Генетические алгоритмы

Сергей Николенко

Машинное обучение — ИТМО, осень 2006

Outline

Эволюция

Генетические алгоритмы тоже списаны с природы.

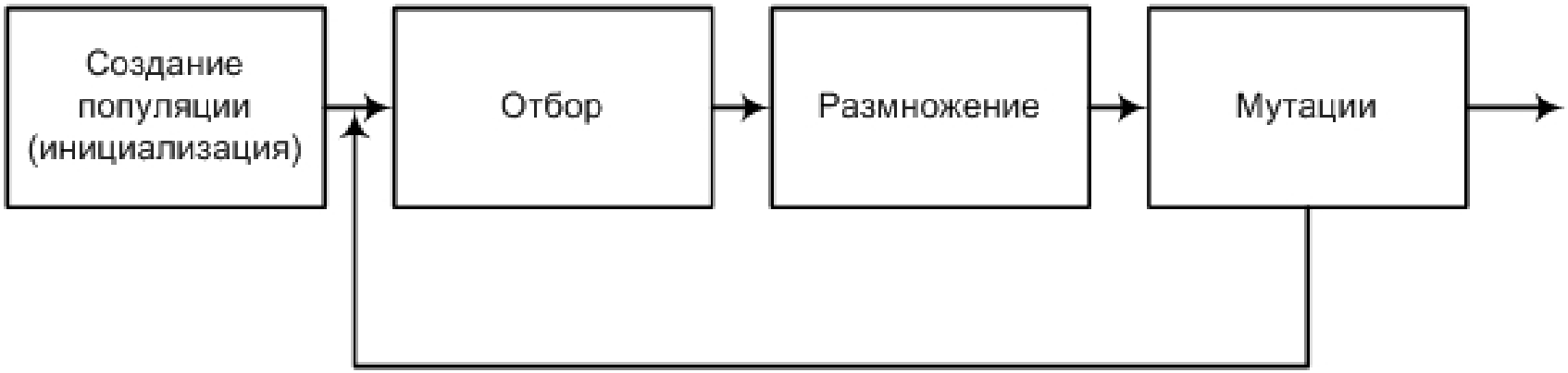
- Организмы эволюционируют со временем, изменяя свой генотип.
- Механизм дарвиновской эволюции:
  - Родилось новое поколение.
  - Из него часть особей выросла и дала потомство, часть погибла.
  - Погибают неприспособленные, выживают приспособленные, у потомков остаются лучшие черты.

Основные компоненты

- Пространство гипотез, из которых мы должны выбрать лучшую
- Функция приспособленности $\mathrm{Fitness}$
- Набор генетических операций, которые можно применять:
  - Операции скрещивания (кроссовер) — размножение особей.
  - Мутации — редкие изменения отдельных особей.
- Целевое значение $\mathrm{Fitness}_{max}$, к которому мы стремимся

Общая схема алгоритма

- Сгенерировать начальную популяцию.
- Пока не достигнуто значение, большее $\mathrm{Fitness}_{max}$:
  - Выбрать часть существующей популяции (отдавая предпочтение более приспособленным особям).
  - Применить к этой части генетические операции, породив потомков.
  - Подсчитать Fitness для особей новой популяции.

