Population Generation from Statistics Using Genetic Algorithms with MIST + INSPYRED

Jacob Barhak
Freelancer
Austin, Texas
Jacob.barhak@gmail.com

Aaron Garrett
Jacksonville State University
Jacksonville, Alabama
agarrett@jsu.edu

ABSTRACT

Clinical trial population information is typically restricted and individual data is not public. However, clinical trial results and population statistics are regularly published. It is possible to reconstruct mock individual data populations from these statistics to support disease modeling and better understand the population characteristics. This can help in both planning and analysis of trial results on a larger information scope involving multiple clinical trials.

A fairly simple example of generating an individual population from aggregate statistics is as follows: generate 1000 individuals such that their mean age would be 61 with SD of 8.2 and mean age at diagnosis of diabetes would be 53. Even this simple example has constraints such as age at diagnosis of diabetes should be lower than the individual age which will cause a skewed distribution. Reconstructing a mock population that matches clinical trial statistics is more complex and involves multiple objectives and interactions between statistics.

This work improves the Monte Carlo abilities of the MIcro Simulation Tool (MIST) to generate populations from statistics by introducing genetic algorithms supported by the INSPYRED software package. The genetic algorithm improves the accuracy of the reconstructed population and better handles skewed distributions and constraints.

MIST and INSPYRED are both free software available under GPL license and can be downloaded through these links:
https://github.com/Jacob-Barhak/MIST
https://github.com/inspyred/inspyred

ABOUT THE AUTHORS

Jacob Barhak is currently a freelancer specializing in chronic disease modeling with emphasis on using Computational Technological solutions. The Reference Model for disease progression was self developed by Dr. Barhak as a freelancer. Previously Dr. Barhak headed the Michigan Model for Diabetes computing team 2006-2012. A major part of this position involved developing the software environment and the Michigan Model for Diabetes. Dr. Barhak has diverse international background in engineering and computing science. For additional information please visit http://sites.google.com/site/jacobbarhak/

Aaron Garrett is an Assistant Professor in the Department of Mathematical Computing and Information Sciences, Jacksonville State University in Alabama. His interests include evolutionary computation and machine learning. He is the author of INSPYRED, a software library that includes biologically-inspired computation and encompasses a broad range of algorithms including evolutionary computation, swarm intelligence, and neural networks. For additional information please visit http://mcis.jsu.edu/faculty/agarrett/

Population Generation from Statistics Using Genetic Algorithms with MIST + INSPYRED

Jacob Barhak
Freelancer
Austin, Texas
Jacob.barhak@gmail.com

Aaron Garrett
Jacksonville State University
Jacksonville, Alabama
agarrett@jsu.edu

INTRODUCTION

Clinical Trials and observational studies follow a population of people over a period of time and record medical outcomes. While individual data is available to the organization conducting the trial, it is regularly restricted and not published. Nevertheless, the summary data of the trial is typically published and is available. Note that this summary data is, from many aspects, more reliable than data collected from a single individual just due to sample size. For example, for a single person we may know when they had a heart attack, yet calculating the probability of getting this condition for any person requires following many people. Moreover, data collected from individuals may have many inaccuracies due to measurement error, missing data, or other limiting factors.

Therefore, the summary data is valuable and considered more reliable because of the increase in sample size. With this idea in mind, consider combining results from multiple clinical trials to increase reliability of data. Just like a single clinical trial is composed of many individuals, a virtual clinical trial would be composed of summary data of clinical trials. One difficulty with using aggregate data is the loss of variability and heterogeneity that is present with individual data. This work takes steps in this direction by trying to better replicate clinical trial result statistics from summary data.

Summary data of a clinical trial baseline population is composed of population characteristics and their statistics. This information is so important to understand that it is typically placed at the beginning of the paper describing the clinical trial—typically in the first table. See (Knopp et al. 2006) for examples. Our goal is to generate a set of fictitious individuals with specific biomarkers such that their summary data will match the existing summary data. This is not a de-identification task; instead, this is an optimization task that will allow us to better understand the heterogeneity and distribution of individuals in the population.

