Subject: drug score/evolutionary algorithm
Posted by pc419714@ohio.edu on Thu, 04 Apr 2019 18:14:15 GMT
View Forum Message <> Reply to Message

Hello Thomas,

The lab I work in is wondering why Data Warrior was programmed to only allow a certain number of compounds to survive each generation. For instance, if I set the properties in the evolutionary algorithm so that there are 400 generations, 4096 compounds per generation, and 32 surviving each generation, we end up with 12800 compounds surviving. From my understanding, the best compounds each generation have the best drug scores. We have 400 generations, but wouldn't generation 400 have the best compounds? So, why not only have compounds from generation 400 survive because previous generations would have lower drug scores? I would think if you did something like this, the compounds would all be pretty similar to one another and perhaps keeping 32 from each generation would allow for more diversity. Would the compounds from generation 1 be less fit than generation 400? What were the weights for the parameters you used to calculate the drug score?

Thanks! We will be sure to cite you. This is wonderful software.

Patrick Chirdon

Subject: Re: drug score/evolutionary algorithm
Posted by pc419714@ohio.edu on Fri, 05 Apr 2019 00:56:06 GMT
View Forum Message <> Reply to Message

Please correct me if I am wrong--- For our evolutionary algorithm we selected 400 generations, 4096 compounds per generation, and 32 surviving each generation and set it so that it generates structures like approved drugs. The fitness was set to skelsphere similarity with a weight of 1.

After looking at the source code I have come to the following conclusions. Data Warrior uses java's chemistry development kit. Each generation, it finds the compounds most similar to the parent compound based on skelsphere similarity while leaving the selected structure untouched using fragments from known drugs. It selects the 32 most similar to the parent each generation as the most fit because that's what we set it to. It's not calculating fitness based on the drug score. The drug score is only calculated when you select calculate properties after generating all the compounds.

Possible mutations-- add atom, insert atom, change atom, cut out atom, delete atom, add bond, change bond, delete bond, change ring, group migration, swap substituent, delete substituent, cutout fragment. After generating a mutation, it checks to see if the structure has proper valence and if there is ring strain. It also calculates the frequency of the mutation in the population to determine the probability of each mutation. The probability would really depend on your specific molecule and if what's being generated is a valid molecule.

The drug score was that equation I sent you in the previous email-- it's not used in the evolution. It's generated using that equation I sent you in the previous email. Drug Score is different from druglikeness. Druglikeness= nasty incrementsum + increment sum / sqrt(fragment count)

drug
score=(0.5+0.5/(1+exp(cLogP-5))*(1-0.5/(1+exp(cLogS+5))*(0.5+0.5/(1+exp(0.012*Molweight-6))*(1-0.5/(1+exp(Druglikeness))*if(Mutagenic=="high",0.6,if(Mutagenic=="low",0.8,1))*if(Tumorigenic
== "high",0.6,if(Tumorigenic=="low",0.8,1))*if(ReproductiveEffective==
"high",0.6,if(ReproductiveEffective=="low",0.8,1))*if(Irritant== "high",0.6,if(Irritant=="low",0.8,1)

Data warrior assigns nasty functional groups various scores called increments. It also gives compounds with more than 50 druglike fragments as more druglike.

The toxicity predictor relies on compounds from the RTECS database. I can probably request it from the database site.

Thanks so much. They just want to know as many details as possible. I've been trying to write my own machine learning algorithms in rdkit. I haven't played much with chemistry development kit.

Patrick

Subject: Re: drug score/evolutionary algorithm
Posted by pc419714@ohio.edu on Fri, 05 Apr 2019 00:57:06 GMT
View Forum Message <> Reply to Message

smileys are double brackets that somehow accidentally got turned into emoji's.... lol

---

Subject: Re: drug score/evolutionary algorithm
Posted by thomas on Mon, 08 Apr 2019 14:13:04 GMT
View Forum Message <> Reply to Message

Hello Patrick,

DataWarrior does not use the CDK (chemistry development kit). It is based on our own cheminformatics functionality, which is part of the DataWarrior source code. Most of it is also available as open-source on GitHub as "OpenChemLib".

Otherwise your assumptions are mostly correct. I summarize here with my words:

The evolutionary algorithm does not calculate the 'Drug Score' of the 'Property Explorer' unless you define something similar as fitness criteria, as you already have concluded. It uses random mutations to generate structurally slightly changed molecules from a set of previously generated molecules (the previous generation). For applying a random change on a previous generation molecule, DataWarrior generates a list of possible mutations, e.g. adding a atom (C,N,O or other) to any existing atom, changing the bond order of an existing bond, doing a ring expansion,

migrating a substituent, ...
For everyone of these mutations DataWarrior is assigning a likelihood by looking at the atom types that disappear or a created during the change. An atom type is basically the atomic number of an atom and its neighbours plus bond orders between them, plus whether atoms are in a ring, aromatic, allylic, stabilized by carbonyl-neighbours, etc. Now comes into the game, whether you have selected 'drug like' or natural product like', because depending on that DataWarrior uses a different atom type frequency table to calculate mutation likelihoods. The effect is finally, that DataWarrior when selecting a mutation will prefer those that produce drug-like (or NP-like) atoms rather than destroys them. These are atoms with a local environment that is typical for drug like molecules.

Your idea to create more molecules in the last generation than in all the others makes perfectly sense, provided that you are not looking for one, but for many molecules that fulfill your fitness criteria. Alternatively, you may leave the algorithm running after fitness maximization has been achieved. Then, the algorithm will meander around and produce more molecules with a high fitness, which are connected through a similarity chain. A more diverse set of fit molecules will be created, if you let the algorithm run many times with very small generation sizes. In fact, I am currently working on a new functionality to generate random drug-like or NP-like molecules using a new evolution path for every individual molecule starting from a very tiny seed-molecule. This work surprisingly well. For that I fixed issues with the atom typing and the Mutator class.

Please don't hesitate to als, if things are still unclear.

Thomas

Subject: Re: drug score/evolutionary algorithm
Posted by pc419714@ohio.edu on Tue, 09 Apr 2019 18:06:09 GMT
View Forum Message <> Reply to Message

Thanks! This was helpful.

If you want lots of compounds with high drug scores, it would be better to let everything survive, then find the ones with the best drug score, then evolve those so that only those most similar to the parent compound (which have high drug scores to begin with) survive, with a small generation size, and lots of generations (we'd have to define where to stop unless you wanted to select unlimited). This way we would have high drug scores and diversity.

We decided to edit the source code so everything survives first.

I still have to graph generation number vs. drug score, but from what it sounds like when you use the genetic algorithm, the drug score may not be improving from the first generation to the last.

Thanks I will let you know if they have any more questions.