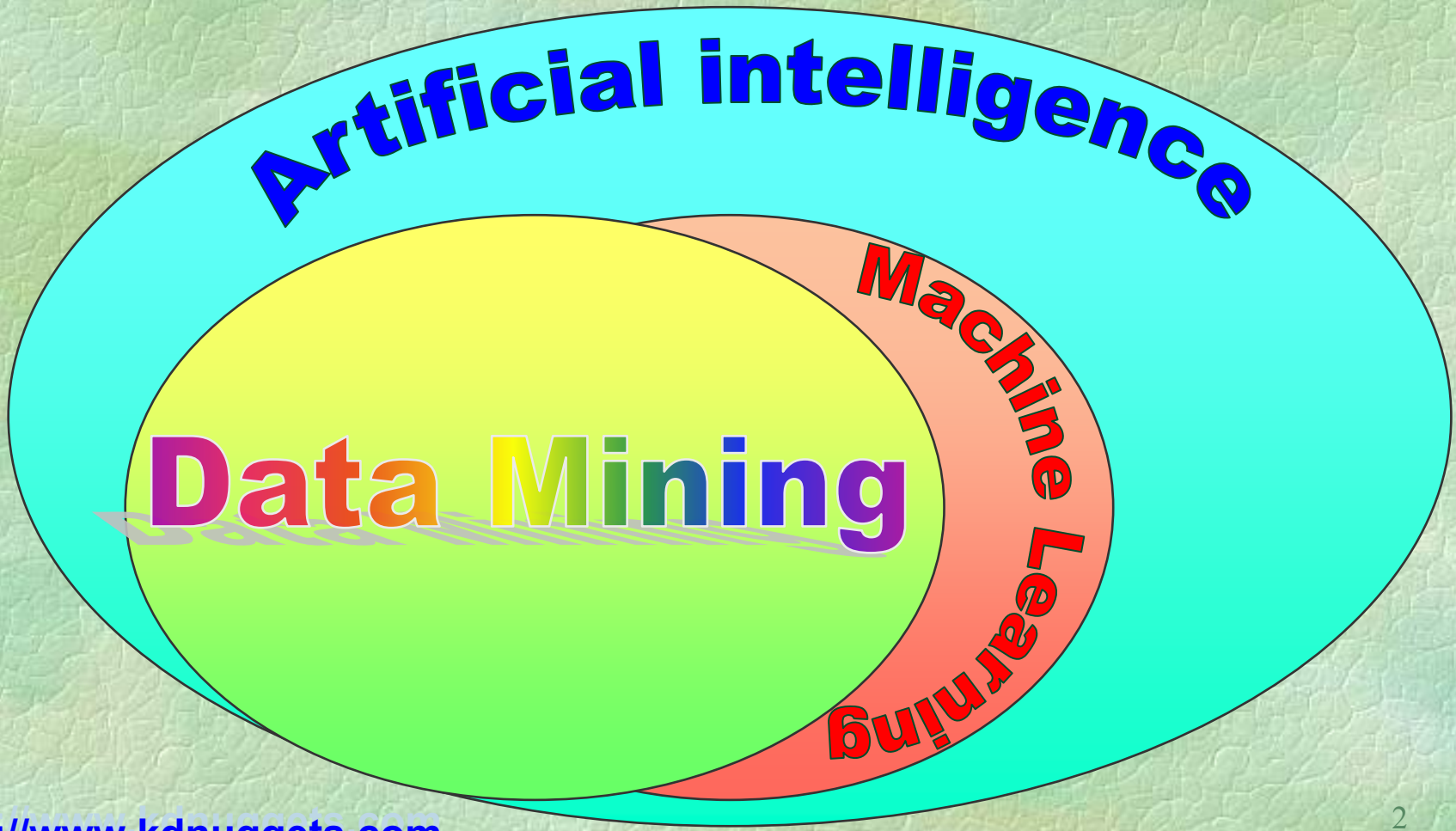
Machine Learning

Хомичёва Ирина Вадимовна
аспирант лаборатории Теоретическая генетика

Artificial intelligence, Machine Learning, Data Mining



http://www.kdnuggets.com

Искусственный интеллект

Интеллектом называется способность мозга решать (интеллектуальные) задачи путем приобретения, запоминания и целенаправленного преобразования знаний в процессе обучения на опыте и адаптации к разнообразным обстоятельствам.

В этом определении под термином "знания" подразумевается не только та информация, которая поступает в мозг через органы чувств. Такого типа знания чрезвычайно важны, но недостаточны для интеллектуальной деятельности. Дело в том, что объекты окружающей нас среды обладают свойством не только воздействовать на органы чувств, но и находиться друг с другом в определенных отношениях. Ясно, что для того, чтобы осуществлять в окружающей среде интеллектуальную деятельность (или хотя бы просто существовать), необходимо иметь в системе знаний модель этого мира. В этой информационной модели окружающей среды реальные объекты, их свойства и отношения между ними не только отображаются и запоминаются, но и, как это отмечено в данном определении интеллекта, могут мысленно "целенаправленно преобразовываться". При этом существенно то, что формирование модели внешней среды происходит "в процессе обучения на опыте и адаптации к разнообразным обстоятельствам".

Machine Learning

Определение. Будем говорить, что программа обучается на эксперименте **E** в классе задач **T** при данной мере выполнения **P**, если её выполнение задач **T** улучшается (в смысле **P**) с использованием опыта **E**

E.g., Learn to play checkers

- T: Play checkers
- P: % of games won in world tournament
- E: opportunity to play against self

Mitchell, T. (1997). *Machine Learning.*

Примеры задач **T**, мер выполнения **P** и экспериментов **E**

- T: Recognizing hand-written words
  P: Percent of words correctly classified
  E: Database of classified handwritten words

- T: Driving four-lane highways using vision sensors
  P: Average distance travelled before human-judged error
  E: A sequence of images and steering commands recorded while observing a human driver.

- T: Diagnosis patients into fixed set of diseases
  P: Percentage of patients diagnosed correctly
  E: Database of patient records with doctor-given diagnoses.

Knowledge Discovery in Databases &Data Mining

" обнаружение знаний в базах данных"

"интеллектуальный анализ данных"

“раскопка” данных

Выделяют пять типов закономерностей, которые позволяют выявлять методы KDD&DM: ***ассоциация***, ***последовательность***, ***классификация***, ***кластеризация*** и ***прогнозирование***.

Ассоциация имеет место в том случае, если несколько событий связаны друг с другом. Если существует цепочка связанных во времени событий, то говорят о ***последовательности***. С помощью ***классификации*** выявляются признаки, характеризующие группу, к которой принадлежит тот или иной объект. Это делается посредством анализа уже классифицированных объектов и формулирования некоторого набора правил. ***Кластеризация*** отличается от классификации тем, что сами группы заранее не заданы. С помощью кластеризации средства Data Mining самостоятельно выделяют различные однородные группы данных. Основой для всевозможных систем ***прогнозирования*** служит историческая информация, хранящаяся в БД в виде временных рядов. Если удается построить, найти шаблоны, адекватно отражающие динамику поведения целевых показателей, есть вероятность, что с их помощью можно предсказать и поведение системы в будущем.

Data Mining приложения

Розничная торговля

- *анализ покупательской корзины* (анализ сходства) предназначен для выявления товаров, которые покупатели стремятся приобретать вместе.
- *исследование временных шаблонов* помогает торговым предприятиям принимать решения о создании товарных запасов. Оно дает ответы на вопросы типа "Если сегодня покупатель приобрел видеокамеру, то через какое время он вероятнее всего купит новые батарейки и пленку?"
- *создание прогнозирующих моделей* дает возможность торговым предприятиям узнавать характер потребностей различных категорий клиентов с определенным поведением.

Банковское дело

- *выявление мошенничества с кредитными карточками*. Путем анализа прошлых транзакций, которые впоследствии оказались мошенническими, банк способен выявить некоторые стереотипы такого мошенничества. Например, можно установить, что одним из предупреждающих сигналов служат многочисленные транзакции в магазинах бытовой электроники в течение короткого периода времени. Полученное знание банк может использовать в своих действующих системах, не разрешая подтверждение транзакции, совпадающей со стереотипом мошенничества, без предварительной беседы с покупателем.

