HAIMLC701 AI & ML in Healthcare

2.0		**AI, ML, Deep Learning and Data Mining Methods for Healthcare**	10
	2.1	Knowledge discovery and Data Mining, ML, Multi classifier Decision Fusion, Ensemble Learning, Meta-Learning and other Abstract Methods.	
	2.2	Evolutionary Algorithms, Illustrative Medical Application-Multiagent Infectious Disease Propagation and Outbreak Prediction, Automated Amblyopia Screening System etc.	
	2.3	Computational Intelligence Techniques, Deep Learning, Unsupervised learning, dimensionality reduction algorithms.	

CO Mapped: CO2-Apply advanced AI and Computational Intelligence techniques for Healthcare Problems (L3)

Textbook to refer:

1. Arjun Panesar, "Machine Learning and AI for Healthcare”, A Press.
2. Arvin Agah, "Medical applications of Artificial Systems ", CRC Press

Computational Intelligence

- Theory, design, application and development of biologically and linguistically motivated computational paradigms

- Refers to the ability of a computer to learn a specific task from data or experimental observation

- Main pillars:
  - Neural Networks, Fuzzy Systems and Evolutionary Computation

- Set of nature-inspired computational methodologies and approaches to address complex real-world problems to which mathematical or traditional modelling can be useless for a few reasons
- Processes might be too complex for mathematical reasoning, it might contain some uncertainties during the process, or the process might simply be stochastic in nature

Evolutionary computation

- Involves the study of the foundations and the applications of computational techniques based on the principles of natural evolution
- Can be viewed either as search methods, or as optimization techniques
- Three basic mechanisms drive natural evolution: reproduction, mutation, and selection
- These mechanisms act ultimately on the chromosomes containing the genetic information of the individual (the genotype), rather than on the individual itself (the phenotype)
- The degree of adaptation of each individual (i.e. candidate solution) to its environment is expressed by an adequacy measure known as the fitness function

Evolutionary Algorithms (EA)

- In computational intelligence, an evolutionary algorithm is a subset of evolutionary computation, a generic population-based metaheuristic optimization algorithm
- An EA uses mechanisms inspired by biological evolution, such as reproduction, mutation, recombination, and selection
- 1948, Turing: "genetical or evolutionary search"
- 1962, Bremermann: optimization through evolution
- 1964, Rechenberg: evolution strategies
- 1965, L. Fogel, Owens and Walsh: evolutionary programming
- 1975, Holland: genetic algorithms
- 1992, Koza: genetic programming

Evolutionary Algorithms

- More precisely, evolutionary algorithms maintain a population of structures that evolve according to rules of selection and other operators, such as recombination and mutation
- Each individual in the population receives a measure of its fitness in the environment
- Selection focuses attention on high fitness individuals, thus exploiting the available fitness information

Evolutionary algorithm

- Recombination and mutation perturb those individuals, providing general heuristics for exploration
- Although simplistic from a biologist's viewpoint, these algorithms are sufficiently complex to provide robust and powerful adaptive search mechanisms
- A population of individual structures is initialized and then evolved from generation to generation by repeated applications of evaluation, selection, recombination, and mutation
- The population size N is generally constant in an evolutionary algorithm

General structure of Evolutionary Algorithms (EA)

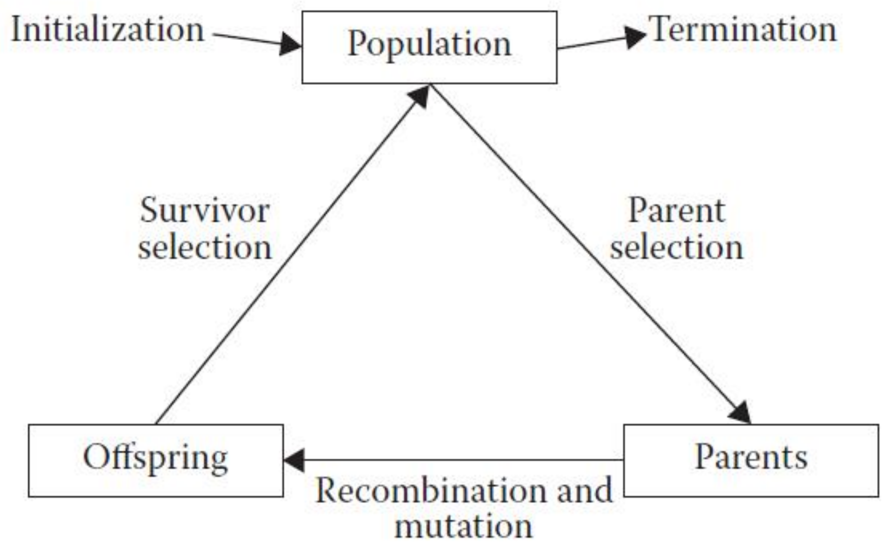


FIGURE 3.4
Iterative evolutionary process by which a population is created and evolved to arrive at a set of candidate solutions that maximize a fitness function.