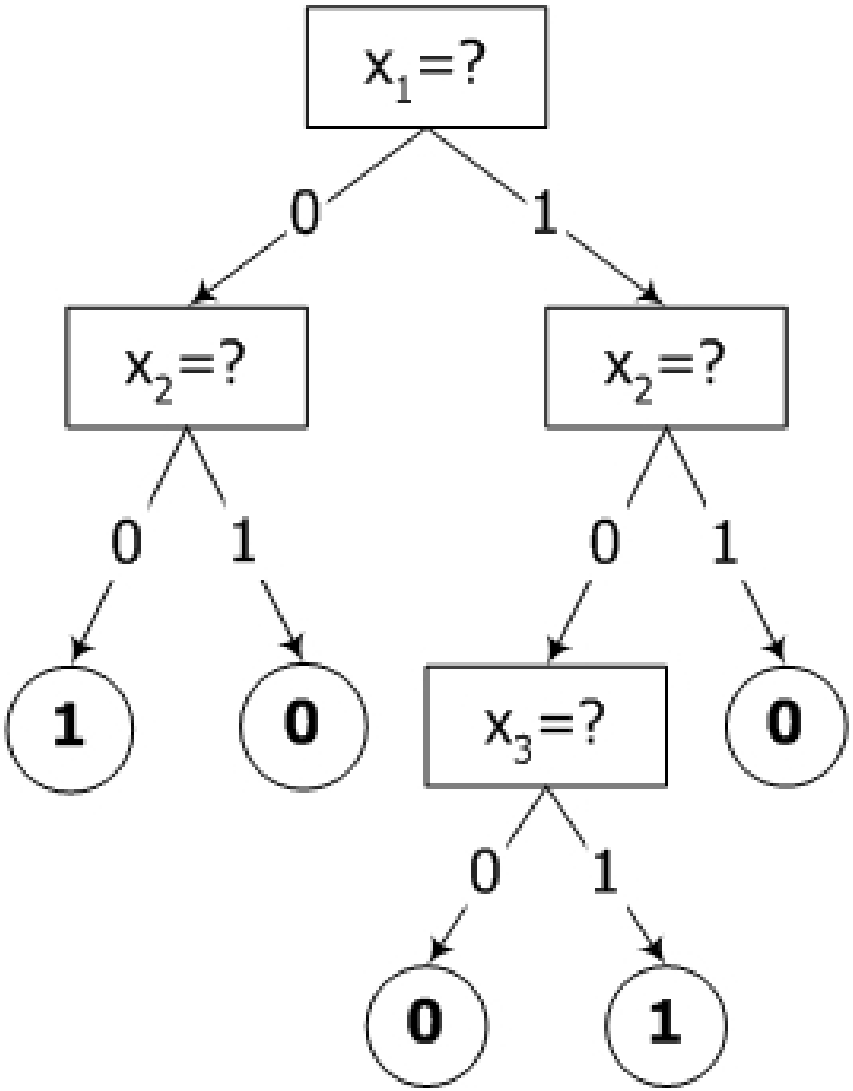


Outline

Представление гипотез

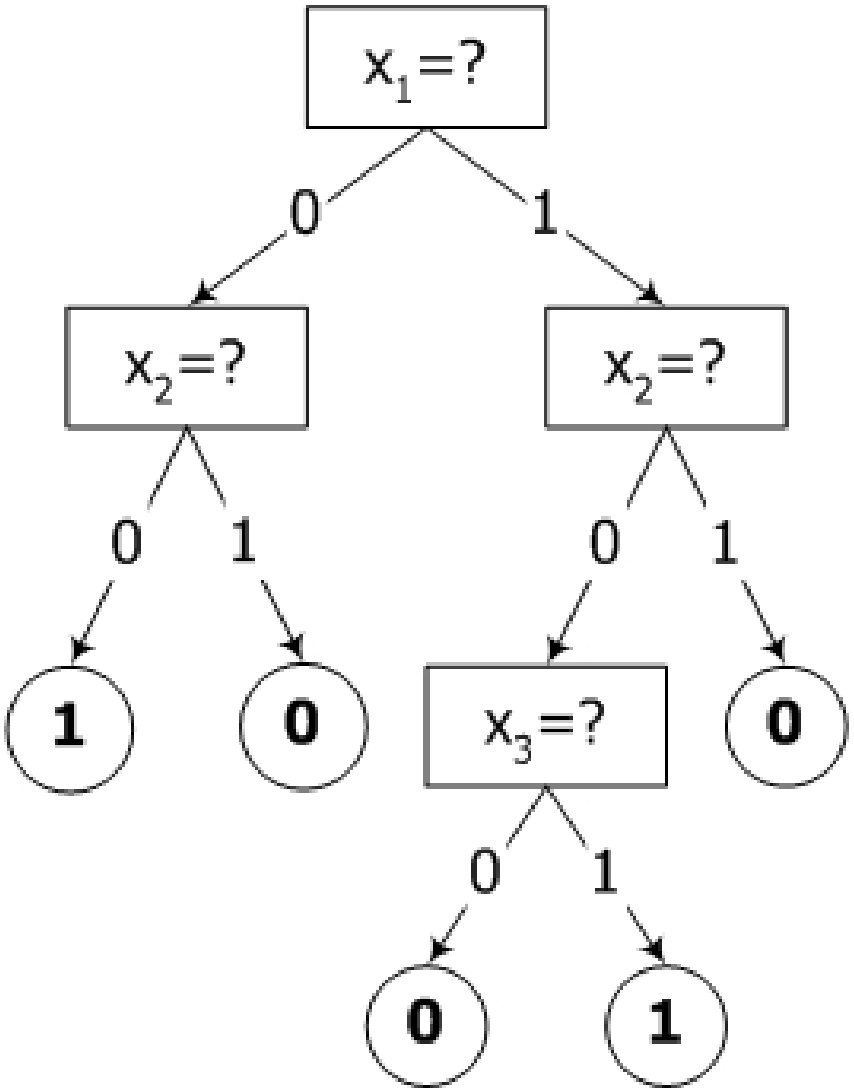
Чтобы успешно применять генетические алгоритмы, гипотезы желательно представлять в виде строки битов. Тогда с ними легко делать что угодно. Но как представлять?

