IS summary

Summary of the Intelligent Systems course lectures at FRI

ib8548

January 20, 2023

Contents

1 Nature inspired computing

1.1 Template of an evolutionary program

1. Generate a population of agents (objects, data structures)
2. Repeat:
   2.1 Compute **fitness** of the agents
   2.2 Select **candidates** for the reproduction (using fitness)
   2.3 Create new agents by **combining** the candidates
   2.4 **Replace** old agents with new ones
   2.5 Stop if satisfied

1.2 Key terms

- **Individual** — any possible solution
- **Population** — a group of all individuals
- **Search space** — all possible solutions to the problem
- **Chromosome** — a blueprint for an individual
- **Trait** — a possible aspect (features) of an individual
- **Allele** — possible settings of a trait
- **Locus** — the position of a gene on the Chromosome
- **Genome** — a collection of all chromosomes for an individual

1.3 Gene representation

- Bit vectors
- Numeric vectors
- Strings
- Permutations
- Trees (representing functions, expressions, programs)

1.4 Linear crossover

Let $\mathbf{x} = (x_1, \ldots, x_N)$ and $\mathbf{y} = (y_1, \ldots, y_N)$.
Select α in $(0, 1)$.
The result of the crossover is $\alpha\mathbf{x} + (1 - \alpha)\mathbf{y}$

1.4.1 Example

$$\alpha = 0.75$$
$$A = (5, 1, 2, 10)$$
$$B = (2, 8, 4, 5)$$

Crossover:

$$\begin{aligned}
(\alpha * A_1 + (\alpha - 1) * B_1, \ldots, \alpha * A_4 + (\alpha - 1) * B_4) &= \\
= (0.75 * 5 + 0.25 * 2, \ldots, 0.75 * 10 + 0.25 * 5) &= \\
= (3.75 + 0.5, 0.75 + 2, 1.5 + 1, 7.5 + 1.25) &= \\
= (4.25, 2.75, 2.5, 8.75)
\end{aligned}$$

We can also choose a different α for each position. For example, using $\alpha = (0.5, 0.25, 0.75, 0.5)$ would result in $(3.5, 6.25, 2.5, 7.5)$.

1.5 Gray coding of binary numbers

Each number differs from the previous by **one bit**.

Binary	Gray
0000	**0000**
0001	**0001**
0010	**0011**
0011	**0010**
0100	**0110**
0101	**0101**
0110	**0100**
0111	**1100**
1001	**1101**
1010	**1111**
1011	**1110**
1100	**1010**
1101	**1011**
1110	**1011**
1111	**1000**

Constructing a n-bit Gray code recursively (example for $n = 2$):

- Create a (n-1) bit list: $\mathbf{0}, \mathbf{1}$
- Reflect: $\mathbf{1}, \mathbf{0}$
- Prefix old entries with 0: $\mathbf{00}, \mathbf{01}$
- Prefix new entries with 1: $\mathbf{11}, \mathbf{10}$
- Concatenate: $\mathbf{00}, \mathbf{01}, \mathbf{11}, \mathbf{10}$

1.6 Lamarckian mutation

- Searches for the locally best mutation

1.7 Gaussian mutation

- Selects a position in the vector of floats and mutates it by adding a Gaussian error

1.8 Selection

- Proportional
- Rank proportional
- Tournament
- Single Tournament
- Stochastic universal sampling

1.8.1 Proportional (roulette wheel) selection

Each individual gets a share on the "wheel" depending on its fitness. Fitter individuals get bigger shares.

1.8.2 Tournament selection

1. Set $\mathbf{t}$ = size of the tournament, $\mathbf{p}$ = probability of a choice
2. Randomly sample t agents from the tournament population
3. With probability p select the best agent
4. With probability $p(1-p)$ select the second best agent
5. With probability $p(1-p)^2$ select the third best agent
6. ...