DIFFICULTIES

There are several difficulties associated with regenerating a population to match statistics:

1. **Constraints and statistical outlier control**: Natural distributions do not behave like statistical number generators. For example considering a normal distribution with age 80 and SD of 10, it is possible to get a person with the age above 130 - very rare, yet quite possible since the theoretical function in unbound and the random generator may produce such an outlier. Therefore there is a need to cope with outliers.

2. **Correlation between biomarkers**: Biomarkers in the real world are tied to each other. For example, cholesterol biomarkers levels are not independent and are related. Total cholesterol can be expressed as a function of HDL, LDL, and Triglycerides using the Friedewald formula (Johnson et. al 1997). Another example is cholesterol statistics reported separately for men and women. The generating system has to cope with many such correlations.

3. **Skewed distributions**: Inclusion and exclusion criteria defined by the clinical trial may skew a distribution by introducing boundaries. For example if a trial inclusion criteria is age of at least 40 and the reported statistic is average age of 45 with SD of 5, it is clear that the distribution has a long tail and is not normal. Note that correlations among biomarkers and outlier control may skew distributions further, like the example in the abstract where age and age at diagnosis of diabetes are related. Finally, any random

generator will include a random error that may introduce a bias to the population. The system should cope with these phenomena.

4. **Missing and error information**: The information collected by the clinical trial is missing and contains errors or even bias in selection, or it contains otherwise unreported information. This is normal since there is no perfect knowledge. There are many issues we still do not understand, or information is unreachable—for example correlations between biomarkers are generally unknown, even though there are some isolated reports on those and some basic rules have been formulated. No system can provide perfect knowledge in such conditions, yet the system can allow raising hypothesis to be tested, and this is the aim in this work.

PROPOSED SOLUTION

The generation system will cope with the above difficulties by a Monte Carlo generation system with a domain-specific language (DSL) (Barhak and Isaman 2010, Barhak 2012B, Barhak 2013C). The system is called MIcro-Simulation Tool (MIST) since each individual is simulated separately. Yet for population generation purposes, the entire population is also considered by invoking evolutionary computation for selection of the most fitting sub-population.

Figure 1 describes the population generation process that is composed of two stages:

Stage 1: individual generation where a large number of individuals are generated from 1) distributions and formulas, 2) bounds and limitations imposed on parameter values, 3) population size and control parameters. This stage takes care of difficulties 1 and 2 by allowing the user to define distributions, bounds, and formulae that tie variables. Previous publications provide information about those (Barhak and Isaman 2010, Barhak 2012B, Barhak 2013C), and this is not the focus in this paper.

Stage 2: where a genetic algorithm is used to select a sub population to best fit population statistics. These statistics are defined as objectives by the user who also supplies optimization parameters for the genetic algorithm. This stage takes care of difficulty 3 of skewed distributions by introducing control over the error from desired objectives. The next section will focus on this stage.