Представление гипотез: пример



Вспомним пример дерева принятия решений.
Гипотеза, она же особь популяции — это в данном случае правило, описывающее поведение целевой функции. A Fitness — это то, насколько хорошо правило соответствует тестовым примерам.

Представление гипотез: пример



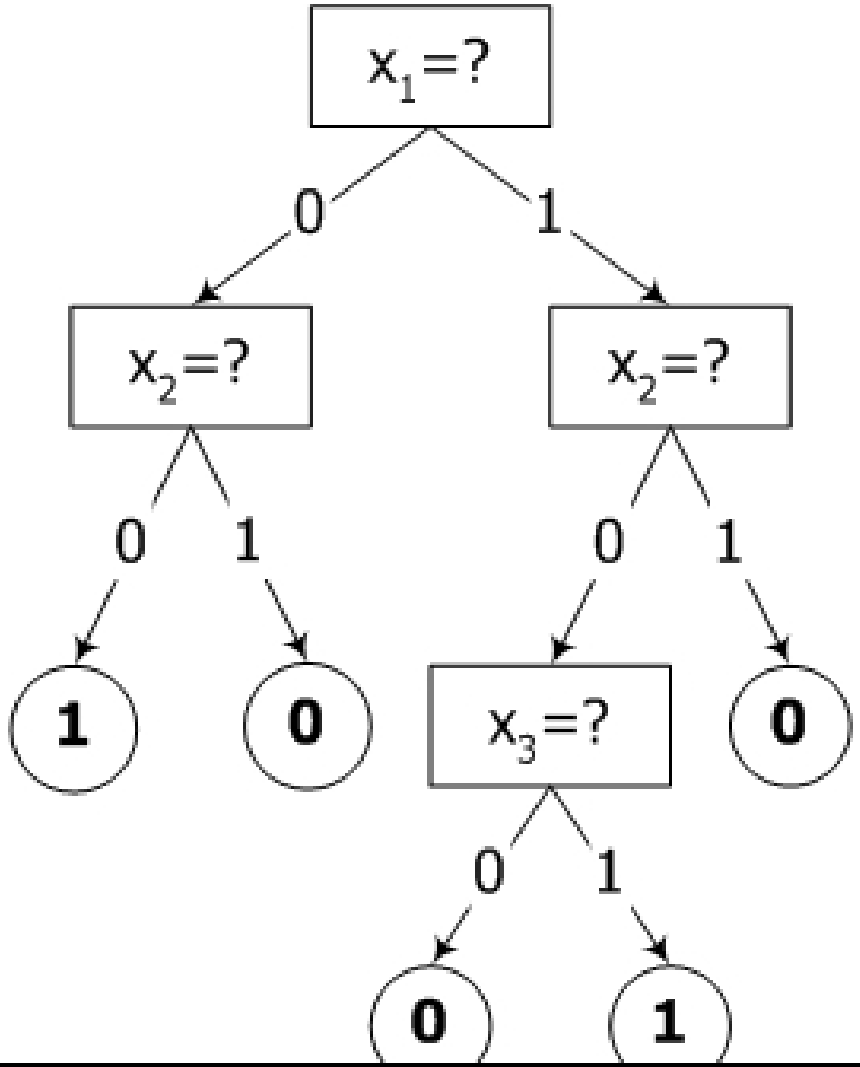
Например, для функции, заданной этим деревом, гипотеза