Sample EA in pseudo-code

```
BEGIN
        INITIALIZE Population with random Candidate solutions
        EVALUATE each Candidate;
REPEAT WHILE (TERMINATION CONDITION is not satisfied)
        SELECT Parents;
        RECOMBINE pairs of Parents;
        MUTATE resulting Offspring;
        EVALUATE new Candidates;
        SELECT Individuals for the next Generation;
END
```

Implementation

- Example of a generic single-objective genetic algorithm
- Step 1:
  - Generate the initial population of individuals randomly. (First generation)
- Step 2:
  - Repeat the following regeneration steps until termination:
    1. Evaluate the fitness of each individual in the population (time limit, sufficient fitness achieved, etc.)
    2. Select the fittest individuals for reproduction. (Parents)
    3. Breed new individuals through crossover and mutation operations to give birth to offspring.
    4. Replace the least-fit individuals of the population with new individuals.

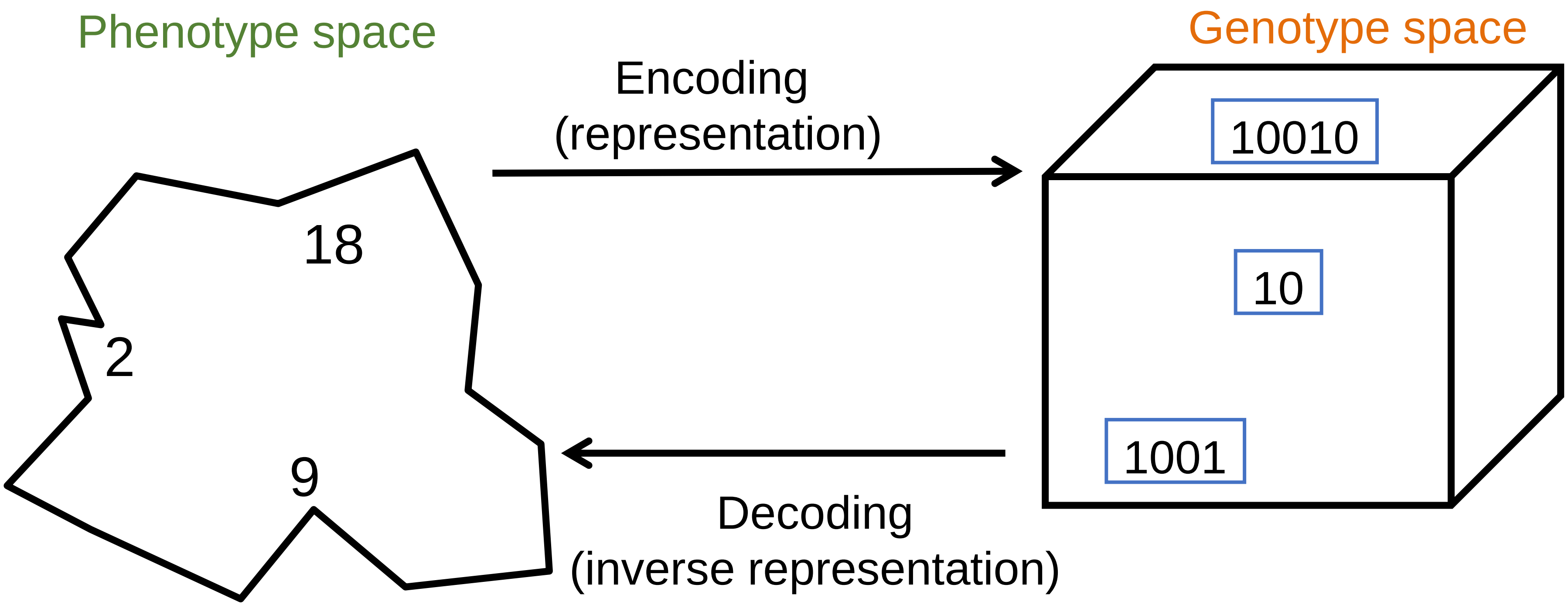
Main EA components: Representation

- Role: provides code for candidate solutions that can be manipulated by variation operators
- Leads to two levels of existence
  - phenotype: object in original problem context, the outside
  - genotype: code to denote that object, the inside (chromosome, “digital DNA”)
- Implies two mappings:
  - Encoding : phenotype=> genotype (not necessarily one to one)
  - Decoding : genotype=> phenotype (must be one to one)
- Chromosomes contain genes, which are in (usually fixed) positions called loci (sing. locus) and have a value (allele)

Each chromosome has two representations.
genotype: The set of genes representing the chromosome.
phenotype: The actual physical representation of the chromosome.