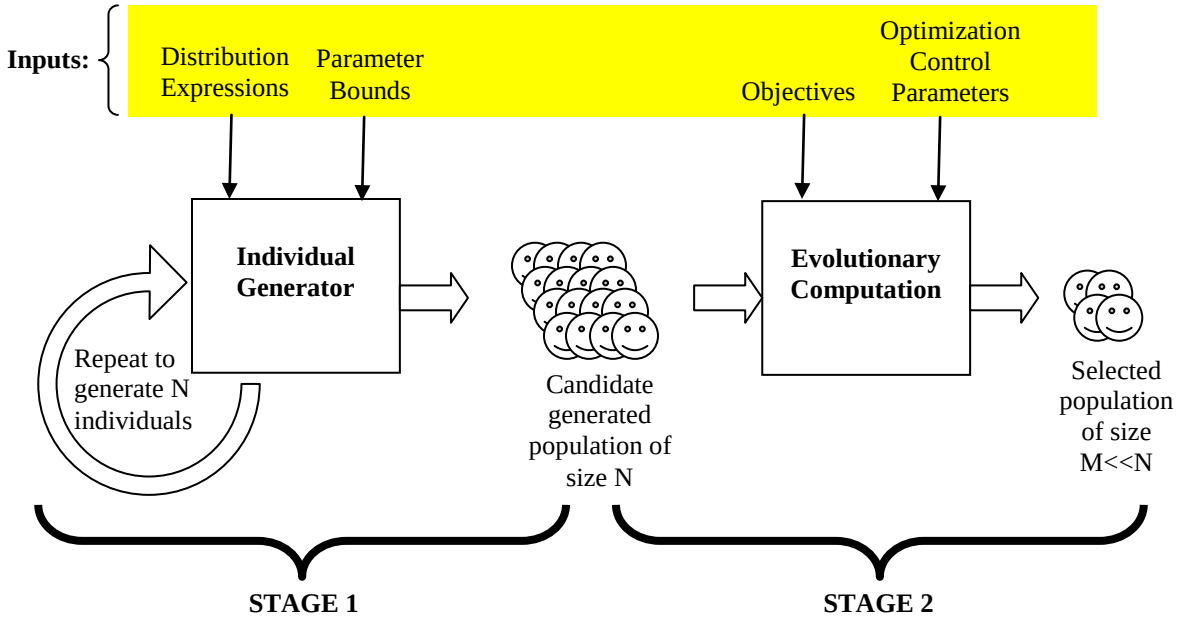

Figure 1. Population generation workflow

Sub-population Optimization

The genetic algorithm optimizes the sum of squares of an objective error vector. Each element in this vector is defined by the following characteristics of the objective j that depends on parameter values $\mathbf{p}_i$ of individual i:

1. **Filter Expression**: an expression using the DSL that, if evaluates to non-zero for a specific individual, this individual will be included in calculating the statistics. This is useful to define objectives that concern only a subgroup of a population, such as counting only women or counting only individuals with high blood pressure (BP). We will denote it $f_j(\mathbf{p}_i)$
2. **Statistics Expression**: This expression using the DSL will define the contribution of each individual to the objective error. In many cases the expression is simply the parameter of interest such as age, BP, or HDL. Yet sometimes it is a different calculation such as Gr(Age,60) if our target is the proportion of people above the age of 60. We will denote it $s_j(\mathbf{p}_i)$
3. **Statistics Function**: this defines how to aggregate the individual contributions defined by the statistics expression. The system supports a variety of statistical functions, such as MEAN, STD, and MEDIAN, along with other helper functions such as COUNT, MIN, and MAX. Once this function is applied to the vector of Statistics Expressions defined for each individual that passes the Filter Expression, the system can calculate a single number for each objective. We will denote it as $\Omega_j(\{s_j(\mathbf{p}_i)\forall i, f(\mathbf{p})_i \neq 0\})$ or Ω_j in short.
4. **Target Value**: a scalar defining the target value generated by the statistics function. We will denote it as t_j.
5. **Weight**: a scalar defining the weight to multiply the square difference between the Target Value and the Statistic Function result. It is useful since objectives have different importance and different scales. The Weight allows compensating and regulating this for each objective to improve optimization. We will denote it as w_j.

The objective error vector elements are therefore: $e_j = [(\Omega_j - t_j)w_j]^2$, and the overall error for all objectives that is optimized is $e = \sum_j e_j$.

Evolutionary Computation

Evolutionary computations (of which genetic algorithms are one type) operate on potential solutions to a given problem. These potential solutions are called individuals. The quality of a particular individual is referred to as its *fitness*, which is used as a measure of survivability (DeJong 2006). Most evolutionary computations maintain a set of individuals (referred to as a *population*). During each *generation*, or cycle, of the evolutionary computation, individuals from the population are selected for modification, modified in some way using evolutionary operators (typically some type of recombination and/or mutation) to produce new solutions, and then some set of existing solutions is allowed to continue to the next generation (Fogel 2000). Viewed in this way, evolutionary computation essentially performs a parallel, or beam, search across the landscape defined by the fitness measure (Russell and Norvig 2000, Spears et al. 1993). A beam search is simply a search algorithm that maintains *k* states, rather than just one state, at each iteration.