$$(x_1 = 0) \wedge (x_2 = 1) \Longrightarrow (f = 0)$$

будет более приспособленной, чем гипотеза

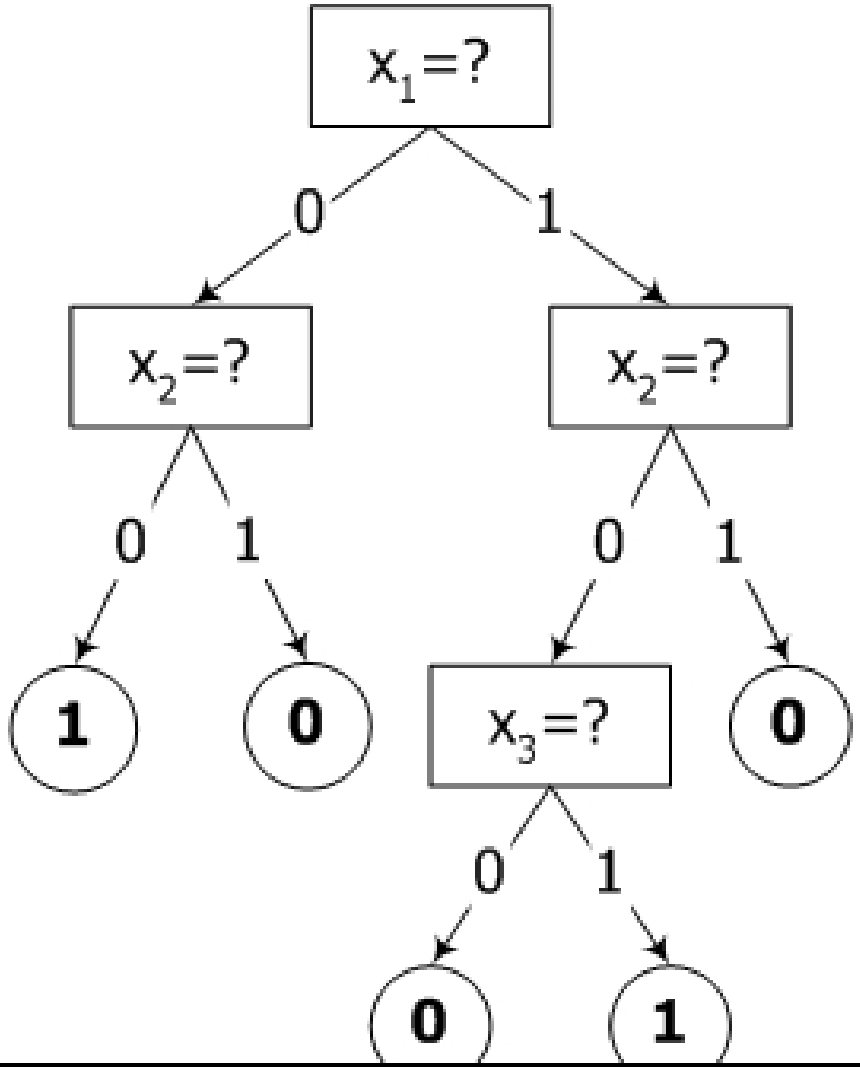
$$(x_1 = 0) \Longrightarrow (f = 0).$$

Представление гипотез: пример



Как задать гипотезу строкой из бит?

