be presented as more lush, dense, and blossom. For the limitation on time and capacity of this project, we could not get the tree to branch as needed when altering with the genetic algorithm. The tree offers a visual presentation of the predicted state of an individual's well-being. The visual is only relative with some calculations and does not guarantee the condition of the future for any individual. The accuracy of prediction is capable of increasing as more data and calculation are incorporated into the application.



The picture represents the project goal of a positive prediction.



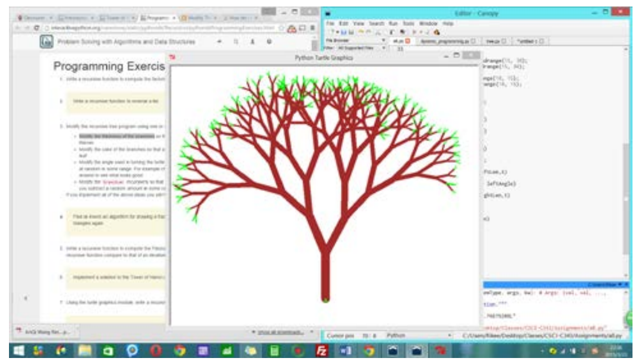
The picture represents the programs goal after genetic algorithm evolving the tree with user input with less positivity than the above.

Production Rule

L-system. At first our alphabet was {'F', '+', '-', '[', ']'}. Since the brackets for branching would not work when we get to the mutate and recombine in the genetic algorithms we changed the production rule to {F, +F, -F}.

There are some limits to python turtle graphics so, we made the production rule to simulate a more realist looking tree as possible. The branches are drew with brown, and the leaves drawn when the branch length gets to less than 20.

The mutation of production rule will be discussed in later sections.



This figure shows the tree that is drawn without genetic algorithm. Leafs are drawn when branch length < 20. Using turtle graphics in with python, color brown is used to represent branches and green are used to represent leaves. Angle is altered randomly.

Application Projective

Our initially tried to generate trees as above to try to represent different areas of individual health. The application offers a picture in mind of how the inputs and relative determining factors can affect the individual's health and conditions. Positive values for the factors can be thought as fertilizers that enhance the growths of the tree, in other words, it grows lush with leaves and branches. Negative values for factors can be thought of as soil that lack nutrition for growth, and may develop to a tree that is near demise if negativity is severe in other words, alter the angle of each branch more widely. Production rule will also be mutated more in the genetic algorithm which are used to evolve the tree or l-system and to make more accurate generation and prediction of individual health. The bounds for how much the tree is altered is determined by restriction on user input. The tree of the figure above stand for an average predicted health—non-mutated, non-altered.

User Inputs

The user inputs are integers and are restricted based on limit on time of the day. For example, the workout level is 0-10, with 0 being no work out at all, and 10 being work out regularly. Every user input is scaled on a range from 0 to 10 to ensure accuracy of the drawing of the tree. But this limits the accuracy of the prediction of real life health condition of the user.

Prompted User Input

The application prompts for five user inputs. These prompted user inputs are simplification of what is the determining factors of personal well-being.

The following are the user prompts:

"Please enter a number from 0 – 10 with 0 indicating no cigarette intake and 10 indicating very frequent cigarette intake: "
"Please enter a number from 0 – 10 with 0 indicating no alcohol intake and 10 indicating very frequent alcohol intake: "
"Please enter a number of hours you spend to workout in a week "
"Please enter the hours of sleep per day "
"Please enter 'yes' or 'no' for whether or not you are in a relationship.

The prompted user input are as follows:

Health factors	Possible input
Cigarette intake level	No bound
Alcohol intake level	No bound
Workout frequency	24 * 7
Sleep amount	24
In Relationship	'yes' or 'no'

Interpreting User Input

The above inputs are separated into two categories in the programming. The first four are calculated into physical health, which stimulates or tranquilizes the growth of the left subtree, and the last input is categorized into social health and is calculated to stimulate the growth of the right subtree. The figure above is a representation for social health that has a higher score than physical health.

Input Incorporation

The top four factors above in the table above are calculated in the function body_pressure (input). This function normalizes the scale of the factors with the top two subtracting from the scale making it more negative and the next two adding to the scale to make the scale more optimistic.

The last input which is a True or False question which is calculated using social_pressure(input) function. Which scales and normalizes the input accordingly.

These input can be better interpreted if there are precise data relations that exist between them, and if the functions incorporate these relationships, it would better evolve the l-system. Input relations will better portrait the wellness of the user but we could not find precise research on the relationships between each health factor and each other.

Evolving With Genetic Algorithms

The production rule for our tree of health l-system is altered using genetic algorithm. The genetic algorithms create genotypes with alphabets used to produce branches in a tree. The algorithm contains the below algorithms as with other genotype algorithms.