The evolutionary computation chosen for optimization is implemented using INSPYRED and uses the following evolutionary operators. Selection of individuals for reproduction is carried out using tournament selection, which means that a group of *k* existing individuals (where *k* is the tournament size) is selected from the population and the individual with the greatest fitness is selected. (This process is repeated as necessary in order to select the required number of individuals for reproduction.) Selection of individuals for survival into the next generation is accomplished using generational replacement with elitism, which means that all newly created children replace the existing members of the population, except for a small number (*e*, a nonnegative integer parameter) of elite individuals that have greater fitness values than those of the children. Using elitism is one approach that prevents a good solution from being discarded during the evolution.

The variation operators used were crossover and mutation. Crossover merged two potential solutions by including all individuals used in both solutions and randomly selecting individuals that differ between parents. Mutation changes a small number of individuals in a solution from the pool of individuals not being used. To avoid confusion with regards to individual definition note that both variation operators handle sub-populations of the generated individuals of stage 1. Each sub-population solution is considered as an individual by the evolutionary computation. In other words, the evolutionary computation in this case handles a population of populations of individuals.

EXAMPLE

This example is provided as part of the MIST version 0.90.0.0 and demonstrates the system capabilities. To reproduce these results use the example "Population set for Simulation Example 21."

This example will generate 6000 candidate Individuals with the following generating functions: for Stage 1:

Age = Uniform(1,59)

Gender = Bernoulli(0.6)

In stage 2 of the generation process, 600 individuals will be selected that will fit most the objectives in Table 1:

Objective #	Filter Expression	Statistics Expression	Statistics Function	Target	Weight
1	Le(Age,20)	Age	MEAN	5	1
2	And(Gr(Age,20), Le(Age,40))	Age -25	MEAN	0	1
3	Gr(Age,40)	Age	MEAN	45	1
4	1	Gender	MEAN	0.5	1
5	1	Age*(Gender-0.5)	MEDIAN	0	1
6	Le(Age,20)	Age	STD	1	0.1
7	And(Gr(Age,20), Le(Age,40))	Age	PERCENT25	24	0.1
8	And(Gr(Age,20), Le(Age,40))	Age	PERCENT75	26	0.1
9	Gr(Age,40)	Age	MIN	42	0.1
10	Gr(Age,40)	Age	MAX	48	0.1
11	And(Le(Age,20), Eq(Gender,0))	1	SUM	100	0.02
12	And(Gr(Age,20), Le(Age,40) , Eq(Gender,0))	1	SUM	100	0.02
13	And(Gr(Age,40) , Eq(Gender,0))	1	SUM	100	0.02
14	And(Le(Age,20), Eq(Gender,1))	Age **2	COUNT	100	0.02
15	And(Gr(Age,20), Le(Age,40) , Eq(Gender,1))	Age **2	COUNT	100	0.02
16	And(Gr(Age,40) , Eq(Gender,1))	Age **2	COUNT	100	0.02

Table 1. Population objectives

These objectives show a variety of ways to define targets—some are overly complicated to show and test features of the system. These objectives try to generate a population with three age groups (0-20, 20-40, 40-60) with an equal number of men and women in each, where the mean age per group is 5, 25, 45, respectively. This is a very strange distribution that is particularly skewed and deformed, considering that the initial generated population is uniform in age with more individuals with Gender =1. Therefore, this example demonstrates the capabilities of the genetic algorithm to select a skewed population. Furthermore some of the objectives are defined in overly complicated and redundant ways to demonstrate a variety of possibilities the system supports.

The system parameter RandomSeed was set to 1 before generation to allow reproducibility; otherwise the code for this example is supplied with MIST 0.90.0.0 and the example ran on Win7-64bit with Anaconda 1.9.1 and INSPYRED 1.0. Table 2 shows results after running the generation process twice. The first time the generation process was executed using default parameters for 7500 evaluations of the genetic algorithms, and the second time it ran until no change in overall error was encountered for 6 generations (approximately 600 evaluations), which improved results.