Представление гипотез: пример



Как задать гипотезу строкой из бит?

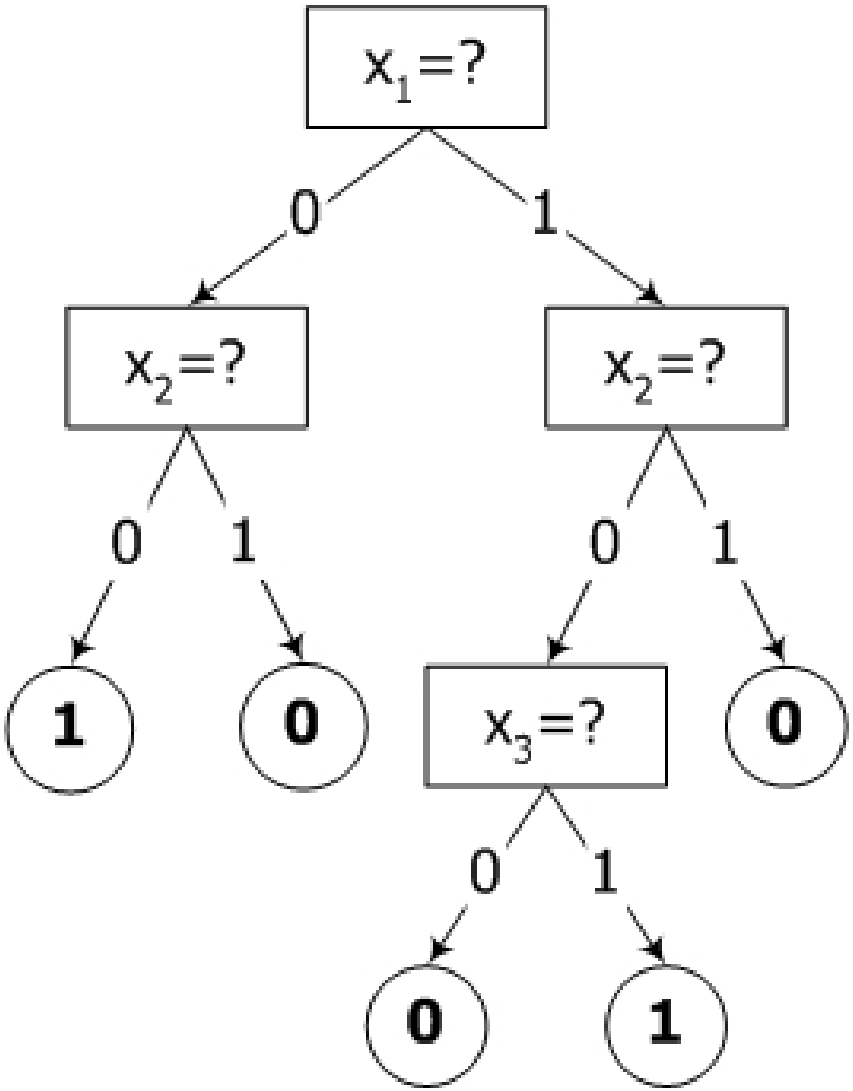
Логичная идея: закодировать каждую булевскую переменную битом; например, гипотеза