Data Mining приложения

Страхование

- *анализ риска*. Путем выявления сочетаний факторов, связанных с оплаченными заявлениями, страховщики могут уменьшить свои потери по обязательствам. Известен случай, когда в США крупная страховая компания проверила заявления, по которым были выплачены значительные суммы за последние два года. При этом обнаружилось, что суммы, выплаченные по заявлениям людей, состоящих в браке, вдвое превышает суммы по заявлениям одиноких людей. Компания отреагировала на это новое знание пересмотром своей общей политики предоставления скидок семейным клиентам.

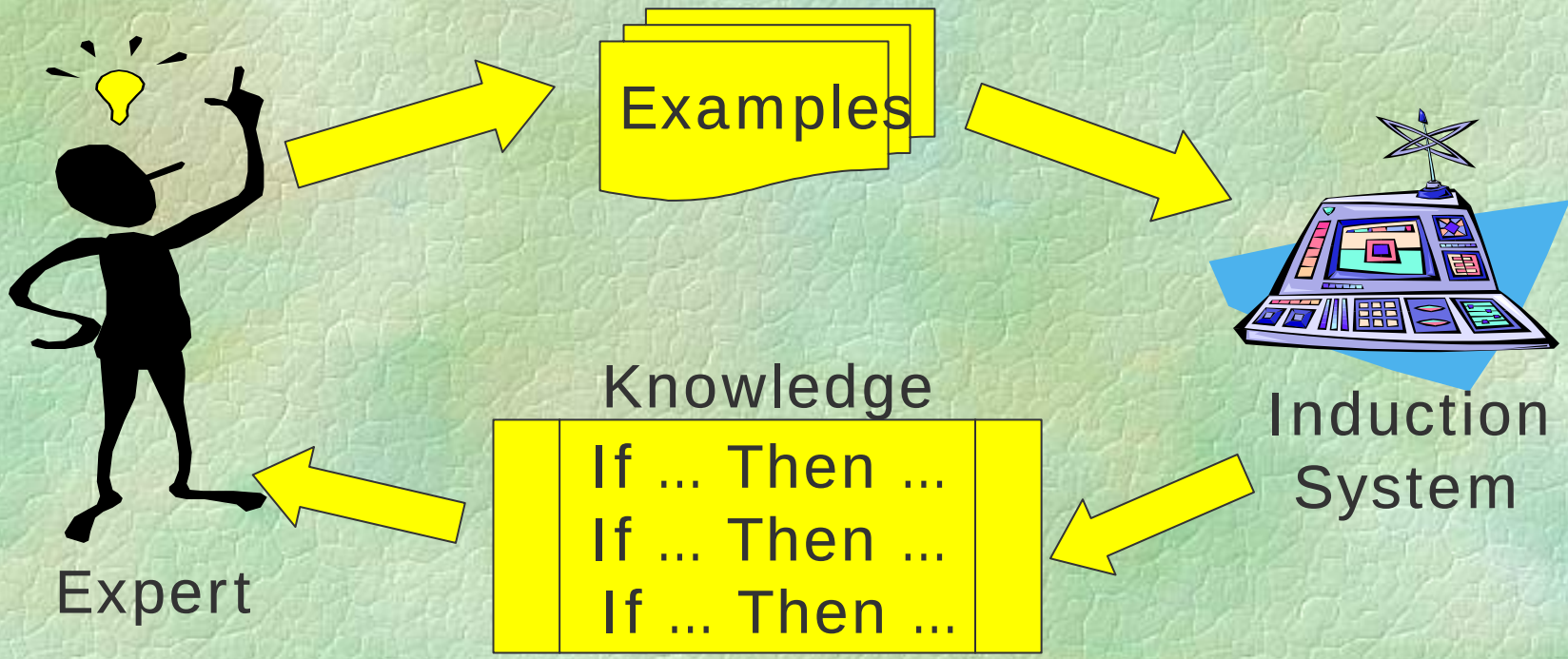
Другие приложения в бизнесе

- *сегментация рынка*. Все отрасли могут воспользоваться методами Data Mining для выявления отдельных сегментов своей клиентуры. Data Mining дает предприятиям возможность учитывать намного больше параметров, чем это делалось на основе традиционных методов хранения неструктурированной информации;
- *развитие автомобильной промышленности*. При сборке автомобилей производители начинают учитывать требования каждого отдельного клиента, поэтому им нужны возможности прогнозирования популярности определенных характеристик и знание того, какие характеристики обычно заказываются вместе;
- *политика гарантий*. Производителям нужно предсказывать число клиентов, которые подадут гарантийные заявки, и среднюю стоимость заявок;
- *поощрение часто летающих клиентов*. Авиакомпании могут обнаружить группу клиентов, которых данными поощрительными мерами можно побудить летать больше.

Извлечение знаний

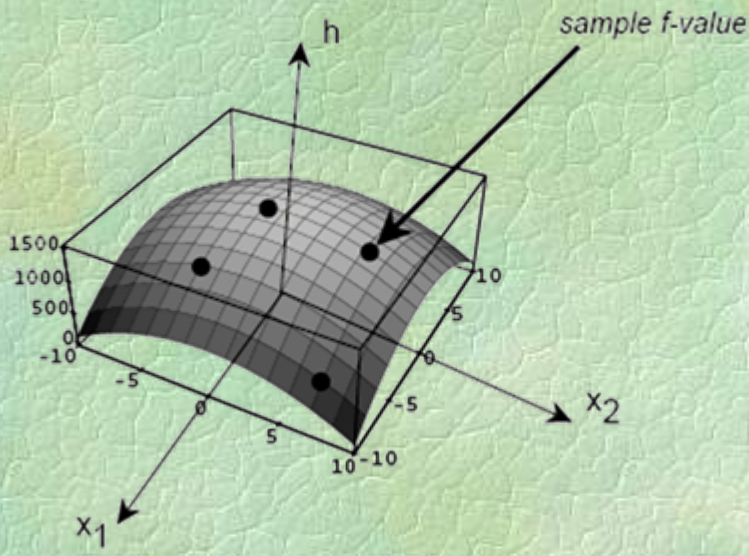
Decision tree
Genetic algorithm

Neural network
Bayesian network

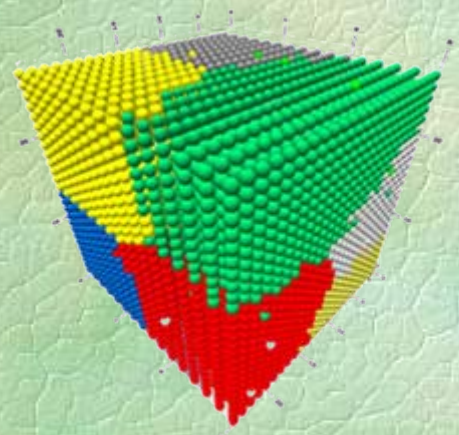


Основные парадигмы обучения

Обучение с учителем (supervised learning).
При таком типе обучения известны значения аппроксимируемой функции на некотором ограниченном наборе данных. Предположим, нам известны значения двумерной функции f в четырёх точках. По этим точкам восстановим параболическую поверхность h такую, что $h \approx f$ для этих точек. В таком случае, поверхность h является нашей гипотезой относительно f.



Обучение без учителя (unsupervised learning).
При данной парадигме обучения значения целевой функции априори не известны. Обучающие данные представлены векторами в многомерном пространстве, требуется выделить подмножества векторов, в соответствии с некоторым принципом.



Постановка задачи

Допустим, требуется описать концепцию “дни, благоприятные для игры в теннис”

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes

• • •

Признаки Значения

outlook = {Sunny, Overcast, Rain}
temperature = {Hot, Mild, Cool}
humidity = {High, Normal}
wind = {Weak, Strong}

For
Outlook=Sunny,
Humidity=Normal,
Windy=Strong
PlayTennis = ?

Постановка задачи в общем виде

Формально, пусть имеется $X = \langle X_1, ..., X_n \rangle$-набор m-мерных векторов (n – количество обучающих примеров, m - количество признаков).

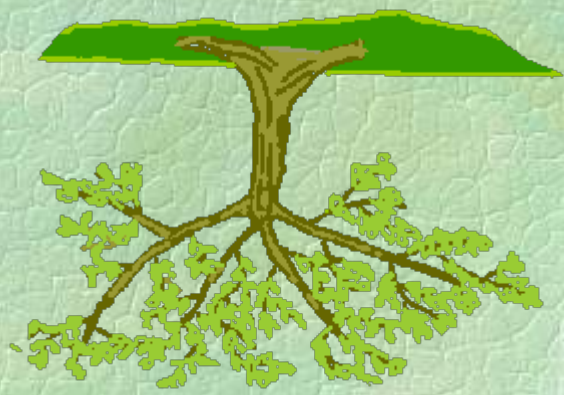
Целевая концепция $\text{PlayTennis} : X \to \langle Yes, No \rangle$ представлена конечным набором реализаций $D = \langle X_i, PlayTennis(X_i) \rangle, i = 1...n$ - множество обучающих данных.

