

Введение в искусственный интеллект

На базе дисциплины «Вычислительные сети,
системы и телекоммуникации»

Технологический университет
Королёв
2020

Введение в искусственный интеллект

Лекция №4 — «Эволюционные методы».

- Эволюционное моделирование
- Генетический алгоритм
- Нейроэволюция
- Деревья решений и случайный лес

Эволюционное моделирование

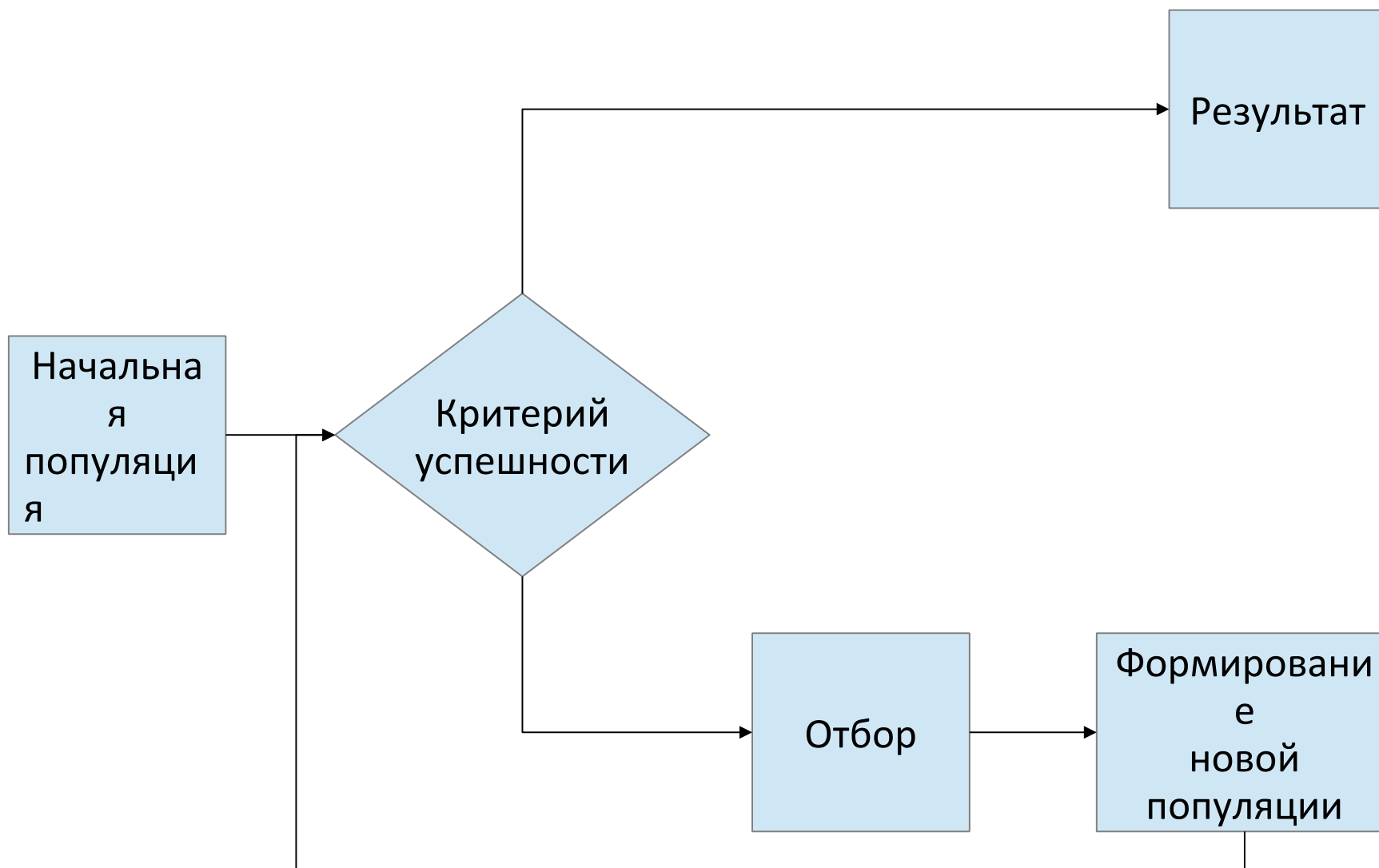
Рассматривается популяция объектов.

Цель популяции – оптимизация особой величины – приспособленности объектов.

По некоторым правилам происходит обновление популяции: часть объектов исключается, новые добавляются.

Новые объекты строятся путем изменения старых объектов с максимальной приспособленностью.

