

# Самообучающиеся системы и нечёткая логика

---

Сергей Николенко, 03.05.2006



# SE2004 Curriculum Guidelines

---

- SAS.ab. Agent-based systems
  - SAS.ab.1 Machine learning
  - SAS.ab.2 Fuzzy logic
  - SAS.ab.3 Knowledge engineering



# План лекции

---

- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



# Machine Learning: суть

---

- Что значит – обучающаяся машина?  
Как определить «обучаемость»?



# Machine Learning: суть

---

- Что значит – обучающаяся машина?  
Как определить «обучаемость»?

Компьютерная программа *обучается* по мере накопления опыта относительно некоторого класса задач  $T$  и целевой функции  $P$ , если качество решения этих задач (относительно  $P$ ) улучшается с получением нового опыта.



# Machine Learning: суть

---

Компьютерная программа *обучается* по мере накопления опыта относительно некоторого класса задач  $T$  и целевой функции  $P$ , если качество решения этих задач (относительно  $P$ ) улучшается с получением нового опыта.

- Определение очень (слишком?) общее;
- Какие конкретные примеры можно привести?



# Применение ML

---

Компьютерная программа *обучается* по мере накопления опыта относительно некоторого класса задач  $T$  и целевой функции  $P$ , если качество решения этих задач (относительно  $P$ ) улучшается с получением нового опыта.

- Поиск закономерностей:
  - Распознавание образов
  - Автоматическая классификация



# Применение ML

---

Компьютерная программа *обучается* по мере накопления опыта относительно некоторого класса задач  $T$  и целевой функции  $R$ , если качество решения этих задач (относительно  $R$ ) улучшается с получением нового опыта.

- Там, где невозможно достичь идеала:
  - Распознавание речи
  - Естественные языки



# Применение ML

---

Компьютерная программа *обучается* по мере накопления опыта относительно некоторого класса задач  $T$  и целевой функции  $R$ , если качество решения этих задач (относительно  $R$ ) улучшается с получением нового опыта.

- Игры:
  - TD-Gammon
  - Deep Blue, Fritz и другие...