Main EA components: Representation

Example: represent integer values by their binary code



In order to find the global optimum, every feasible solution must be represented in genotype space

Main EA components: Evaluation (fitness) function

- Role:
  - Represents the task to solve, the requirements to adapt to (can be seen as “the environment”)
  - Enables selection (provides basis for comparison)
- A.k.a. *quality* function or *objective* function
- Assigns a single real-valued fitness to each phenotype which forms the basis for selection
  - So the more discrimination (different values) the better
- Typically we talk about fitness being maximised
  - Some problems may be best posed as minimisation problems, but conversion is trivial

Main EA components: Population

- Role: holds the candidate solutions of the problem as individuals (genotypes)
- Formally, a population is a multiset of individuals, i.e. repetitions are possible
- Population is the basic unit of evolution, i.e., the population is evolving, not the individuals
- Selection operators act on population level
- Variation operators act on individual level

Main EA components:Selection mechanism

Role:

- Identifies individuals
  - to become parents
  - to survive
- Pushes population towards higher fitness
- Usually probabilistic
  - high quality solutions more likely to be selected than low quality but not guaranteed

Mutation Operators

We might have one or more mutation operators for our representation:

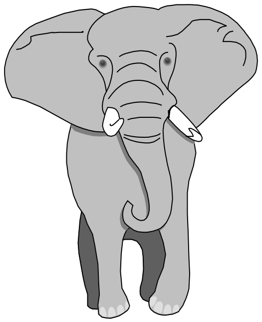
- At least one mutation operator should allow every part of the search space to be reached
- The size of mutation is important and should be controllable
- Mutation should produce valid chromosomes

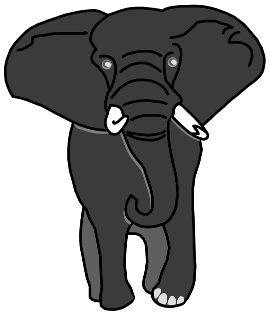
Example: Mutation for Discrete Representation

before 1 1 1 1 1 1 1

after 1 1 1 0 1 1 1

mutated gene

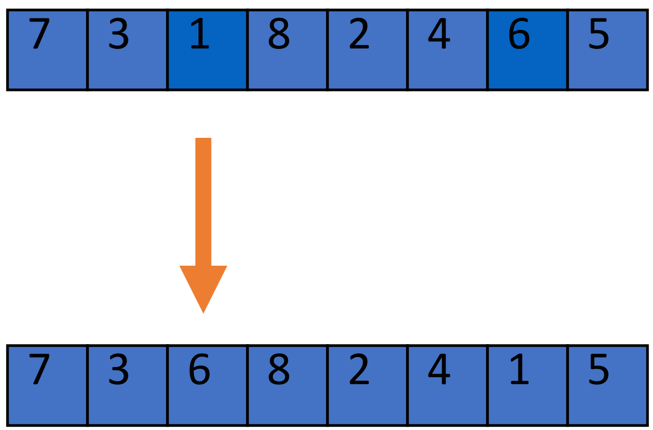




Mutation usually happens with probability p_m for each gene

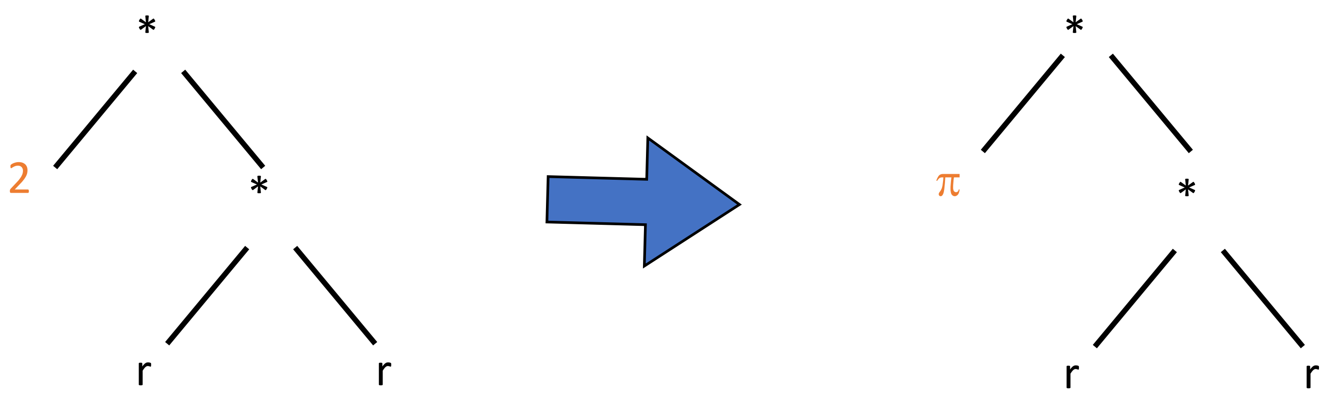
Example: Mutation for order based representation (Swap)

Randomly select two different genes and swap them.