Figure 2 shows convergence to the requested objectives in this relatively constrained scenario. The statistics of the population are met as far as possible considering the optimization parameters that includes the ratio between the generated population size and the selected sub population, i.e. 6000/600 in this case. The greater this ratio the more selection is available for the genetic algorithm, making it easier. Other parameters will control the convergence of the genetic algorithm, such as maximum number of iterations.

Objective Definition		Generation Stopped after 7500 evaluations		Generation stopped after no chance encountered for 6 generations	
#	Target	Value Reached	Objective Error	Value Reached	Objective Error
1	5	5.883993	0.781443	5.088756	0.007878
2	0	0.686442	0.471202	-0.059375	0.003525
3	45	45.725778	0.526753	45.059937	0.003592
4	0.5	0.498333	0.000003	0.500000	0.000000
5	0	-0.514655	0.264869	-0.004685	0.000022
6	1	3.980135	0.088812	3.244109	0.050360
7	24	22.019375	0.039229	21.893020	0.044394
8	26	28.715509	0.073740	26.823484	0.006781
9	42	40.004509	0.039820	40.004509	0.039820
10	48	57.826645	0.965629	53.229683	0.273496
11	100	92	0.025600	99	0.000400
12	100	98	0.001600	98	0.001600
13	100	111	0.048400	103	0.003600
14	100	95	0.010000	100	0.000000
15	100	106	0.014400	100	0.000000
16	100	98	0.001600	100	0.000000
Overall Error		3.353101		0.435468	
Time (Min)		~3.5		~7	

Table 1. Population generation results per objective

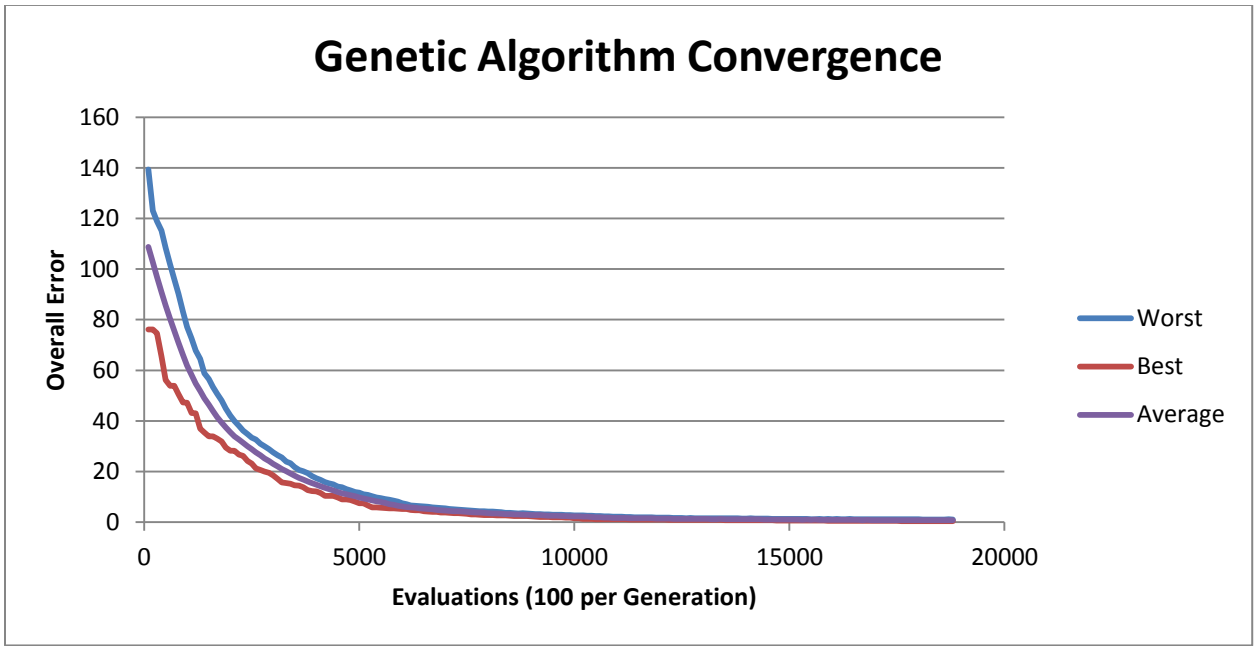

Figure 2. Convergence of the evolutionary computation

DISCUSSION:

The example demonstrates how the system copes with difficulty 3 of skewed distributions. With sufficient time and selection, the target objectives will be reached within reasonable tolerance while skewing the initially generated population distribution. This is very similar to the way population distributions are skewed when inclusion/exclusion criteria are introduced into clinical trials. Therefore, with sufficient information about the populations from which individuals are recruited, it may be possible to reach a similar skewed distribution result. Although this is a promising step forward, this task is still difficult since our knowledge is still quite limited as noted on difficulty 4 above.

This fourth difficulty of insufficient knowledge and information is the real obstacle to planning and understanding trial results. However, with the ability to reliably generate populations, it is possible to at least offer hypotheses and test those for fitness against observed phenomena. This is the main contribution of this work.

This example demonstrates how the issue of fitting a distribution becomes a problem of computational power whereas before it was an issue of human expertise without proper support tools. With the availability of computing power and High Performance Computing (HPC) environments, the entire task become much simpler.

The Reference Model for disease progression (Barhak 2012A, 2012B, 2012C, 2012D, 2012E, 2013B, 2013D, Barhak and Leff 2013A) is one application that relies on computing power and requires this ability to generate better populations. Other possible future applications for this work may include support for design/planning/prediction of recruitment to reach a desired distribution of a clinical trial.

To allow others to benefit from this work, the implementation is offered under General Public License that allows copying a reuse without fees. The MIST software that controls population generation is available through: https://github.com/Jacob-Barhak/MIST and it uses the INSPYRED software that implements the genetic algorithms available through https://github.com/inspyred/inspyred .

ACKNOWLEDGEMENTS

The IEST GPL disease modeling framework was initially supported by the Biostatistics and Economic Modeling Core of the MDRTC (P60DK020572) and by the Methods and Measurement Core of the MCDTR (P30DK092926), both funded by the National Institute of Diabetes and Digestive and Kidney Diseases. The modeling framework was initially defined as GPL and was funded by Chronic Disease Modeling for Clinical Research Innovations grant (R21DK075077) from the same institute. The current version of MIST was developed from this GPL framework without financial support.

REFERENCES

Barhak J., Isaman D.J.M., Ye W., Lee D. (2010), Chronic disease modeling and simulation software, Journal of Biomedical Informatics, 43(5) 791-799, http://dx.doi.org/10.1016/j.jbi.2010.06.003

Barhak J. (2012A), The Reference Model in the Mount Hood #6-2012 validation challenge and the uncertainty challenge. The Mt hood challenge 6, June 7-8, 2012. Johns Hopkins Mt. Washington Conference Center.

Barhak J. (2012B), The Reference Model for Disease Progression. SciPy 2012, Austin Tx, 18-19 July 2012. Paper:
https://github.com/Jacob-Barhak/scipy_proceedings/blob/2012/papers/Jacob_Barhak/TheReferenceModelSciPy2012.rst
Poster:
http://sites.google.com/site/jacobbarhak/home/PosterTheReferenceModel_SciPy2012_Submit_2012_07_14.pdf

Barhak J. (2012C) , The Reference Model for Chronic Disease Progression. 2012 Multiscale Modeling (MSM) Consortium Meeting, October 22-23, 2012,
Abstract:
http://www.imagwiki.nibib.nih.gov/mediawiki/images/7/77/Reference_Model_for_Chronic_Disease_Progression_Barhak.pdf

Poster:
http://www.imagwiki.nibib.nih.gov/mediawiki/images/c/c4/PosterTheReferenceModel_IMAGE_MSM_Submit_2012_10_17.pdf