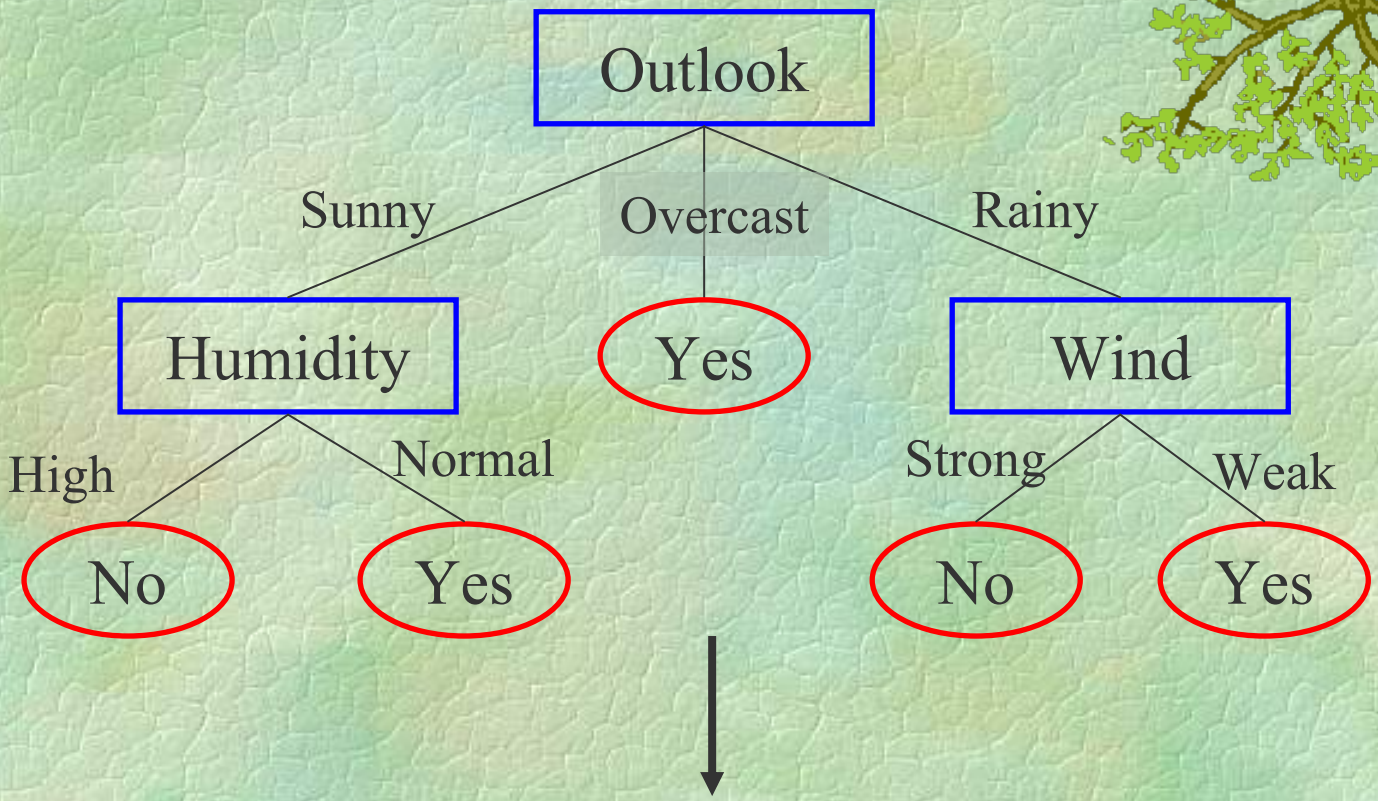
Требуется найти такую гипотезу $h : X \to \langle Yes, No \rangle$, $h \in H$, что для $\forall X_i \;\; h(X_i) = PlayTennis(X_i), i = 1...n$.

H – пространство решающих деревьев

Дерево решений

Определение. Деревом решений называется дизъюнкция коньюнкций значений признаков.





(Outlook = Sunny **AND** Humidity = Normal) **OR**
(Outlook = Overcast) **OR**
(Outlook = Rain **AND** Wind = Weak)

Алгоритм ID3

Осуществляет направленный поиск (greedy search) в пространстве возможных деревьев.

Конструирует дерево последовательно, от корня к листьям, каждый раз отбирая признак, который наилучшим образом (в смысле информационной меры) разделяет обучающие примеры.

Без обратного хода, т.е. не возвращается, не меняет локального решения.

Ross Quinlan (1975)

Сведения из теории информации
Энтропия

Информация есть устранённая неопределённость для достижения цели.

Энтропия (информационная) — мера хаотичности информации, неопределённость появления какого-либо символа первичного алфавита.

Информационная энтропия для независимых случайных событий *x* с *n* возможными состояниями (от 1 до *n*) рассчитывается по формуле:

$$H(x) = -\sum_{i=1}^{n} p(i) \log_2 p(i)$$

$p(i)$ - относительная частота появления события i

Смысл энтропии Шеннона

Рассмотрим пример (скачки).
В заезде участвуют 4 лошади с равными шансами на победу (1/4).

Введем д.с.в. X – номер победившей лошади.

Тогда HX = 2. После каждого заезда по каналу связи достаточно будет передать 2 бита информации о номере победившей лошади. Кодируем номер лошади следующим образом: 1—00, 2—01, 3—10, 4—11.

Если ввести функцию L(X), которая возвращает длину сообщения, кодирующего заданное значение X, то математическое ожидание ML(X) – это средняя длина сообщения, кодирующего X. Можно формально определить L через две функции.

$$L(X) = \text{len}(\text{code}(X)),$$

где code(X) каждому значению X ставит в соответствие некоторый битовый код, причем взаимно-однозначно. А len возвращает длину в битах для любого конкретного кода. В этом примере ML(X) = HX.

Смысл энтропии Шеннона

Пусть теперь д.с.в. имеет следующее распределение:

$$P(X = 1) = 3/4,\ P(X = 2) = 1/8,\ P(X = 3) = P(X = 4) = 1/16,$$

т.е. лошадь с номером 1 – это фаворит. Тогда:

$$HX = \frac{3}{4}\log_2\frac{4}{3} + \frac{1}{8}\log_2 8 + \frac{1}{8}\log_2 16 = \frac{19}{8} - \frac{3}{4}\log_2 3 \approx 1.186 \text{ бит/сим.}$$

Закодируем номера лошадей: 1—0, 2—10, 3—110, 4—111.

В среднем, в 16 заездах 1-ая лошадь должна победить в 12, 2-ая – в 2, 3-я и 4-ая – в одном. Таким образом, средняя длина сообщения о победителе равна:

$$(1*12 + 2*2 + 3*1 + 3*1)/16 = 1.375 \text{ бит/сим.},$$

Или М.О. L(X). Действительно, L(X) сейчас задается следующим распределением вероятностей:

$$P(L(X) = 1) = 3/4,\ P(L(X) = 2) = 1/8,\ P(L(X) = 3) = 1/8.$$

Следовательно,

$$ML(X) = 3/4+2/8+3/8=11/8= 1.375 \text{ бит/сим.}$$

Итак,

$$ML(X) > HX.$$

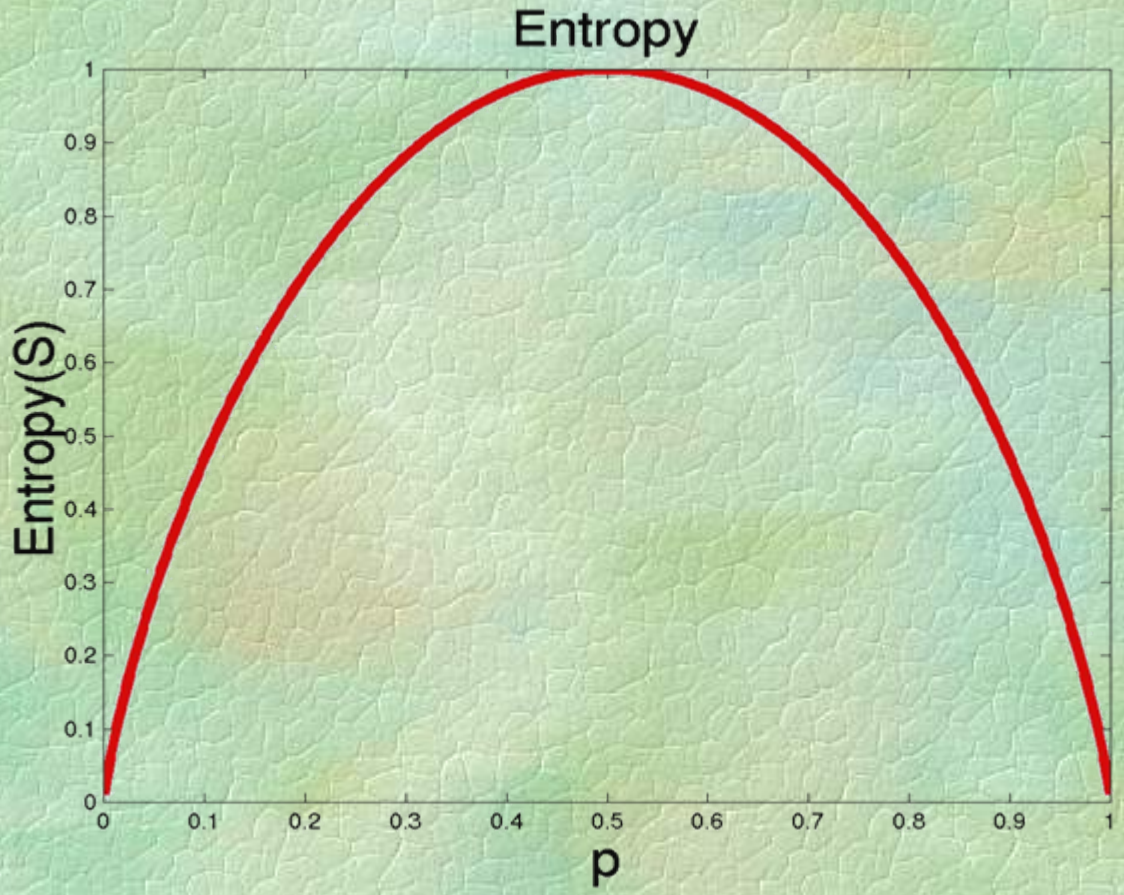
Можно доказать, что более эффективного кодирования для двух рассмотренных случаев не существует.

Энтропия д.с.в. – это минимум среднего количества бит, которое нужно передавать по каналу связи о текущем значении данной д.с.в.

Смысл энтропии Шеннона

То, что энтропия Шеннона соответствует интуитивному представлению о мере информации, может быть продемонстрировано в опыте по определению среднего времени психических реакций. Опыт заключается в том, что перед испытуемым человеком загорается одна из *N* лампочек, которую он должен указать. Проводится большая серия испытаний, в которых каждая лампочка зажигается с определенной вероятностью p_i ($\sum_{i=1}^{N} p_i = 1$), где *i* – это номер лампочки. Оказывается, среднее время, необходимое для правильно ответа испытуемого, пропорционально величине энтропии $-\sum_{i=1}^{N} p_i \log_2 p_i$, а не числу лампочек *N*, как можно было бы подумать. В этом опыте предполагается, что чем больше информации будет получено человеком, тем дольше будет время её обработки, соответственно, реакции на неё.

Энтропия



S - набор обучающих примеров, $S \subseteq D$

p_+ - часть позитивных примеров в S

p_- - часть негативных примеров в S

$$Entropy(S) = -p_+ \log_2(p_+) - p_- \log_2(p_-)$$

Обучающие данные

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Информационная мера признака

$$\text{Gain}(S,X) = \text{Entropy}(S) - \sum_{v \in \text{values}(X)} |S_v|/|S| \; \text{Entropy}(S_v)$$

values(X) - набор возможных значений признака X

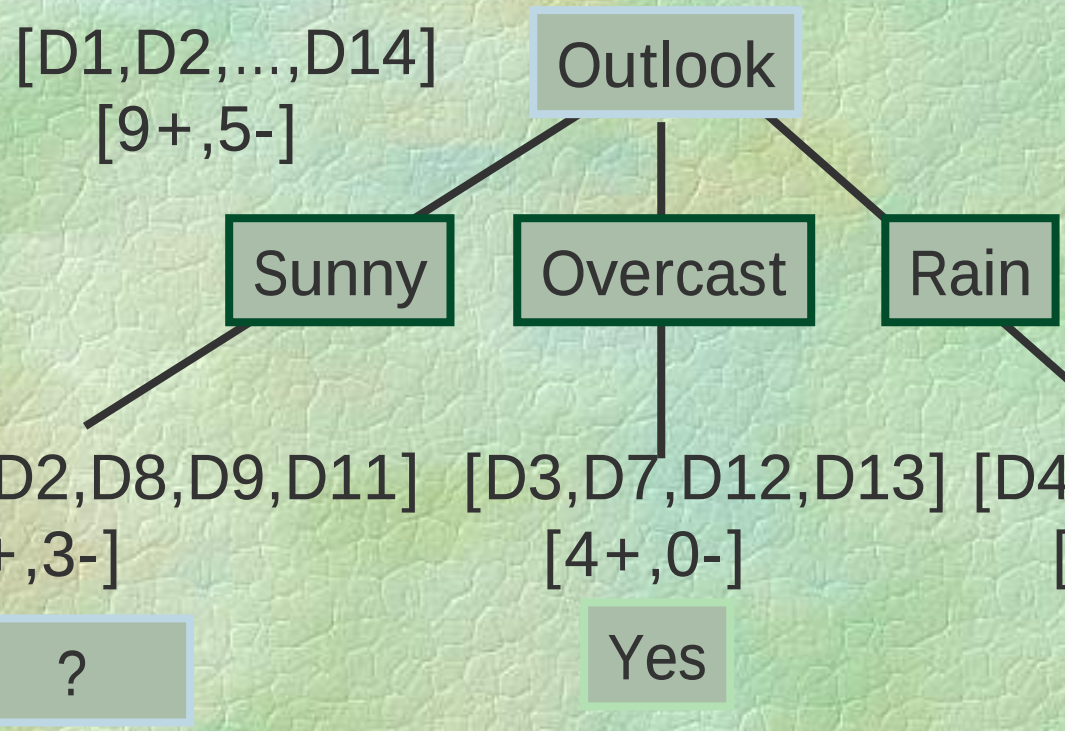
$$S_v = \{ s \in S \mid X(s) = v \}$$

такие примеры из S , для которых значение признака X равно v

Gain(S,X) – это количество бит информации о значении целевой функции при известных значениях признака X. Определим корневой признак дерева:

- Gain(S, Outlook) = 0.246
- Gain(S, Temperature) = 0.029
- Gain(S, Humidity) = 0.151
- Gain(S, Wind) = 0.048

Алгоритм ID3



Gain(S_{sunny} , Humidity)=0.970-(3/5)0.0 – 2/5(0.0) = 0.970

Gain(S_{sunny} , Temp.)=0.970-(2/5)0.0 –2/5(1.0)-(1/5)0.0 = 0.570

Gain(S_{sunny} , Wind)=0.970= -(2/5)1.0 – 3/5(0.918) = 0.019

Априорные предположения алгоритма ID3

Методологический принцип “бритвы Оккама”

“Сущности не должны быть умножены без необходимости”

Современная интерпретация

“Предпочесть наиболее простую гипотезу, которая удовлетворяет данным”