Example: Mutation for tree based representation

Single point mutation selects one node and replaces it with a similar one.



Genetic Algorithms

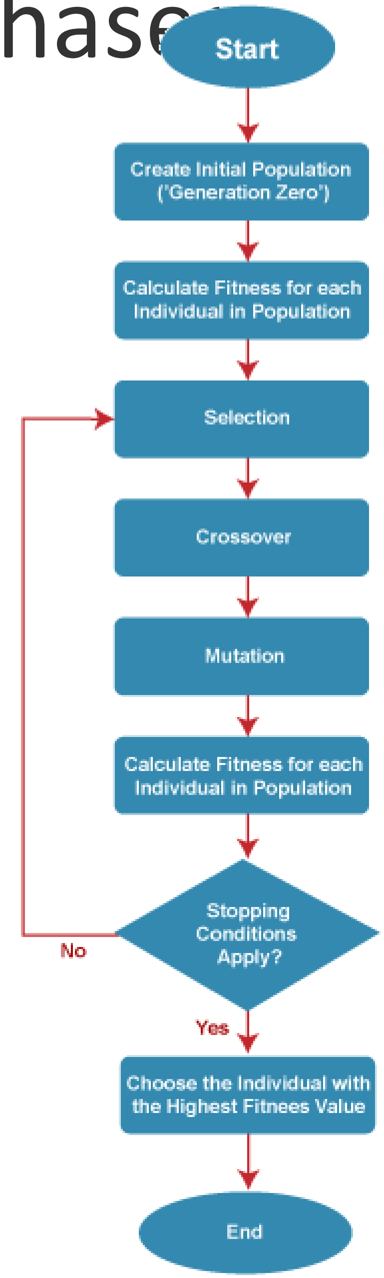
- Widely used in different real-world applications
- **Designing electronic circuits, code-breaking, image processing, and artificial creativity**
- It is a subset of evolutionary algorithms, which is used in computing.
- Uses genetic and natural selection concepts to solve optimization problems
- ***A genetic algorithm is an adaptive heuristic search algorithm inspired by "Darwin's theory of evolution in Nature*** used to solve optimization problems in machine learning.
- It helps solve complex problems that would take a long time to solve

Genetic Algorithms-Terminologies

- **Population:** Population is the subset of all possible or probable solutions, which can solve the given problem.
- **Chromosomes:** A chromosome is one of the solutions in the population for the given problem, and the collection of gene generate a chromosome.
- **Gene:** A chromosome is divided into a different gene, or it is an element of the chromosome.
- **Allele:** Allele is the value provided to the gene within a particular chromosome.
- **Fitness Function:** The fitness function is used to determine the individual's fitness level in the population. It means the ability of an individual to compete with other individuals. In every iteration, individuals are evaluated based on their fitness function.
- **Genetic Operators:** In a genetic algorithm, the best individual mate to regenerate offspring better than parents. Here genetic operators play a role in changing the genetic composition of the next generation.
- **Selection**

Genetic Algorithm-Five phases

- **Initialization**
- **Fitness Assignment**
- **Selection**
- **Reproduction**
- **Termination**



Numerical

Consider the problem of maximizing the function,

$$f(x) = x^2$$

Where x is permitted to vary between 0 to 31.

(i) 0(00000) and 31(11111) code x into finite length string

(ii) Select initial population at random (size 4)

(iii) Calculate fitness value for all strings

(iv) probability of selection by:

$$Prob_i = \frac{f(x)_i}{\sum_{i=1}^{n} f(x)_i},$$

Fitness Assignment

String No.	Initial population	X Value	Fitness value $f(x) = x^2$	Prob. $Prob_i = \frac{f(x)_i}{\sum_{i=1}^{n} f(x)_i}$	%age Prob.	Expected Count $\frac{f(x)_i}{(Avg\,f(x))_i}$	Actual Count
1.	01100	12	144	0.1247	12.47%	0.4987	1
2.	11001	25	625	0.5411	54.11%	2.1645	2
3.	00101	5	25	0.0216	2.16%	0.0866	0
4.	10011	19	361	0.3126	31.26%	1.2502	1
Sum			1155	1.0000	100%	4.0000	
Avg.			288.75	0.2500	25%	1.0000	
Max.			625	0.5411	54.11%	2.1645	