Init()

Evaluate (population)

Select (population, fitnesses)

Recombine (genotype_1, genotype 2, recombination rate)

Mutate (genotype, muatation_rate)

Algorithm Detail

The init() function generates random genotypes population with the size 20. for other functions to use

The evaluate(population) function evaluates that population generated by init() and takes each genotype and gives it a fitness score. For our algorithm, the fitness score is calculated in such a way:

- + and – are calculated for each genotype,
- + characters are calculated based on user's physical condition input using the function body_pressure(input)
- - characters are calculated based on user's social health condition using the function social_pressure(input)
- Fitness scores are calculated based on the balance of both areas of health. (the amount of + signs and – signs)

The evaluate function returns the fitness score for each genotype as a list.

The select(population, fitnesses) function takes the population from init() and the fitness scores form evaluate(population) and sets a random variable for comparison and returns two genotypes who's total fitness scores exceed the variable. This function returns two genotypes.

Recombine(genotype_1, genotype_2, recombination_rate) use the two genotypes generated from select(population, fitnesses) to recombine using the two genotypes with a recombination rate which is the social pressure/ total pressure

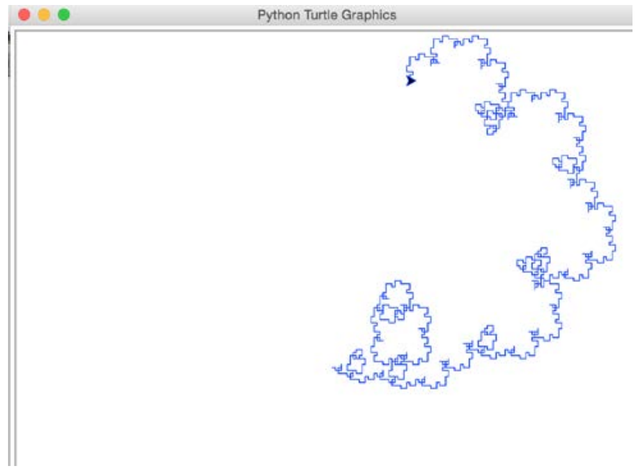
Mutate(genotype, mutation_rate)

To decide whether to mutate a genotype, we set the mutation rate to body pressure / total_pressure, since we think physical factors dominate in mutation in the real world.

F, +F, -F

Analyzing Results

After altering the l-system with our genetic algorithm. The following results show.



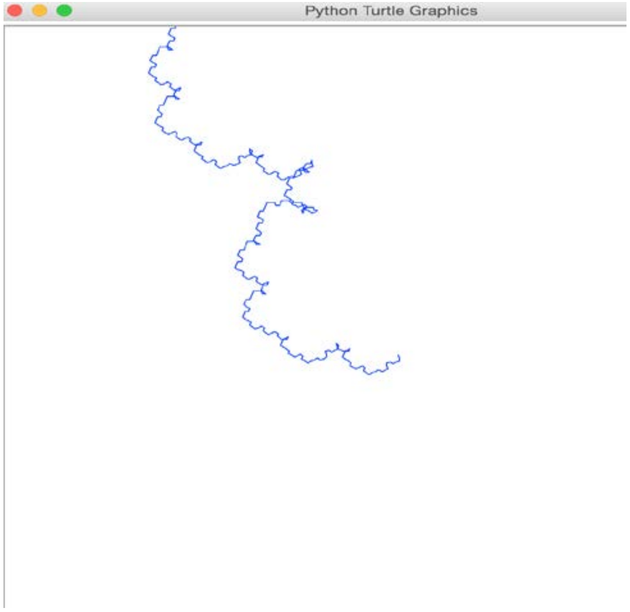
The above picture represents the first generation of our genetic algorithm, since our genetic algorithm's fitness score is based on the turns that the genotypes generally takes.

The input for this result is

Health factors	input
Cigarette intake level	0
Alcohol intake level	0
Workout frequency	2
Sleep amount	7
In Relationship	no

As one can see in the below image, the second generation of genotypes calculates the health conditions of the user more accurately.

The previous image has shown more curvature than the image below. This is due to more + and – signs in the genotypes. The second generation will return results with more precision after evolution with and selection.



Conclusion Project further development

With more time and research the project can be further developed with better accuracy and graphics. We believe this to be a meaningful project that can predict the health behaviors of human beings. More data and logic can be added both to user input and into the calculations. Further research can be done on data relations which will give a more precise outcome and return. The application could also include other factors than the body and social health to represent a more realistic individual health. For example, mental and emotional health factors could be added to the user input to better predict the health conditions for users. We believe that with further development, an application like this can be used for personal health tracking, health insurance calculations and needs and etc.

Reference

http://www.informatics.indiana.edu/rocha/i-bic/lab4/

http://www.informatics.indiana.edu/rocha/i-bic/pdfs/i485_lecture6_slides.pdf

http://www.informatics.indiana.edu/rocha/i-bic/pdfs/i485_lecture19_slides.pdf