$$(x_1 = 1) \wedge (x_2 = 0) \wedge (x_3 = 1) \Rightarrow \Rightarrow (f = 1)$$

представляется как

x_1	x_2	x_3	f
1	0	1	1

Представление гипотез: пример



Но как тогда закодировать гипотезу

$$(x_1 = 0) \wedge (x_2 = 1) \Rightarrow (f = 0)?$$

x_1	x_2	x_3	f
0	1	???	0

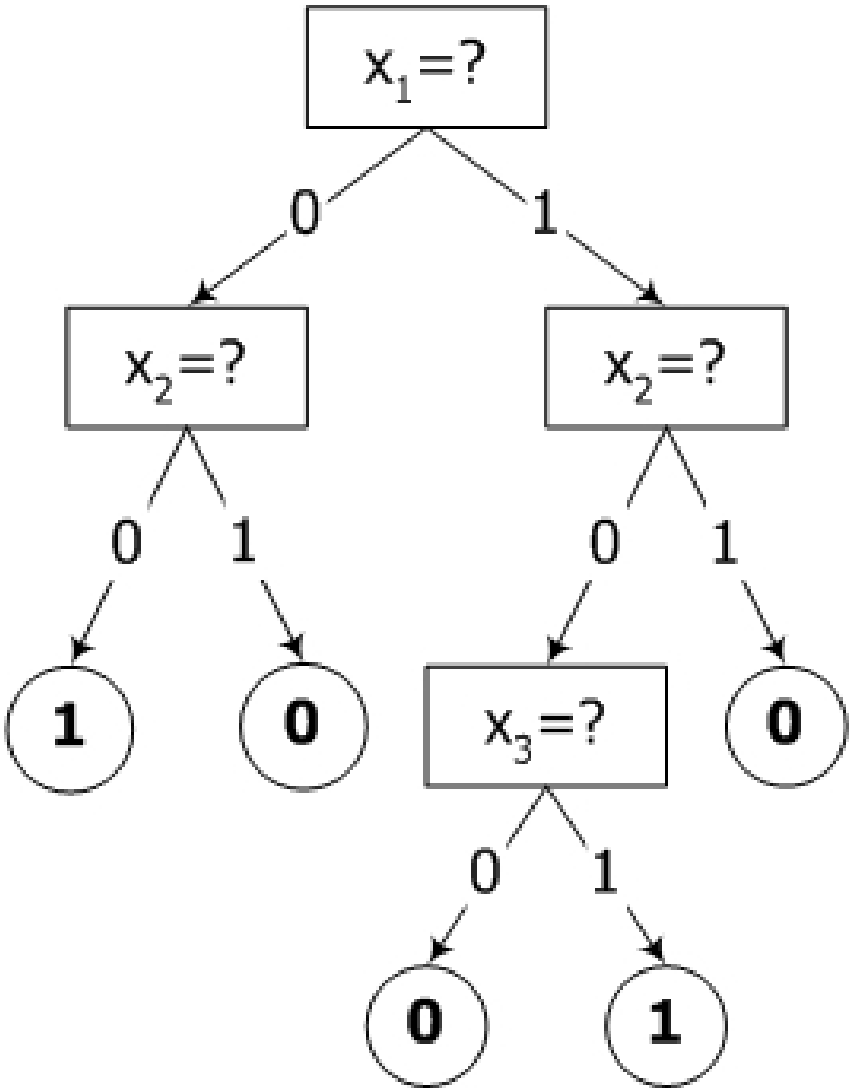
Представление гипотез

Обычно для кодирования гипотез используют по одному биту на каждое значение атрибута; если стоит 1, значит, это значение участвует в посылке гипотезы; значения одного и того же атрибута соединяются, конечно, как OR.

x_1	x_2	x_3	f
10	01	11	1

Строка 11 значит, что на этот атрибут можно не обращать внимания. А для целевой функции можно оставить один бит — гипотеза со следствием «будет какое-то значение» не имеет смысла.

Гипотезы из нескольких правил



До сих пор мы кодировали одно *правило* (одну ветку дерева). Но, например, это дерево соответствует нескольким правилам. Их мы будем записывать простой конкатенацией строк.

Упражнение

Выразить это дерево битовой строкой по нашей схеме.

Упражнения

Упражнение

Разработать схему кодирования для примера с играми «Зенита».

Упражнение

Реализовать процедуру подсчёта по правилу функции Fitness — квадрата отношения успешно пройденных тестовых примеров к их общему количеству.

Кроссовер

- При размножении особь должна унаследовать черты обоих предков. Как этого достичь на битовых строках?

Кроссовер

- При размножении особь должна унаследовать черты обоих предков. Как этого достичь на битовых строках?