Cross over

String No.	Mating Pool	Crossover point	Offspring after crossover	X value	Fitness value
1.	01100	4	01101	13	169
2.	11001	4	11000	24	576
3.	11001	3	11011	27	729
4.	10011	3	10001	17	289
Sum Avg. Max.					1763 440.75 729

Mutation

String No.	Offspring After crossover	Mutation chromosomes	Offspring after mutation	X value	Fitness value
1.	01101	10000	11101	29	841
2.	11000	00000	11000	24	576
3.	11011	00000	11011	27	729
4.	10001	00100	10101	20	400
Sum **Avg.** **Max.**					2546 636.5 841

- 625→after crossover 729→after mutation→841 better solution

Illustrative Medical Applications

1. Multiagent disease propagation and outbreak prediction
2. Automated Amblyopia screening system

Multiagent Infectious Disease Propagation and Outbreak Prediction

- Anticipating, predicting, and monitoring disease outbreaks can assist public health officials in their response efforts- "Disease Propagation Modeling"

  (1) to predict when and where an outbreak will occur;

  (2) should an outbreak occur in a certain location, to predict how it will spread; and

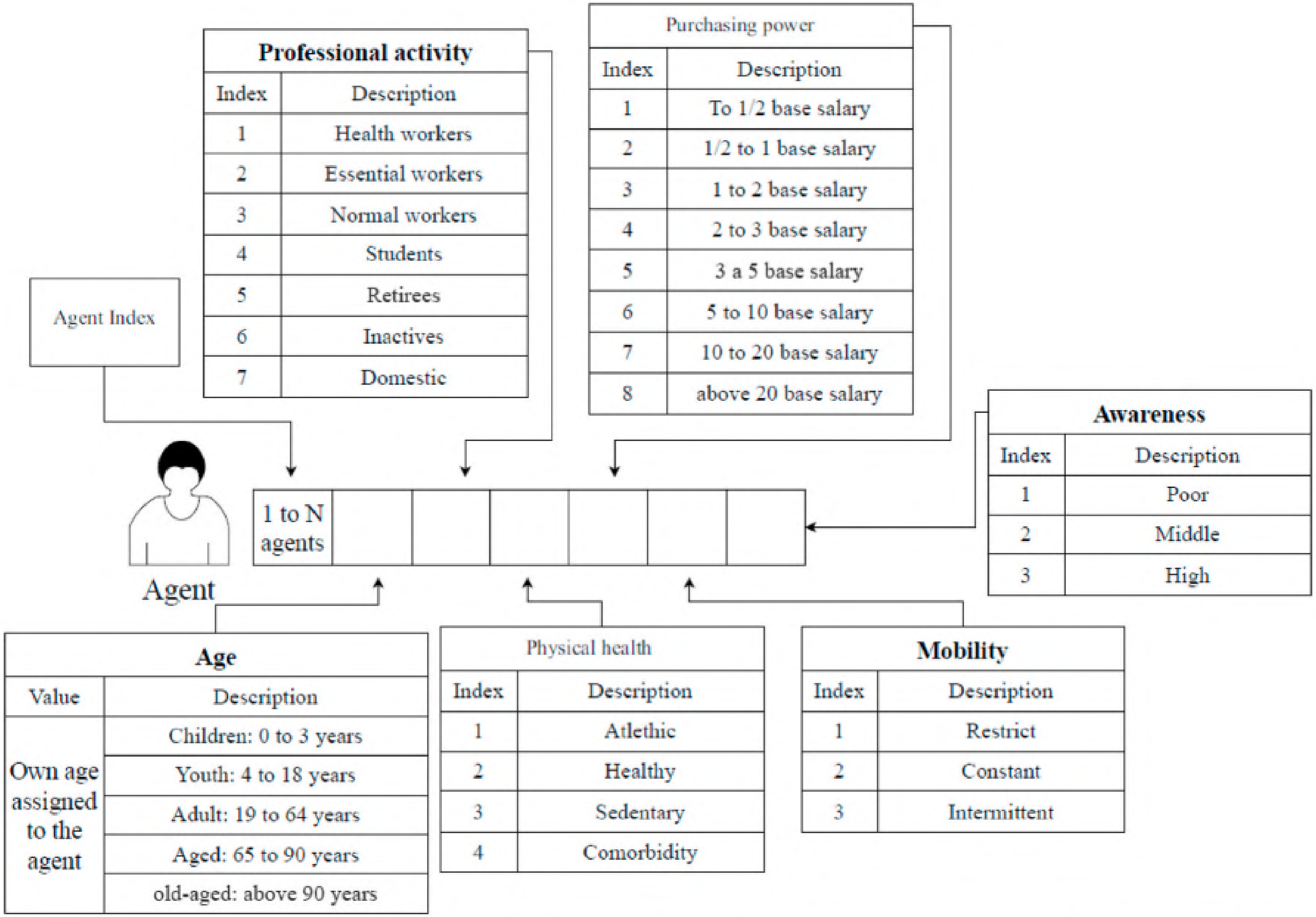
  (3) to monitor an outbreak as it is occurring
- Historical models based on differential equations to predict disease spread
  - the number of people who are susceptible to contracting the disease
  - are infected/infectious with the disease, and have recovered from the disease
  - However, they typically do not describe how the disease will spread spatially
- Agent-based modeling, which is based on the multiagent system paradigm
  - ➢ Rather than consider broad groups of people, this technique represents people individually and attempts to characterize their behavioral patterns.