Принцип “бритвы Оккама” применительно к алгоритму ID3

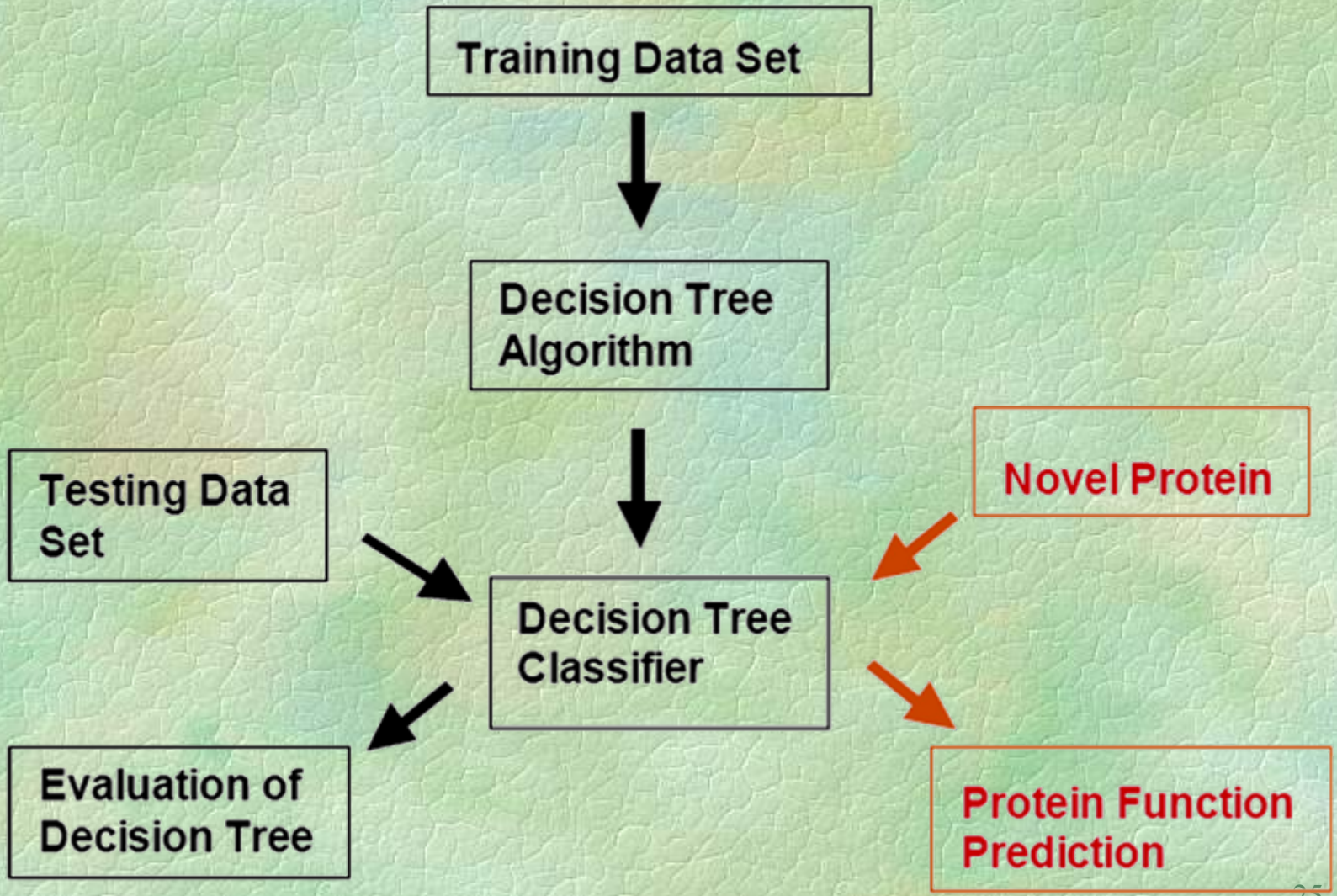
Поиск решения алгоритмом ID3 осуществляется в пространстве деревьев в условиях индуктивного предположения: “предпочесть более короткие деревья, листья которых ближе к корню обладают большей информационной мерой”

Data-Driven Generation of Decision Trees for Motif-Based Assignment of Protein Sequences to Functional Families

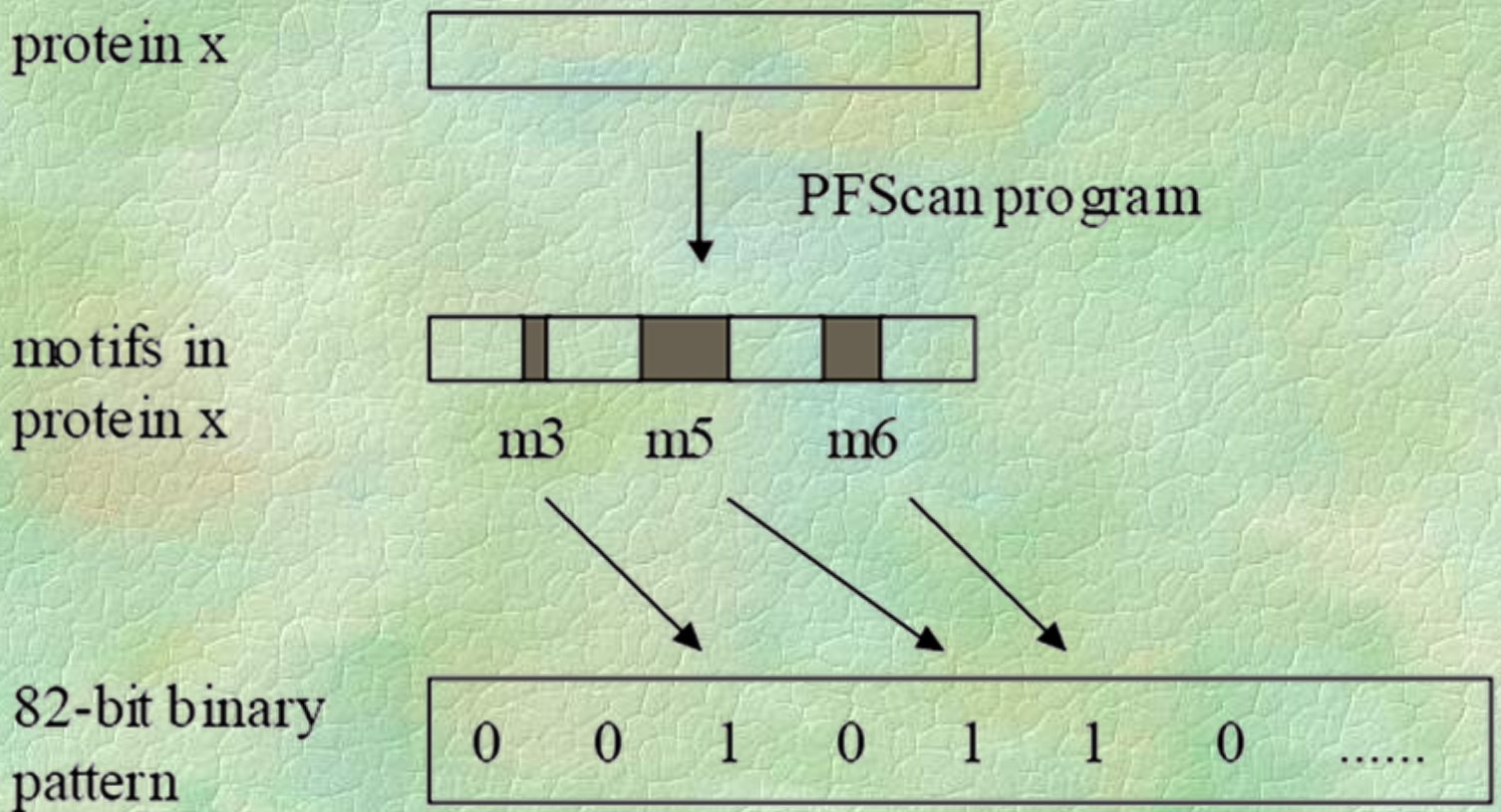
Dake Wang, Xiangyun Wang, and Vasant Honavar

Предсказание функций неизвестных белков является важнейшей задачей функциональной геномики. Функции белков коррелируют с высоко-консервативными участками аминокислот, мотивами. Созданные базы экспериментальных данных содержат мотивы (кластеры мотивов) и при запросе определённого мотива способны возвращать функцию, ассоциированную с этим мотивом. Однако такая модель является слишком упрощенной, т.к. многие белки содержат несколько мотивов, и, в общем случае, функция белка определяется паттерном мотивов, присутствием/отсутствием определенных мотивов.

Модель испытаний



Представление данных



Построение деревьев

Обучающие данные содержали 585 белков, принадлежащих к 10 известным функциональным классам, и одному негативному классу (false positive). Решающие деревья строились для разных объёмов обучающих данных (11, 20…585 последовательностей). Каждый эксперимент, для фиксированного объёма данных, проводился три раза, весь объём данных случайным образом делился на обучение и контроль. Усреднённая точность (для 3-х независимых запусков) решающих деревьев, в зависимости от объёма обучающих данных, представлена в таблице.