- *Кроссовер* (crossover) — операция, которая по заданной маске делает из двух строк одну.
- Есть несколько разных видов кроссовера.

Виды кроссовера

- Single-point crossover:

Исходные строки	Маска	Результат
*10011*01011		
00101*01100*	1111100000	1001101100

Виды кроссовера

- Double-point crossover

Исходные строки	Маска	Результат
100*1101*011		
*001*0101*00*	0001111100	0010101100

Виды кроссовера

- Uniform crossover

Исходные строки	Маска	Результат
1*00*1*1*01*0*1*1*		
*0*010*1*01*1*00*0*	0110100110	0000101010

Кроссовер на строках переменной длины

Все эти виды — для строк одинаковой длины. Но наши гипотезы — из нескольких правил, и строки разной длины.

Что делать?

Кроссовер на строках переменной длины

Решение:

- Используем double-point crossover так, чтобы сохранять постоянное расстояние до краёв правил.
- Например, правила длины 5, и мы случайно выбрали две точки из гипотезы

$$0[0101\ 110]10.$$

- Тогда во второй гипотезе нужно выбирать такие точки, чтобы расстояние от левой точки до левого края правила и от правой точки до правого края правила были теми же. Например, в правиле длины 15 могут быть варианты:

1[10]11 01010 01110	1[1011 010]10 01110
1[1011 01010 011]10	11011 0[10]10 01110
11011 0[1010 011]10	11011 01010 0[11]10

Кроссовер на строках переменной длины

- Теперь кроссовер будет порождать корректные гипотезы:

Исходные строки	Результат
0[0101 110]10	01010
1[10]11 01010 01110	10101 11011 01010 01110

Мутации

Мутации на битовых строках:

- Изменить один случайный бит.
- Сделать несущественным (забить единичками) один случайный атрибут.
- ...

Fitness

- Функция Fitness должна зависеть от того, насколько хорошо гипотеза справляется с задачей.
- В случае задачи классификации разумная функция:

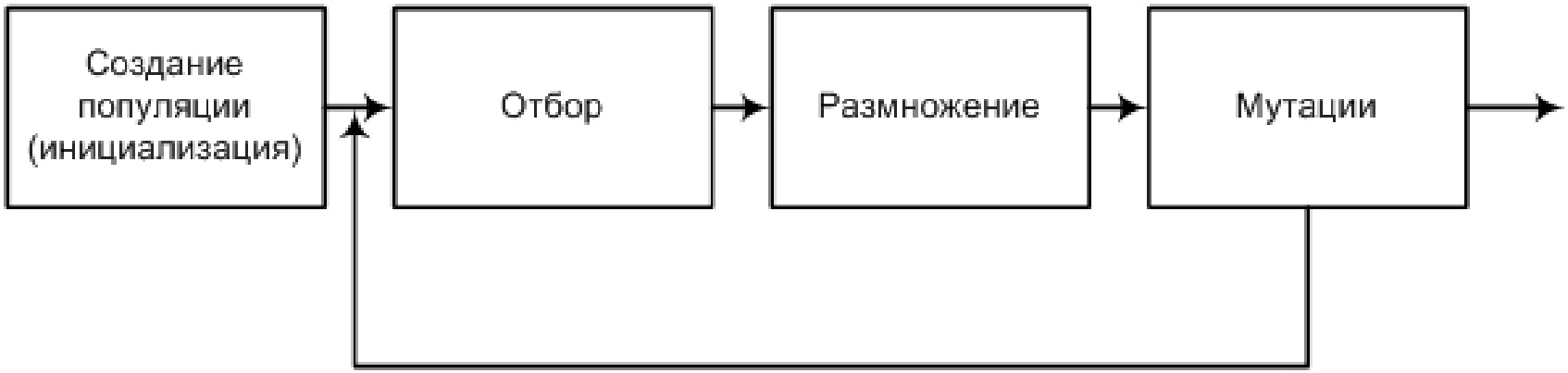
$$\text{Fitness}(h) = \left(\frac{\text{Correct}(h)}{\text{TotalExamples}} \right)^2,$$

где $\text{Correct}(h)$ — количество примеров, верно расклассифицированных гипотезой h, TotalExamples — общее количество примеров.