Eg. Multi-agent simulation model for the evaluation of COVID-19 transmission

- An agent-based model to analyze the spread processes of the COVID-19 epidemics in open regions and based on hypothetical social scenarios of viral transmissibility
- achieved by modeling an individual as an agent with a wide range of features (health condition, purchasing power, awareness, mobility, professional activity, age, and gender)
- Simulation results show that it can be applied to support decision-makers to better understand the epidemic spread and the actions that can be taken against the pandemic

Multi-agent simulation model for the evaluation of COVID-19 transmission

- A multi-agent system is an organization of autonomous agents interacting with each other within a shared environment
- model is able to represent individuals' heterogeneity, environmental diversity and social interaction
- An agent has several features to express age, comorbidities, disease presentation (asymptomatic, mild or severe) among others
- Model incorporates aspects of social dynamics through simple rules based on statistical principles and probability analysis



Professional activity

Index	Description
1	Health workers
2	Essential workers
3	Normal workers
4	Students
5	Retirees
6	Inactives
7	Domestic

Purchasing power

Index	Description
1	To 1/2 base salary
2	1/2 to 1 base salary
3	1 to 2 base salary
4	2 to 3 base salary
5	3 a 5 base salary
6	5 to 10 base salary
7	10 to 20 base salary
8	above 20 base salary

Awareness

Index	Description
1	Poor
2	Middle
3	High

Age

Value	Description
Own age assigned to the agent	Children: 0 to 3 years
	Youth: 4 to 18 years
	Adult: 19 to 64 years
	Aged: 65 to 90 years
	old-aged: above 90 years

Physical health

Index	Description
1	Atlethic
2	Healthy
3	Sedentary
4	Comorbidity

Mobility

Index	Description
1	Restrict
2	Constant
3	Intermittent

Fig. 2. Classification of the attributes according to their respective aspects for the elaboration of the matrix that stores the agents' attributes in the model.