Эволюционное моделирование



Эволюционное моделирование

Каждый объект в популяции задается набором параметров, от которых зависит приспособленность, т.е.

Приспособленность = $f(\text{набор параметров})$

Эволюционное моделирование

Мутации

Каждый параметр x с некоторой вероятностью изменяется на небольшую величину:

$$x_{i+1} = x_i + a,$$

где $a \in N(0, \sigma^2)$, $|\sigma| \ll |x|$.

Эволюционное моделирование

Рекомбинация

Каждый параметр x с равной вероятностью берется от одного из родителей:

$$X_{i+1} = Y_i \text{ при } a < 0,5,$$

$$X_{i+1} = Z_i \text{ при } a \geq 0,5,$$

где $a \in (0,1)$;

y и z – параметры родительских объектов.

Генетический алгоритм

Объекты $X = (a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8)$, где $a_j \in \{0, 1\}$.

Цель $Y = (1, 1, 1, 1, 1, 1, 1, 1)$.

Приспособленность: $f(X) = \sum |X_i - Y_i| = \sum |X_i - 1|$

Генетический алгоритм

Пример начальной популяции из 4 объектов:

$$X_1 = (0,0,0,0,1,1,1,1), \quad f(X_1) = 4$$

$$X_2 = (0,1,0,1,0,1,0,1), \quad f(X_2) = 4$$

$$X_3 = (0,0,0,0,0,0,0,0), \quad f(X_3) = 8$$

$$X_4 = (1,1,0,0,0,0,1,1), \quad f(X_4) = 4$$

Генетический алгоритм

Отбор

$$X_1 = (0,0,0,0,1,1,1,1), \quad f(X_1) = 4$$

$$X_2 = (0,1,0,1,0,1,0,1), \quad f(X_2) = 4$$

$$X_3 = (0,0,0,0,0,0,0,0), \quad f(X_3) = 8$$

$$X_4 = (1,1,0,0,0,0,1,1), \quad f(X_4) = 4$$

Генетический алгоритм

Рекомбинация

Формируем 4 пары: (X_1, X_2) , (X_1, X_4) , (X_2, X_4) , (X_1, X_2) .

На примере пары (X_1, X_2) :

X_1	0	0	0	0	1	1	1	1
X_2	0	1	0	1	0	1	0	1
X_5	0	0	0	1	0	1	1	1

$$X_5 = (0, 0, 0, 1, 0, 1, 1, 1)$$

Генетический алгоритм

Мутации

Берем получившийся объект:

$$X_5 = (0,0,0,1,0,1,1,1)$$

Случайным образом выбираем небольшое количество элементов:

$$X_5 = (0,0,0,1,0,1,1,1)$$

Изменяем значения выбранных элементов:

$$X_5 = (0,0,1,1,0,1,1,1)$$

$$f(X_5) = 5$$

Генетический алгоритм

Плюсы:

- Хорошо распараллеливается.
- Быстро сходится.

Минусы:

- Часто сходится к локальному экстремуму.
- Много настроечных параметров.

Нейроэволюция

Эволюционное моделирование, в том числе генетический алгоритм, могут быть применены к подбору параметров нейронной сети.

Это может быть удобно в ситуации, когда входные данные не размечены (как в обучении с учителем), но мы можем тем или иным образом вычислить приспособленность. В таком случае это один из видов обучения без учителя.

Нейроэволюция

Кодирование параметров:

В качестве элементов объекта в популяции используются веса связей между нейронами.

Кодирование структуры:

В качестве элементов объекта в популяции используется информация о количестве слоев и нейронов, об активационных функциях и т.д.

Нейроэволюция

Прямое кодирование:

В параметрах объекта находится непосредственно информация о весах или настройках сети.

Непрямое кодирование:

В параметрах объекта находятся информация о настройках алгоритма формирования сети.

Деревья решений

Дерево решений – визуальная модель алгоритма принятия решения.