Outline

Вспомним общую схему

- Сгенерировать начальную популяцию.
- Пока не достигнуто значение, большее $\mathrm{Fitness}_{max}$:
  - Выбрать часть существующей популяции (отдавая предпочтение более приспособленным особям).
  - Применить к этой части генетические операции, породив потомков.
  - Подсчитать Fitness для особей новой популяции.



Что осталось

Мы уже научились:

- Делать кроссовер.
- Мутировать.
- Подсчитывать функцию Fitness.

Осталось

- Научиться выбирать самых приспособленных.

Выбор самых приспособленных

- Задаём долю выживших s — остальная часть популяции будет заменена на результаты кроссовера другой части.
- Нужно выбрать sN гипотез из популяции размера N.

- *Метод рулетки* (roulette wheel selection): у каждой гипотезы h_i вероятность быть выбранной

$$Pr(h_i) = \frac{\mathrm{Fitness}(h_i)}{\sum_{j=1}^{N} \mathrm{Fitness}(h_j)}$$

Выбор самых приспособленных

- Задаём долю выживших s — остальная часть популяции будет заменена на результаты кроссовера другой части.
- Нужно выбрать sN гипотез из популяции размера N.

- *Турнирный метод* (tournament selection): случайно выбираем две гипотезы. С фиксированной вероятностью p выживает более приспособленная, с вероятностью $1 - p$ — менее приспособленная.

Выбор самых приспособленных

- Задаём долю выживших s — остальная часть популяции будет заменена на результаты кроссовера другой части.
- Нужно выбрать sN гипотез из популяции размера N.

- *Ранговый метод* (ranking selection): сначала сортируем гипотезы по приспособленности. Затем как в методе рулетки, но вероятность выжить пропорциональна не значению $\mathrm{Fitness}(h)$, а месту, которое заняла гипотеза.

Выбор самых приспособленных

- Задаём долю выживших s — остальная часть популяции будет заменена на результаты кроссовера другой части.
- Нужно выбрать sN гипотез из популяции размера N.

- Мы будем использовать метод рулетки.

Алгоритм

$Genetic(N, p, s, m, \mathrm{Fitness}, \mathrm{Fitness}_{\max})$

- Создать N случайных гипотез $H = \{h_1, \ldots, h_n\}$.
- Для каждой гипотезы $h \in H$ вычислить $\mathrm{Fitness}(h)$.
- Пока $\max_h \mathrm{Fitness}(h) < \mathrm{Fitness}_{\max}$:
  - $H' = \emptyset$.
  - Случайно выбрать sN гипотез из H и добавить их в H'. Вероятность выбрать гипотезу h_i $Pr(h_i) = \frac{\mathrm{Fitness}(h_i)}{\sum_{j=1}^{N} \mathrm{Fitness}(h_j)}$.
  - Случайно выбрать $\frac{(1-s)p}{2}$ пар гипотез из H с теми же вероятностями. Для каждой пары (h_i, h_j) запустить операцию кроссовера и добавить её результат в H'.
  - Равномерно выбрать mN случайных гипотез из H' и в каждой из них инвертировать случайный бит.
  - $H = H'$.
  - Для каждой гипотезы $h \in H$ вычислить $\mathrm{Fitness}(h)$.

Упражнения

Упражнение

Реализовать генетический алгоритм, выбирающий гипотезу для объяснения игры «Зенита» из примера в первой лекции.

Спасибо за внимание!

- Lecture notes, слайды и коды программ появятся на моей homepage:

  `http://logic.pdmi.ras.ru/~sergey/index.php?page=teaching`

- Присылайте любые замечания, коды программ на других языках, решения упражнений, новые численные примеры и прочее по адресам:
  `sergey@logic.pdmi.ras.ru`, `smartnik@inbox.ru`