The **n-th** fittest agent is therefore selected with the probability of $\mathbf{p(1-p)^{(n-1)}}$

1.9 Stochastic universal sampling (SUS)

- **Unbiased**
- Selecting N agents with combined fitness of F
- Randomly chosen first position $r \in [0, \frac{F}{N}]$
- The positions $r + i * \frac{F}{N}; i \in 0, 1, \ldots, N-1$ determine the chosen agents

1.10 Niche specialization

- Punish too similar agents

1.11 Stopping criteria

- Number of generations
- Progress
- Availability of computational resources
- ...

1.12 Parameters of genetic algorithms

- Encoding
- Length of the Strings
- Size of the pupolation (from 20–50 to a few thousand)
- Selection method
- Probability of performing a crossover (usually high — ~ 0.9)
- Probability of performing a mutation (usually low — below 0.1)
- Termination criteria (usually a number of generations or a target fitness)

1.13 Pros and cons of GA

Pros:

- **Low** time and memory **requirements** compared to searching a very large feature space
- A solution can be found without any explicit analytical work

Cons:

- Randomized and therefore not optimal
- Can get stuck on local maxima
- We have to figure out how to represent candidates with the available methods (e.g. as a bit string)

2 Predictive modelling

2.1 Learning

Acquiring new, modifiying and/or reinforcing existing:

- Knowledge
- Behaviors
- Skills
- Values
- Preferences

Statistical learning deals with finding a predictive function based on the data.

2.2 Notation

$$X = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix}$$

$$Y = f(X) + \varepsilon$$

$$Y_i = f(X_i) + \varepsilon$$

ε — measurement errors, its mean is 0 and it is independent from X.

2.3 Goals of learning

- Prediction
- Inference

2.4 Prediction

- If we have a good estimate for f, we can make accurate predictions for Y based on a new value of X.

2.5 Inference

- Finding out the relationship between Y and X.
- Sometimes more important than prediction (for example in medicine)

2.6 Parametric methods

- We estimate f by estimating the set of parameters

1. Come up with a model based on assumptions about the form of f, e.g. a linear model:

$$f(\mathbf{X_i} = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_p X_{ip})$$

   - More complicated and flexible models for f are often more realistic

2. Use the training data to fit the model (estimate f) — e.g. the least squares method in the case of linear models

2.7 Non-parametric methods

- They do not make assumptions about the funct. form of f. They fit a wider range of possible shapes, but require a large number of observations compared to parametric methods.

2.8 Types of learning

- Supervised
- Unsupervised
- Semi-Supervised
- Self-Supervised
- Weakly-supervised

2.8.1 Supervised learning

Both the predictors (X_i) and the response (Y_i) are given.

2.8.2 Unsupervised learning

We don't know Y — we are looking for similarities between attributes (X).

2.8.3 Semi-supervised learning

Only a small sample of labelled instances is observed, but a large set of instances is unlabelled.

- A supervised model is used to label unlabelled instances
- The most reliable predictions are then added to the training set for the next iteration of supervised learning

2.8.4 Self-supervised learning

- A mixture of supervised and unsupervised learning
- Learns from unlabelled data
- The labels are obtained **from related properties of the data**

Examples:

- in NLP: predicting hidden words in a sentence based on the remaining words
- in video processing: predicting past or future frames from the observed ones

2.8.5 Weakly-supervised data

Using imprecise or noisy sources to supervise labelling large amounts of training data, then performing supervised learning.
Example: using a smart electricity meter to estimate household occupancy

2.9 Criteria of success for machine learning

How to select the best model? Most popular criterions:

- For **regression**: mean squared error (MSE)

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - f'(x_i)\right)^2$$

- For **classification**: classification accuracy (CA)

$$CA = \frac{1}{n}\sum_{i=1}^{n} I(y_i = y_i')$$

2.10 No free lunch theorem

If no information about $f(X)$ is provided:

- No classifier is better than some other in the general case
- No classifier is better than random in the general case

3 Bias, variance and predictive models

4 Feature selection

5 Ensemble methods

6 Kernel methods

7 Neural networks

8 Inference and explanation

9 Natural language processing

10 Reinforcement learning