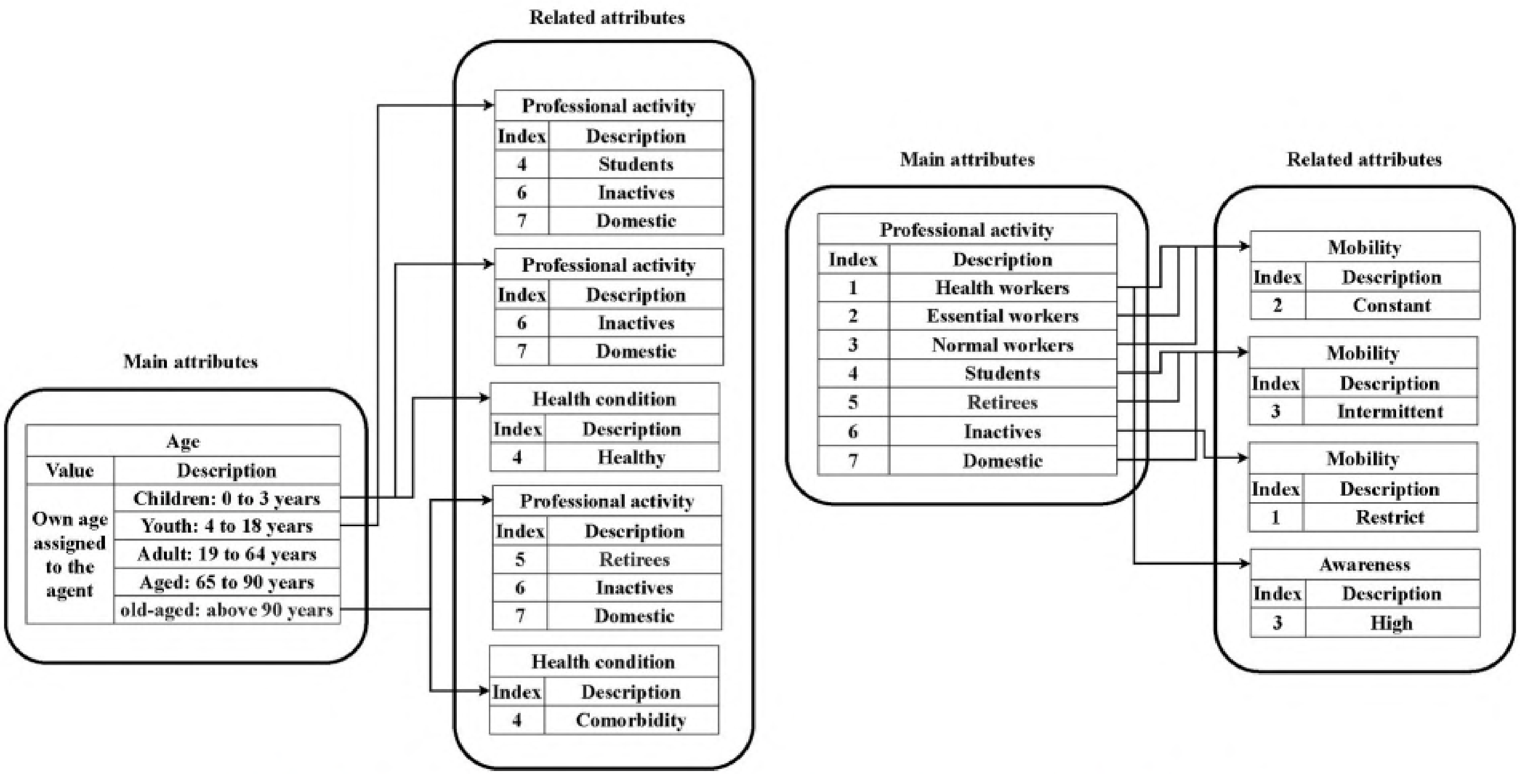


Fig. 3. Attributes and hypotheses related to the agents' age and professional activity classification.