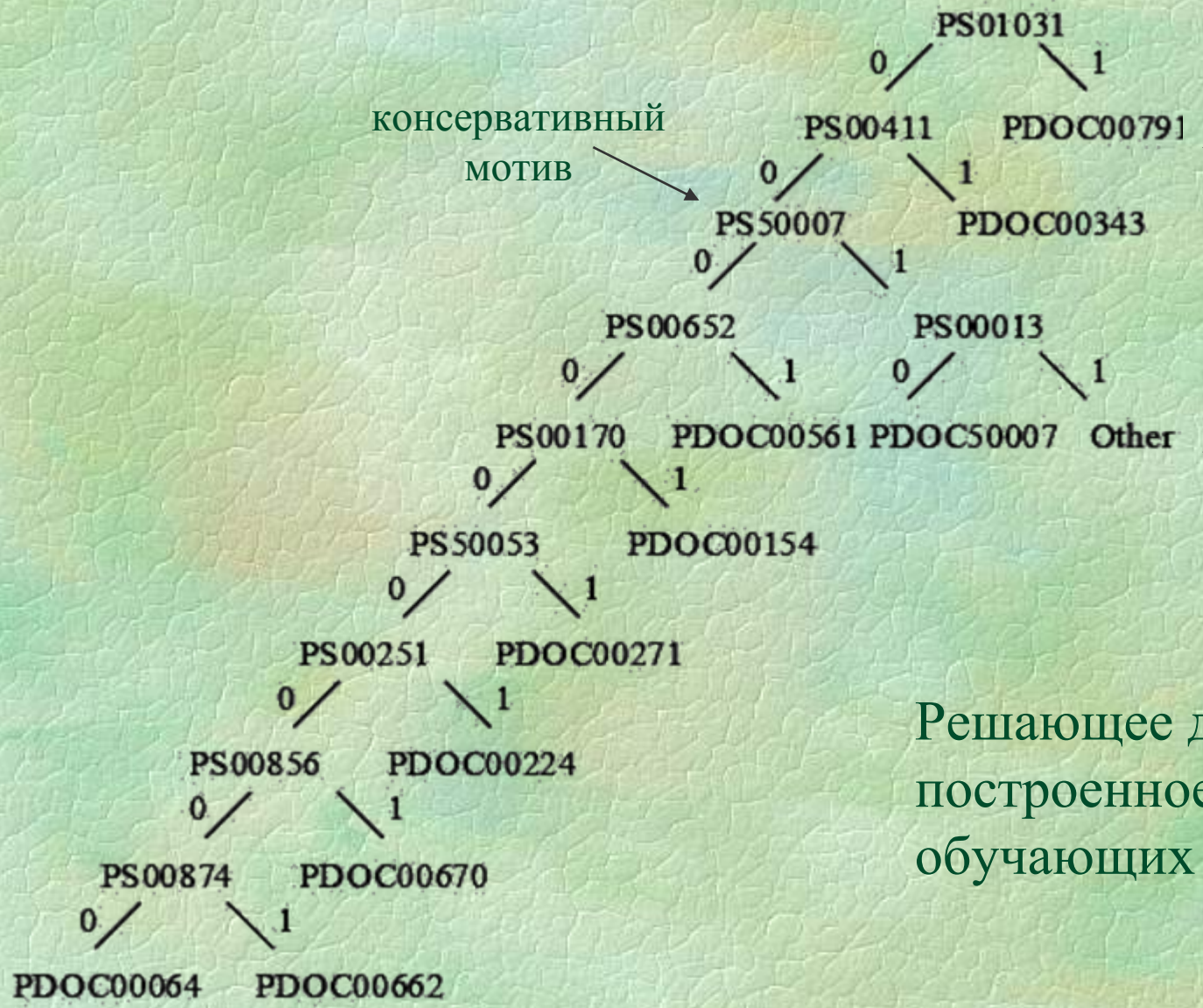
10%

Table 1. Effect of training set size on classification accuracy*.

Training set size	11	20	29	58	117	175	234	294	351	585
Accuracy (%)	59.1	71.7	82.7	94.0	95.1	96.5	98.4	99.0	98.7	99.8

* The results are the average of three independent runs.

