Australian National University

Evolutionary Computation

Part of COMP4660/8420:
Neural Networks, Deep Learning and Bio-inspired Computing

4. Limitations of Evolutionary Computation

Prof. Tom Gedeon and Mr. Zhenyue Qin (秦震岳)

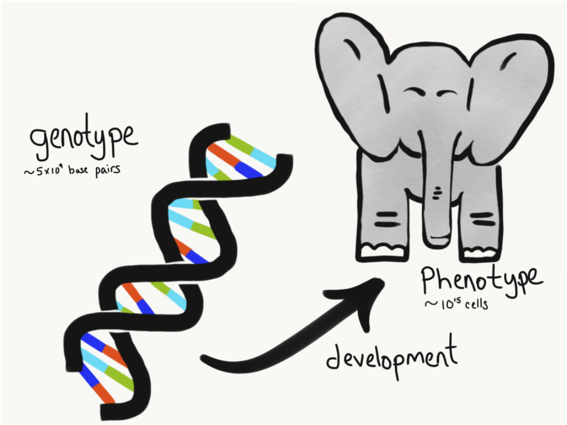
tom@cs.anu.edu.au; zhenyue.qin@anu.edu.au

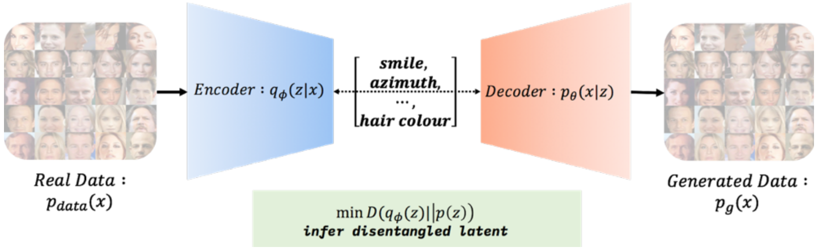
Human Centered Computing (HCC) Laboratory

2021 Semester 1

Representations from Genotypes to Phenotypes

- Like human genes, locations on genotypes can correspond to semantic features.
- Desirable properties can accumulate.
- Mutation and recombination can help to alter semantic features. Selection filters out the highly fit individuals.
- Similar with "representation disentanglement" in deep learning communities.





Australian National University

General Randomized Search Heuristics

- Evolutionary Algorithms are general randomized search heuristics
  - just like:
    - Simulated Annealing,
    - the Metropolis Algorithm,
    - Tabu Search,
    - Scatter Search,
    - Particle Swarm Optimization,
    - Ant Colony Optimization,
    - and countless others.

When to use Evolutionary Algorithms?

- General randomized search heuristics:
- This means:
  - No proven upper bound for running time.
  - Optimal solutions are not guaranteed.
  - No guarantee for any solution quality.

- Conclusion:
  - If there is a problem-specific algorithm known, use that.

When to use Evolutionary Algorithms?

- Benefits of Evolutionary Algorithms:
  - easy to apply
  - easy to implement
  - easy to test
  - often deliver satisfactory results in acceptable time
  - not much harm done, if no success
- Use, if
  - no better algorithm is known,
  - there is no time to develop a problem-specific algorithm,
  - there is no expertise to develop a problem-specific algorithm.

(computation time vs. time for development)

General Limits

- There are so many different evolutionary algorithms. Which one is the best?
- Is there a single best evolutionary algorithm?
- “There ain’t no such thing as a free lunch.”
- No Free Lunch Theorem: “On average, all randomized search heuristics perform equal.”
- Is this surprising?

No Free Lunch (NFL) Theorem Explanation 1

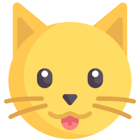
- Apparently mapping 1, 3, and 4 are “wrong”.
  - However, such labeling may happen.
    - Consider a country where people recognize cats and dogs reversely (label 3), or cannot differentiate cats and dogs (label 1 and 4).