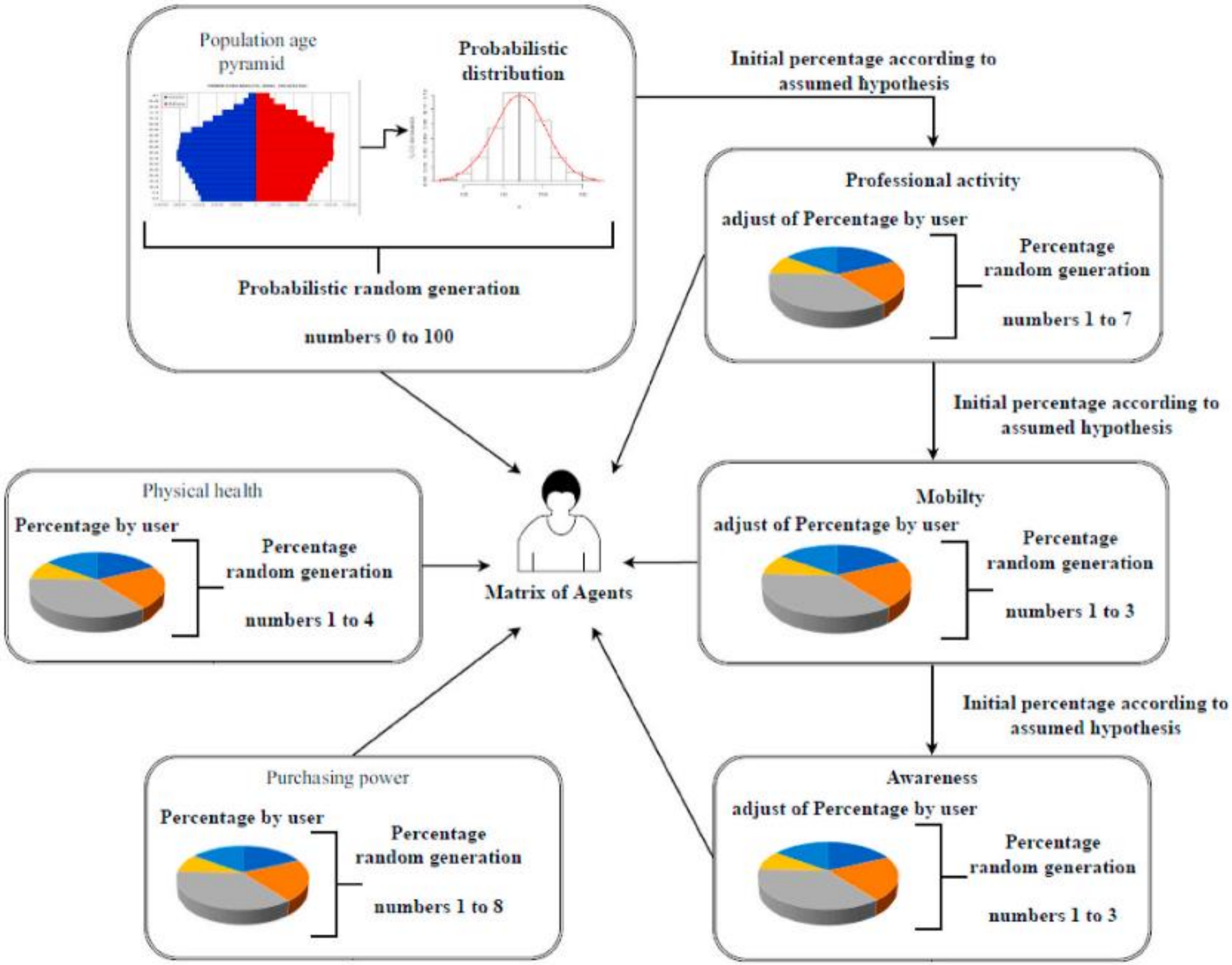


Fig. 4. Attribute classification process and agent attributes matrix elaboration.

Table 1
Scores assigned to each aspect of agents' attributes.

Professional activity		
Index	Description	Weigth
1	Health workers	75.39
2	Essential workers	62.98
3	Normal workers	59.13
4	Students	58.71
5	Retirees	54.12
6	Inactives	52.07
7	Domestic	52.07

Purchasing power		
Index	Description*	Weigth
1	up to 1/2	60
2	1/2 to 1	55
3	1 to 2	50
4	2 to 3	45
5	3 a 5	40
6	5 to 10	35
7	10 to 20	32
8	above 20	30

Physical health		
Index	Description	Weigth
1	Atlethic	30
2	Healthy	25
3	Sedentary	45
4	Comorbidity	50

Mobility		
Index	Description	Weigth
1	Restrict	20
2	Constant	60
3	Intermittent	40

Awareness		
Index	Description	Weigth
1	Poor	60
2	Middle	40
3	High	20

* using minimum wage as base

Automated Amblyopia screening system

- Amblyopia, commonly referred to as lazy eye, is a neurological vision disorder that studies show affects 2%–5% of the population.
- common childhood condition that can affect one or both eyes when the brain and eye don't work together properly, and the part of the brain that receives images from the affected eye doesn't develop fully

Causes

- nearsightedness, farsightedness, or astigmatism, especially if it's greater in one eye , cataracts, crossed eyes, when one or both eyes wander in, out, up, or down
- Nutritional amblyopia caused by deficiencies of one or more B complex vitamins, such as thiamin, niacin, vitamin B12, or folate

Automated Vision Defect Detection Supported Deep Convolutional Neural Networks

- Deep convolutional neural networks are used for automated amblyopia detection on tele amblyopia dataset.
- The proposed algorithm comprises of 2 phases
  - In first phase, the Enhanced Firefly Algorithm (EFA) is used to segment the eye region
  - In second phase, a DCNN is designed and trained to classify the segmented eye areas as amblyopia or normal
- Firefly algorithm
  - Inspired by flashing behavior of fireflies
  - Attracted towards each other
  - Attractiveness directly proportional to brightness
  - Less brighter gets attracted towards brighter
  - Attractiveness decrease as distance increases
  - If brightness both are same, fireflies move randomly
  - New solutions are generated by random walk and attraction

Firefly Algorithm

- 6 steps
  - Initialize parameters
  - Generate population of n fireflies
  - Calculate fitness value for each firefly(light intensity value)
  - Check if(t:=1 to Maxt)
  - Update position and light intensity of each firefly
  - Report the best solution

Paper Title: Applied Artificial Intelligence Techniques for Identifying the Lazy Eye Vision Disorder

- use artificial intelligence techniques to automatically identify children who are at risk for developing the amblyopic condition and should therefore be referred to a specialist, i.e., pediatric ophthalmologist.
- Three techniques, namely, decision tree learning, random forest, and artificial neural network, are studied in this paper
- The features used by the techniques are extracted from images of patient eyes and are based on the color information.
- The efficacy of pixel color data is investigated with respect to the measurement of the rate of change of the color in the iris and pupil, i.e., color slope features.
- After the individual eye images are available and cropped, the pixel values are to be extracted from different angles off the center of the eye.
- During phase one of the project these angles are the 0, 45, and 135 degrees on the center of the eye and across the iris and the pupil
- Phases two and three pixel for color extraction