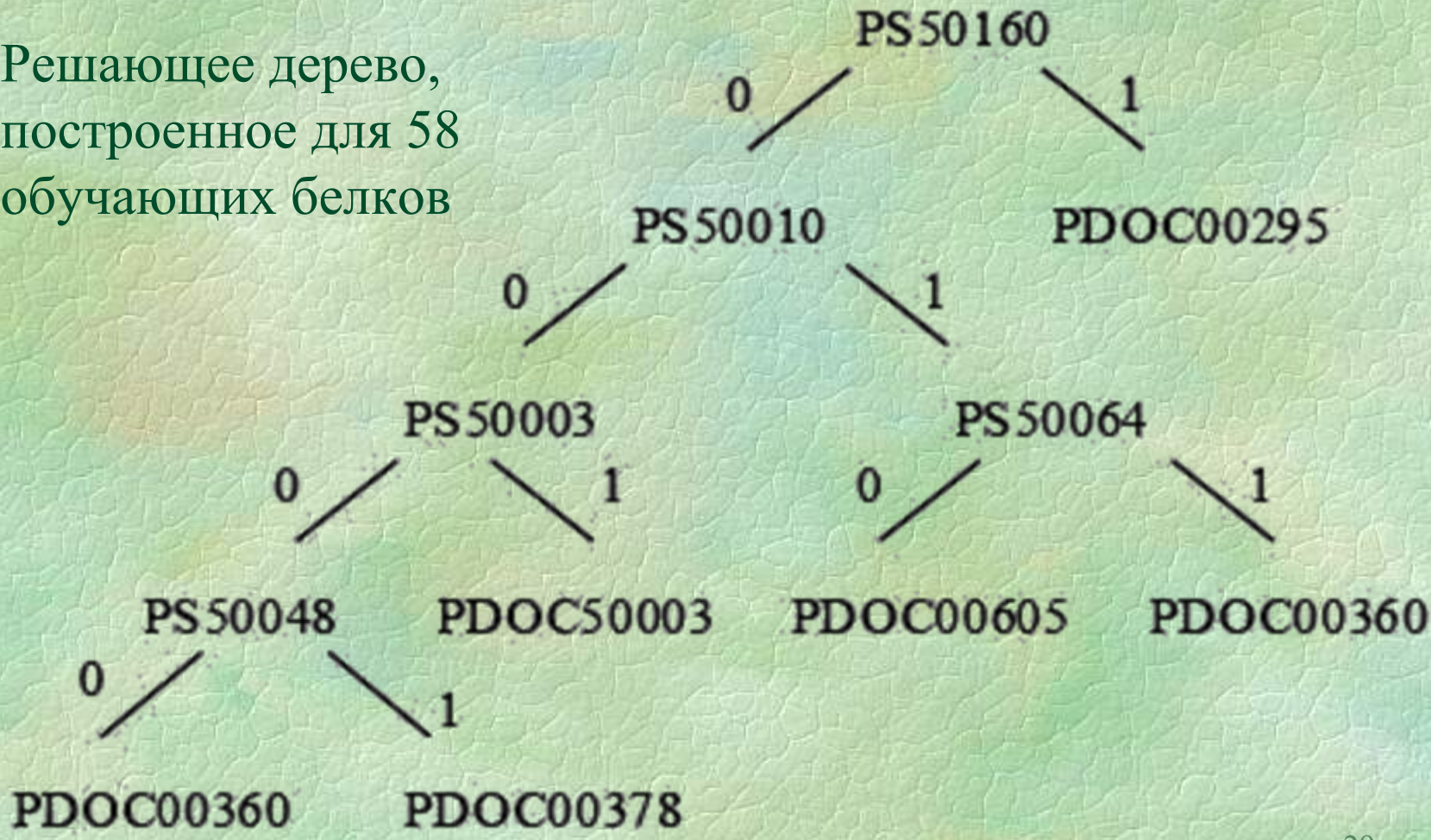
Построение деревьев



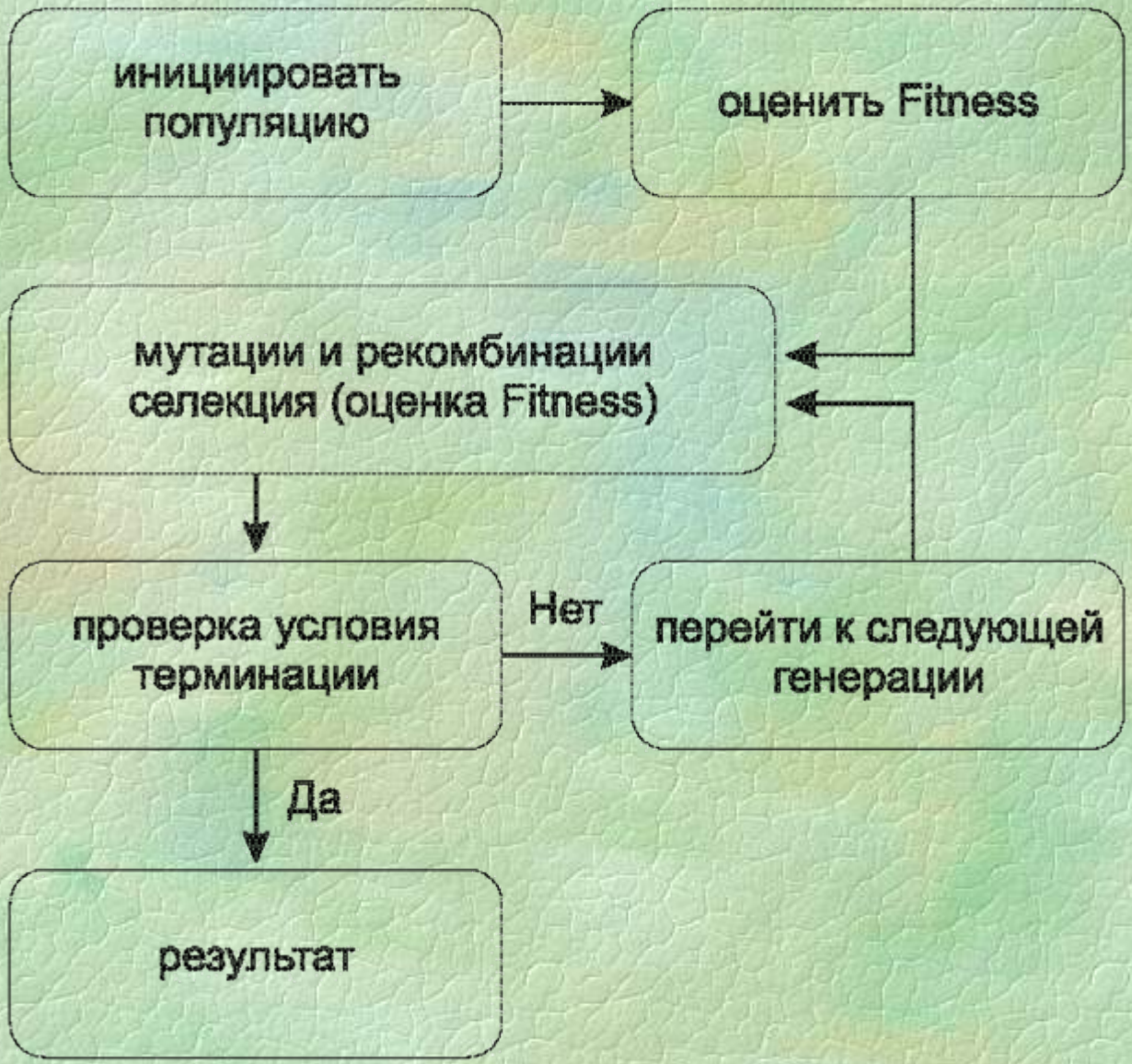
Решающее дерево, построенное для 58 обучающих белков

Построение решающих деревьев для белковых семейств, содержащих общие мотивы

Решающее дерево, построенное для 58 обучающих белков



Генетический алгоритм



Представление гипотез

В генетических алгоритмах гипотезы представляются бинарными строками, для удобной манипуляции операторами мутации, кроссовера.

(Outlook = Overcast v Rain) AND (Wind = Strong)
THEN PlayTennis = yes <=>

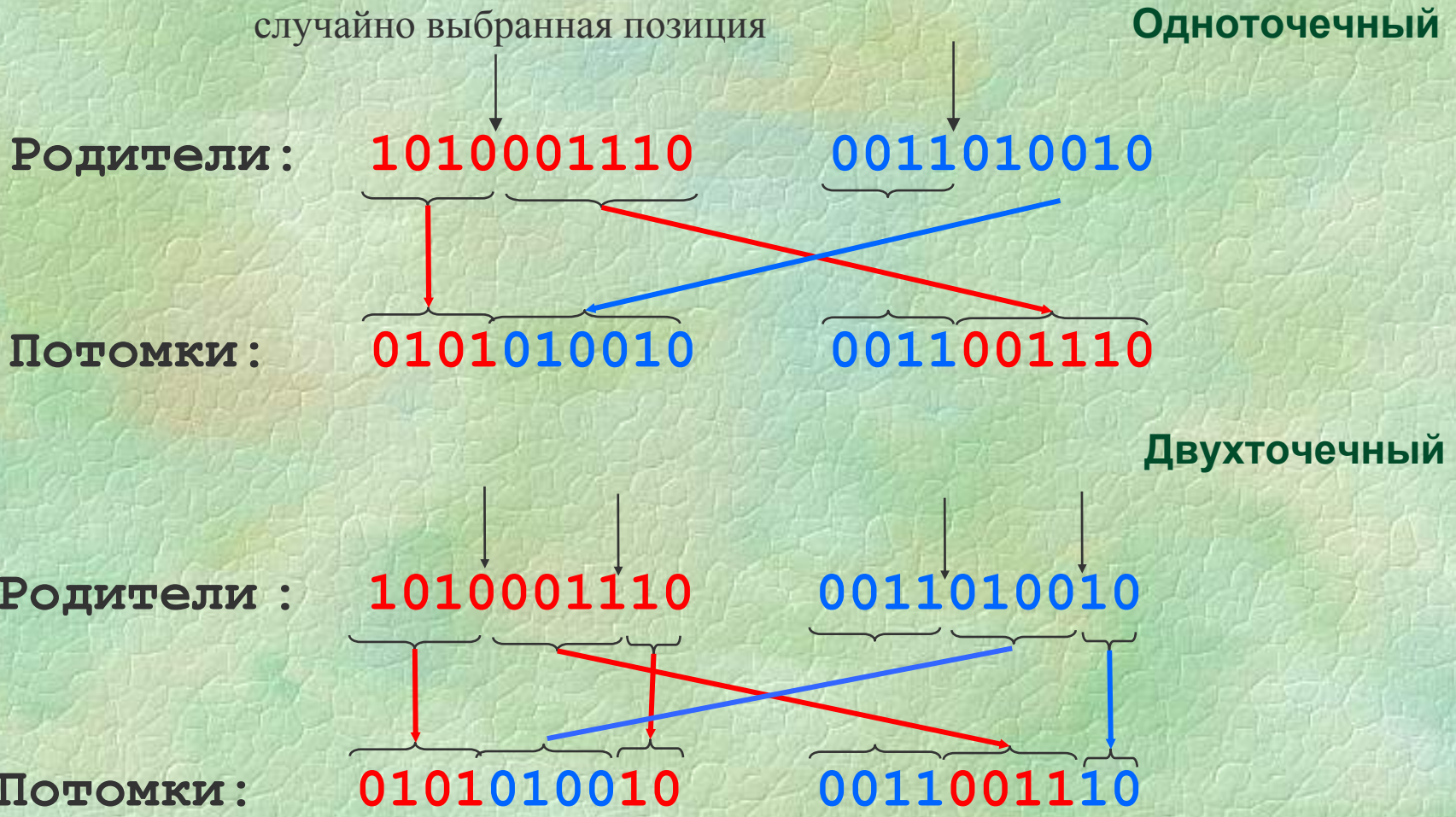
011 10 10

где

группа 1 = 3-значный Outlook,
группа 2 = 2-значный Wind
группа 3 = 2-значный PlayTennis

Генетические операторы

кроссовер



Генетические операторы

кроссовер

Маска кроссовера – бинарная строка, генерится случайным образом и обозначает, какая информация должна быть взята от каждого из родителей.