	1	2	3	4
	Cat	Cat	Dog	Dog
	Cat	Dog	Cat	Dog

No Free Lunch (NFL) Theorem Explanation 2

- We consider all 4 labeling cases are equally likely to happen.
- We define the prediction error as:

  $E_{ote}(\xi_a|X,f) = \sum_h \sum_{x \in \mathcal{X}-X} P(x) I\big(h(x) \neq f(x)\big) P(h|X,\xi_a)$,
  where ξ_a is a "stupid" algorithm that randomly guess the label,
  $\mathcal{X}$ is the given data, x is a data instance, X is the training data,
  f is the "correct" model,
  h is a model generated by our "stupid" algorithm,
  predicting things randomly.
- We have the following:

	1	2	3	4
	Cat	Cat	Dog	Dog
	Cat	Dog	Cat	Dog

Australian National University

No Free Lunch (NFL) Theorem Explanation 3

$$
\begin{aligned}
E_{ote}(\xi_a|X,f) &= \sum_h \sum_{x\in\mathcal{X}-X} P(x) I\big(h(x)\neq f(x)\big) P(h|X,\xi_a) \\
&= \sum_f \sum_h \sum_{x\in\mathcal{X}-X} P(x) I\big(h(x)\neq f(x)\big) P(h|X,\xi_a) \\
&= \sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\xi_a) \sum_f I\big(h(x)\neq f(x)\big) \\
&= \sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\xi_a) \tfrac{1}{2} 2^{|\mathcal{X}|} \\
&= \tfrac{1}{2} 2^{|\mathcal{X}|} \sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\xi_a) \\
&= 2^{|\mathcal{X}|-1} \sum_{x\in\mathcal{X}-X} P(x) \cdot 1
\end{aligned}
$$

Remember every labeling is possible, thus we have $2^{|\mathcal{X}|}$ total cases.

That is, the error does not depend on our algorithm ξ_a!

Thus, we can use another algorithm ξ_b, giving the same error.

	1	2	3	4
	Cat	Cat	Dog	Dog
	Cat	Dog	Cat	Dog

Australian National University

No Free Lunch (NFL) Theorem Explanation 4

- If all algorithms perform equally, why don't we just use a random one each time?
- However, remember, we consider all 4 labeling cases the same likely to happen.
- In practice, it is unlikely to be true.
- $$\begin{aligned}
E_{ote}(\xi_a|X,f) &= \textstyle\sum_h \sum_{x\in\mathcal{X}-X} P(x) I\big(h(x)\neq f(x)\big) P(h|X,\xi_a) \\
&= \textstyle\sum_f \sum_h \sum_{x\in\mathcal{X}-X} P(x) I\big(h(x)\neq f(x)\big) P(h|X,\xi_a) \\
&= \textstyle\sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\xi_a) \sum_f I\big(h(x)\neq f(x)\big) \\
&= \textstyle\sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\xi_a) \frac{1}{2} 2^{|\mathcal{X}|} \\
&= \textstyle\frac{1}{2} 2^{|\mathcal{X}|} \sum_{x\in\mathcal{X}-X} P(x) \sum_h P(h|X,\xi_a) \\
&= \textstyle 2^{|\mathcal{X}|-1} \sum_{x\in\mathcal{X}-X} P(x) \cdot 1
\end{aligned}$$

f here is not uniformly distributed.
Consider what if we only have one f, then our h can match every single $f(x)$.
Thus, we can do better than a half.

	1	2	3	4
	Cat	Cat	Dog	Dog
	Cat	Dog	Cat	Dog

Australian National University

Final Personal Comments

- We are still naively mimicking the natural evolution.
  - For example, the mapping from genotype to the phenotype is far more complicated than people are currently doing.
- We still need to develop new theory tools for better mathematical analysis.
  - Evolutionary processes exhibit very complex dynamics that allow only limited theory forming.

- Thank you all very much! Any discussion questions?

Australian National University

References

- https://devolab.org/learning-an-evolvable-genotype-phenotype-mapping/
- https://www.flaticon.com/premium-icon/cat_1596812
- https://www.flaticon.com/free-icon/dog_534096
- Zhi-Hua Zhou, Machine Learning.