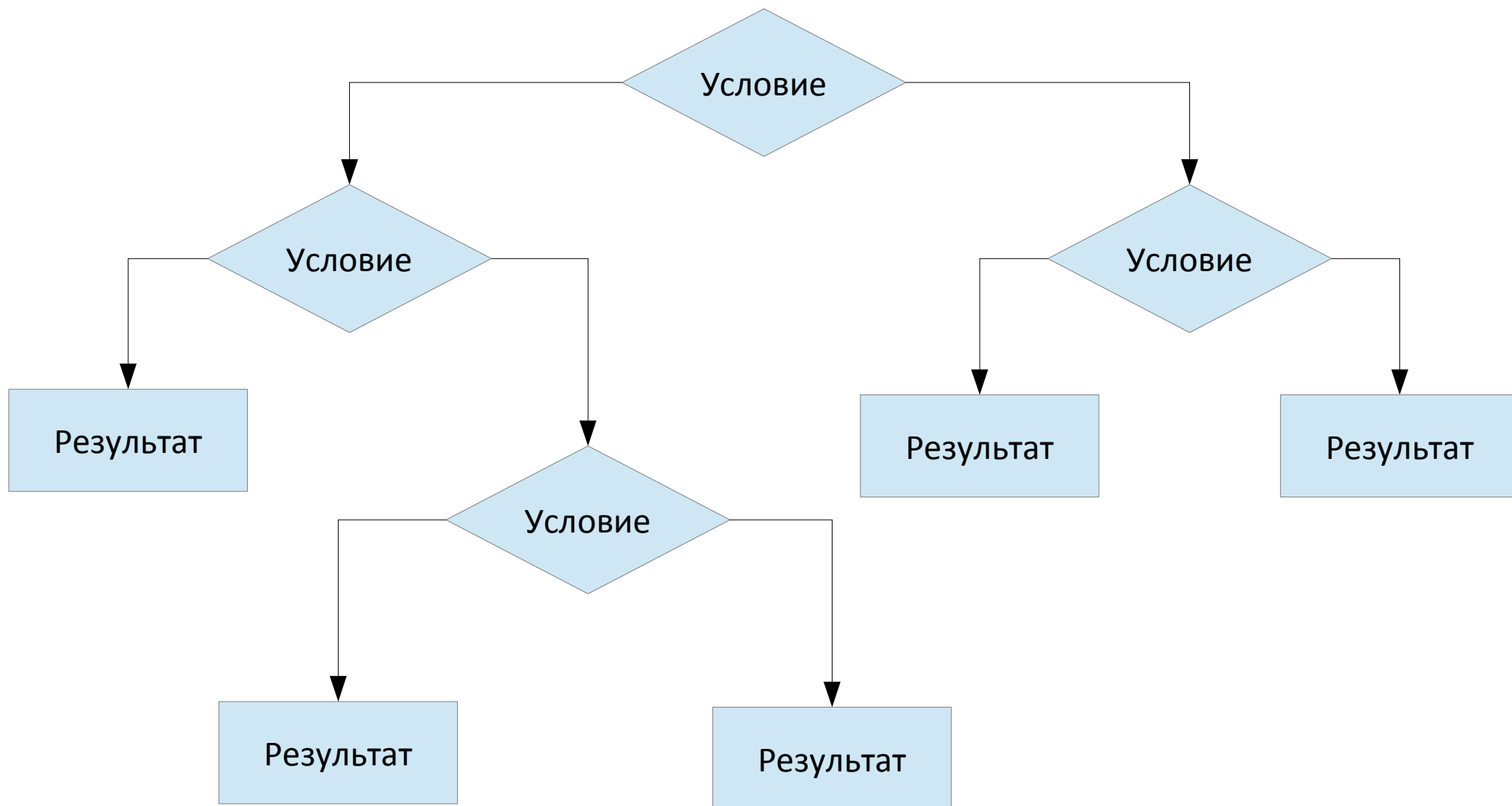
Элементы:

- Узлы — задают условия выбора ветви.
- Листья — задают итоговые значения.

Задачи:

- Классификация
- Регрессия

Деревья решений



Случайный лес

Основная идея – использование большого количества решающих деревьев, каждое из которых даёт «плохой» результат, но в целом результат получается «хорошим».

Алгоритм является одним из видов обучения с учителем.

Случайный лес

Рассмотрим обучающую выборку с N записями, каждая из которых задана M параметрами.

Бэггинг:

1. Создаем подвыборку с повторениями размером N из обучающей выборки.
2. Случайно выбираем $m < M$ параметров.
3. Любым алгоритмом строим дерево решений.
4. Дерево добавляется в «случайный лес».

Случайный лес

Способ работы случайного леса: каждое дерево принимает решение (определяет класс), затем все деревья голосуют и финальным решением является то, за которое отдано больше всего голосов.

Количество деревьев в случайном лесу выбирается так, чтобы минимизировать ошибку на тестовой выборке.

Материалы

1.<https://basegroup.ru/community/articles/ga-math>

2.<https://habr.com/ru/post/345950/>

3.<https://dyakonov.org/2016/11/14/случайный-лес-rand>

Спасибо за внимание!