В общем виде

Маска: 0110011000

Родители: 1010001110 0011010010

Потомки: 0011001010 1010010110

мутация

Родители: 1010001110

Потомки: 1000001110

Параметры алгоритма

Функция **Fitness**, степень соответствия гипотезы данным. Например,

$$\text{Fitness}(h)=(\text{correct}(h))^2,$$

Порог соответствия, **threshold**.

p: число гипотез в текущей популяции

r: часть гипотез, замещаемая оператором кроссовер

m: уровень мутаций

Формирование популяции

Выбрать (1-r)p членов популяции $P_{current}$ и добавить к популяции P_{new}. Вероятность выбора индивида:

$$P(h_i) = Fitness(h_i) / \Sigma_j Fitness(h_j)$$

Кроссовер- выбрать rp/2 пар гипотез из $P_{current}$ в соответствии с $P(h_i)$. Для каждой пары (h1,h2) произвести два потомка посредством применения оператора кроссовер и добавить к P_{new}.

Мутация. Выбрать m членов P_{new} с равномерным распределением. Изменить один бит в записи на случайно выбранном месте.

Заменить $P_{current}$ на P_{new}

Для каждой гипотезы h оценить значение функции Fitness.

Поиск в пространстве гипотез

Поиск гипотез в генетическом алгоритме характеризуется как “внезапный”, “хаотичный”, рандомизированный, когда траектория алгоритма не может быть предсказана, даже если известна отправная точка, в отличие от направленного поиска гипотез.

Имеет место феномен перенаселения (crowding), соответствующий быстрому воспроизводству небольшого числа наиболее приспособленных гипотез (с максимальным значением функции Fitness). Для преодоления этого феномена используют другие критерии выбора гипотез, например, турнир (tournament selection), или назначают штрафы гипотезам, имеющим копии в популяции.

Для увеличения разнообразия популяции проделывают так называемый акт “геноцида”, когда 80% популяции замещается случайными гипотезами, таким образом к популяции добавляются различные “сущности”.

Терминальными условиями алгоритма являются

- отсутствие событий
- значение максимальной приспособленности не меняется

Эволюция популяции

Определение. Схемой является запись, состоящая из 0, 1, и *, и представляющая собой набор бинарных строк, в записи которых стоит 0 и 1 в соответствии со схемой, и произвольные значения на местах, обозначенных *.
Схема * * 1 * * 0 0 * * 1 представляет все строки длин 10:

1 1 **1** 0 1 **0 0** 1 0 **1**
0 1 **1** 1 0 **0 0** 0 1 **1**
…

Бинарные строки, удовлетворяющие условиям схемы, называются представителями схемы.

H – схема, имеющая по крайней мере одного представителя на момент времени ***t***.

m*(*H*, *t*)** – количество представителей ***H на момент ***t***.

f*(*H*, *t*)** - среднее значение функции Fitness гипотез из ***H на момент ***t***.

f*(*x*)** – значение функции Fitness для произвольной гипотезы ***x.

$\overline{f}(t)$ - среднее значение функции Fitness для популяции на момент ***t***.

Мы хотим вычислить ***E*(*m*(*H*, *t*+1))**, ожидаемое число представителей схемы ***H*** на момент времени ***t* + 1**.

Эволюция популяции

выбор гипотез

Ограничиваясь случаем, когда в следующую популяцию попадают гипотезы с заданной вероятностью, пропорциональной их приспособленности, ожидаемое число представителей схемы H на момент времени $t + 1$ вычисляется следующим образом:

$$E\left(m\left(H,t+1\right)\right)=\sum_{x\in H}\frac{f\left(x\right)}{\bar{f}\left(t\right)}=m\left(H,t\right)\sum_{x\in H}\frac{f\left(x\right)}{m\left(H,t\right)\cdot\bar{f}\left(t\right)}=m\left(H,t\right)\cdot\frac{f\left(H,t\right)}{\bar{f}\left(t\right)},$$

где $x\in H$ означает, что гипотеза x является представителем схемы H .

p_{xc} - вероятность того, что одноточечный кроссовер будет применен к строке.

Sc*(*H*)** – вероятность того, что схема ***H выживет после применения одноточечного кроссовера, т.е. по крайней мере один из потомков представителя ***H*** останется представителем ***H***.

L*(*H*)** - расстояния между первым и последним фиксированными битами схемы ***H.

l – длина гипотезы.

Эволюция популяций

эффект одноточечного кроссовера

Вероятность того, что схема будет разрушена не хуже, чем:

$$p_{xo}\left(\frac{L(H)}{l-1}\right),$$

что позволяет оценить вероятность того, что схема выживет при воздействии одноточечным кроссовером:

$$S_c(H) \geq 1 - p_{xo}\left(\frac{L(H)}{l-1}\right).$$

Эволюция популяций

эффект мутаций

Теперь необходимо оценить эффект мутаций, привносимый в разрушение схемы при формировании новой популяции.

p_m - вероятность мутации одного бита.

o*(*H*)** – порядок схемы ***H.

S*_{*m*}(*H*)** – вероятность того, что схема ***H выживет при мутации представителя ***H***.

Имеет место неравенство: $S_m(H) \geq (1 - p_m) o(H)$

Holland's Schema Theorem

$$E\left(m\left(H,t+1\right)\right) \geq \frac{f\left(H,t\right)}{\bar{f}\left(t\right)} \cdot m\left(H,t\right) \cdot \left(1 - p_{xo} \cdot \frac{L\left(H\right)}{l-1}\right) \cdot \left(1 - p_m\right)^{o(H)}.$$

Evolutionary Computation in Bioinformatics: A Review

Sankar K. Pal, Sanghamitra Bandyopadhyay and Shubhra Sankar Ray

NP-трудная проблема

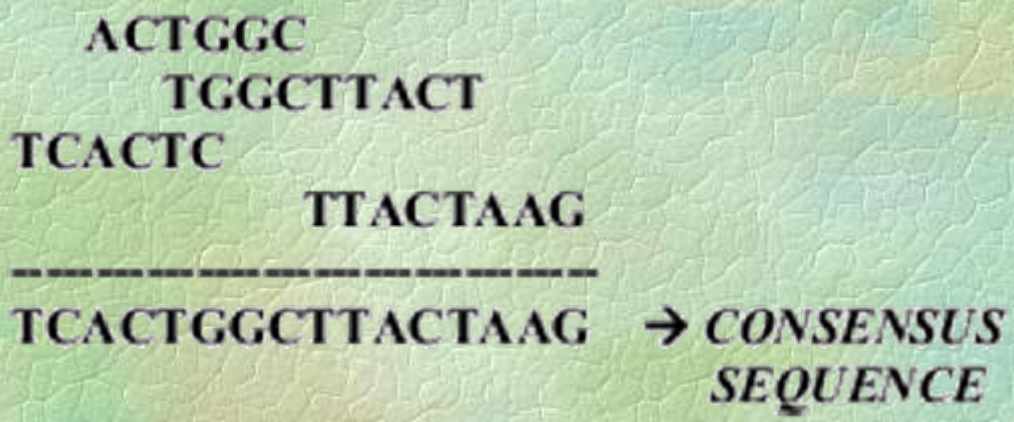
Пусть *1*, *2*,..*n* – метки *n* городов и $C = [c_{i,j}]$ - матрица стоимостей $c_{i,j}$ между городами, *i* и *j*. Требуется найти наиболее дешевый маршрут, соединяющий все города, т.е. минимизировать функционал (совокупную стоимость перемещений):

$$A(n) = \sum_{i=1}^{n-1} c_{i,i+1} + c_{n,1}$$

Проблема сборки фрагментов ДНК является NP-трудной проблемой

IEEE Transactions on Systems, Man, and Cybernetics, Part-C, vol. 36, no. 5, pp. 601-615, 2006

Применение генетических алгоритмов к задаче сборки фрагментов ДНК



$$Fitness = \sum_{i=1}^{n-1} W_{f_i, f_{i+1}}$$

$f_1, f_2, \ldots, f_n$ - некое упорядочивание фрагментов, в котором нет повторов, и запись $f_i = j$ означает, что фрагмент i начинается в позиции j результирующей последовательности (которую определяет упорядоченный набор фрагментов).

Применение генетических алгоритмов к задачам биоинформатики

Генетические алгоритмы успешно применялись для:

- задач множественного выравнивания;
- распознавания структуры РНК;
- извлечения регуляторных генных сетей из профилей экспрессии генов;
- распознавания сайтов связывания транскрипционных факторов;
- разработки медицинских препаратов.

Генетические алгоритмы предназначены для решения сложных многопараметрических задач оптимизации, где прямой перебор параметров затруднен или невозможен