Barhak J. (2012D), The Reference Model: Improvement in Treatment Through Time in Diabetic Populations, The Fourth International Conference in Computational Surgery and Dual Training. The Joseph B. Martin Conference Center at Harvard Medical School. Boston, MA, USA. December 9-10-11, 2012. Presentation: http://www2.cs.uh.edu/~cosine/talks_cosine4/monday/MultidisciplinaryTalks/2_JacobBarhak.pptx
Video: http://web.cs.uh.edu/~cosine/?q=node/140

Barhak J. (2012E), The Reference Model: Improvement in Treatment Through Time in Diabetic Populations, The Fourth International Conference in Computational Surgery and Dual Training. The Joseph B. Martin Conference Center at Harvard Medical School. Boston, MA, USA. December 9-10-11, 2012. Presentation Slides: http://sites.google.com/site/ComputationalSurgery_Presneted_2012_12_LateUploadToOwnWebSite_2014_2_27.pptx

Barhak J., Leff H.S., (2013A), Modeling a Chronic Disease Model and a Mental Health Model Using the Same Modeling Tools, MODSIM World 2013, April 30 - May 2nd, Hampton Roads Convention Center in Hampton, VA.
Paper: http://sites.google.com/site/jacobbarhak/home/MODSIM_World2013_Submitted_04Apr2013.pdf
Presentation: http://sites.google.com/site/jacobbarhak/home/MODSIM_World_Presented_2013_05_2.pptx

Barhak J. (2013B), The Reference Model Scores Fitness of Models and Populations. Poster Presentation. ISPOR 18th Annual International Meeting, May 18-22, 2013, Sheraton New Orleans, New Orleans, LA, USA. Poster: http://sites.google.com/site/jacobbarhak/home/PosterTheReferenceModel_ISPOR_Submit_2013_05_14.pdf

Barhak J. (2013C), MIST: Micro-Simulation Tool to Support Disease Modeling. SciPy, 2013, Bioinformatics track, https://github.com/scipy/scipy2013_talks/tree/master/talks/jacob_barhak,
Presentation: http://sites.google.com/site/jacobbarhak/home/SciPy2013_MIST_Presented_2013_06_26.pptx
Video: http://www.youtube.com/watch?v=AD896WakR94

Barhak J (2013D), The Reference Model for Disease Progression Sensitivity to Bio-Marker Correlation in Base Population - The Reference Model Runs with MIST Over the Cloud! 2013 IMAG Multiscale Modeling (MSM) Consortium Meeting, October 2-3, 2013,
Poster:
http://sites.google.com/site/jacobbarhak/home/PosterTheReferenceModel_IMAG_MSM2013_Submit_2013_09_23.pdf

DeJong, K. A. (2006). Evolutionary Computation: A Unified Approach. MIT Press.

Fogel, D. B. (2000). What is evolutionary computation? IEEE Spectrum 37(2):26–32.

Johnson R., McNutt P., MacMahon S., Robson R., (1997). Use of the Friedewald Formula to Estimate LDL-Cholesterol in Patients with Chronic Renal Failure on Dialysis. Clinical Chemistry
November 1997 vol. 43 no. 11 2183-2184. http://www.clinchem.org/content/43/11/2183.full.pdf+html

Knopp R.H., d'Emden M., Smilde J.G., Pocock S.J., (2006) Efficacy and Safety of Atorvastatin in the Prevention of Cardiovascular End Points in Subjects With Type 2 Diabetes: The Atorvastatin Study for Prevention of Coronary Heart Disease Endpoints in Non-Insulin-Dependent Diabetes Mellitus (ASPEN), Diabetes Care 29:1478-1485, July 2006, http://dx.doi.org/10.2337/dc05-2415

Russell, S., and P. Norvig. (2002). Artificial Intelligence: A Modern Approach. Prentice Hall, 2nd edition.

Spears, W. M., K. A. DeJong, T. Bäck, D. B. Fogel, and H. deGaris. (1993). An overview of evolutionary computation. In Proceedings of the 1993 European Conference on Machine Learning.