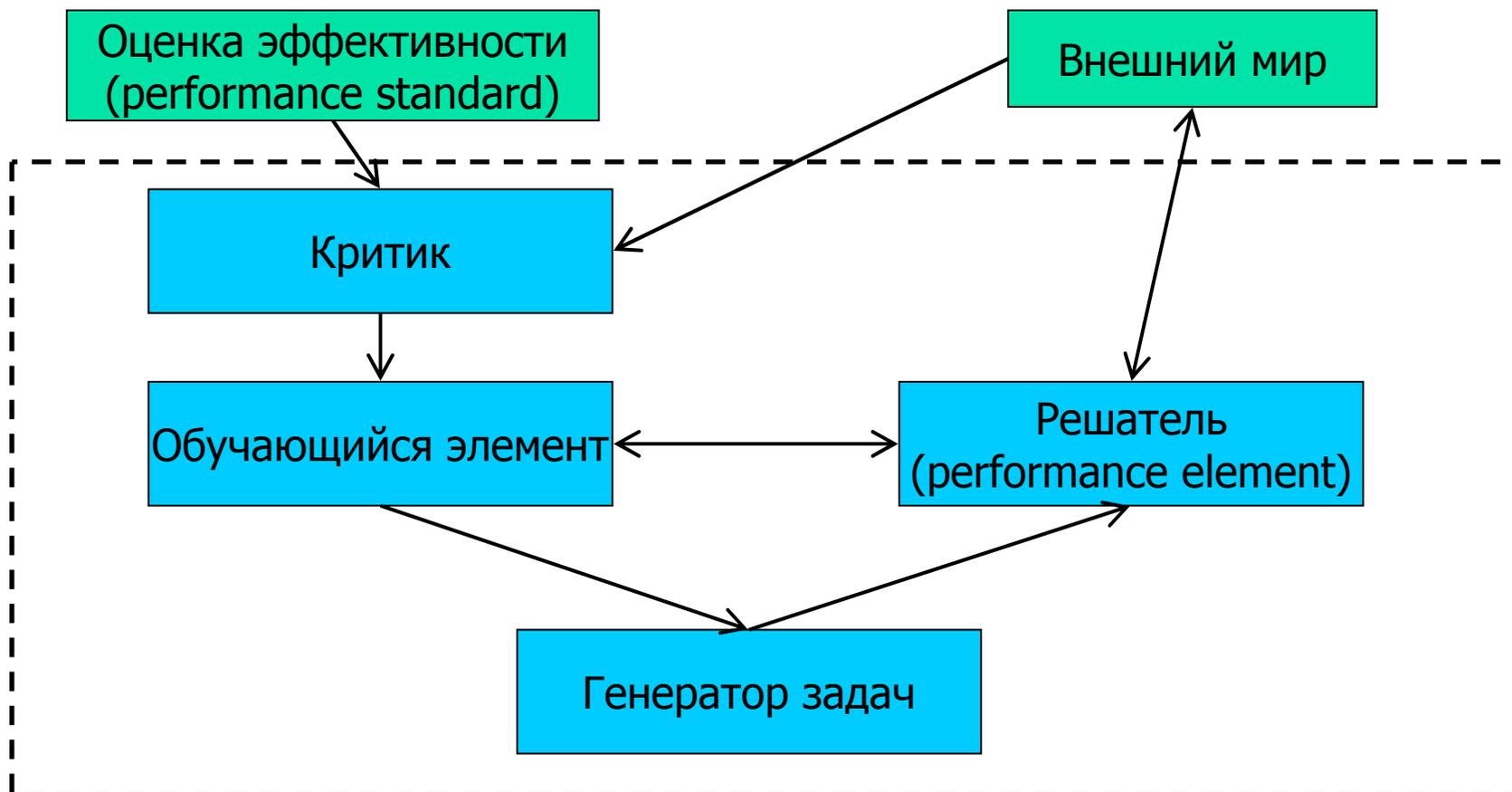
# Machine Learning: суть

---

- Интеллектуальные агенты действуют в условиях неполной информации
- Предсказать исход невозможно, но нужно научиться действовать оптимальным образом
- Если бы можно было предсказать, обучение было бы не нужно, только вычислительные мощности

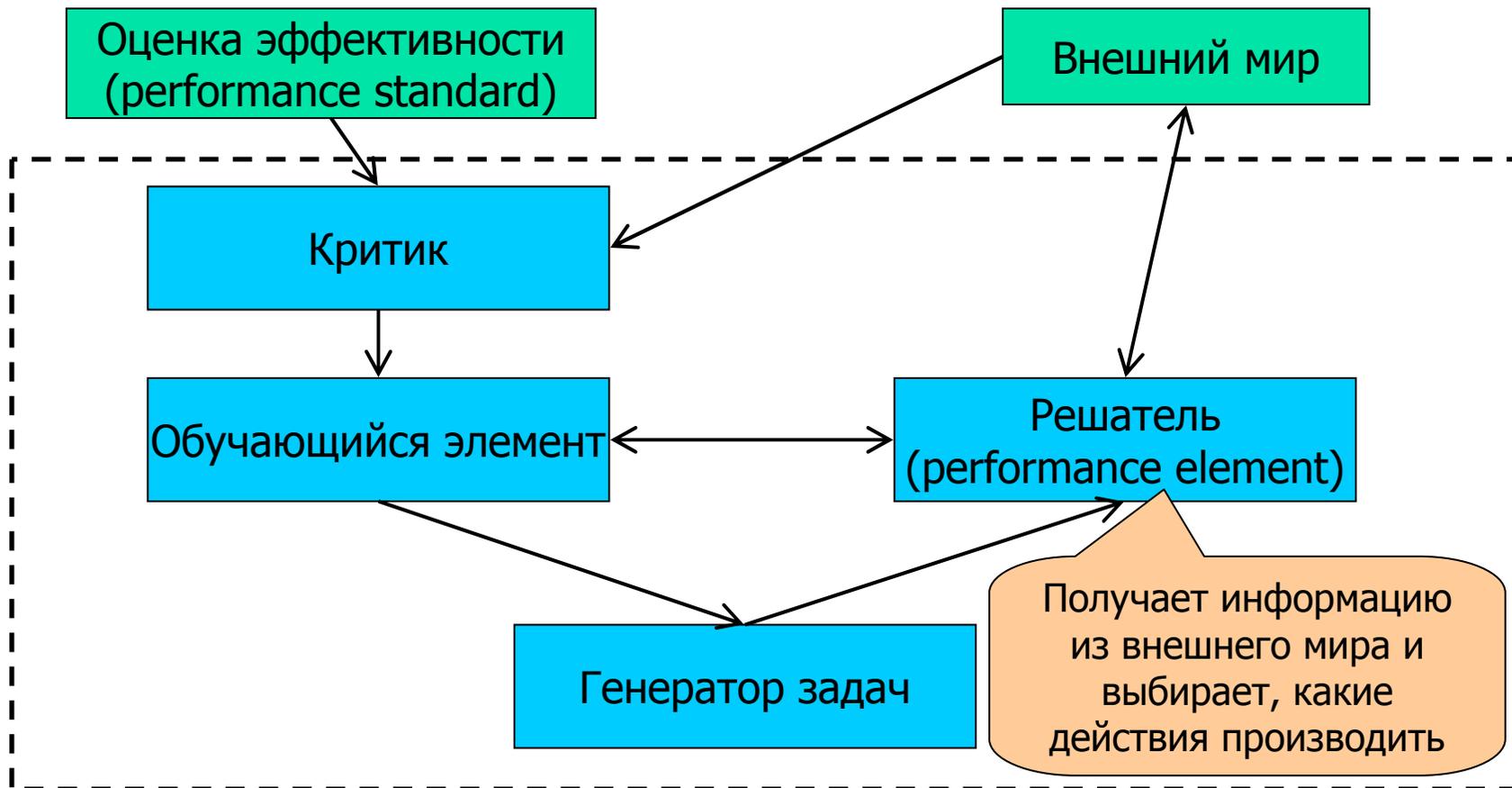
# Самообучающийся агент

- Общая схема самообучающегося агента:



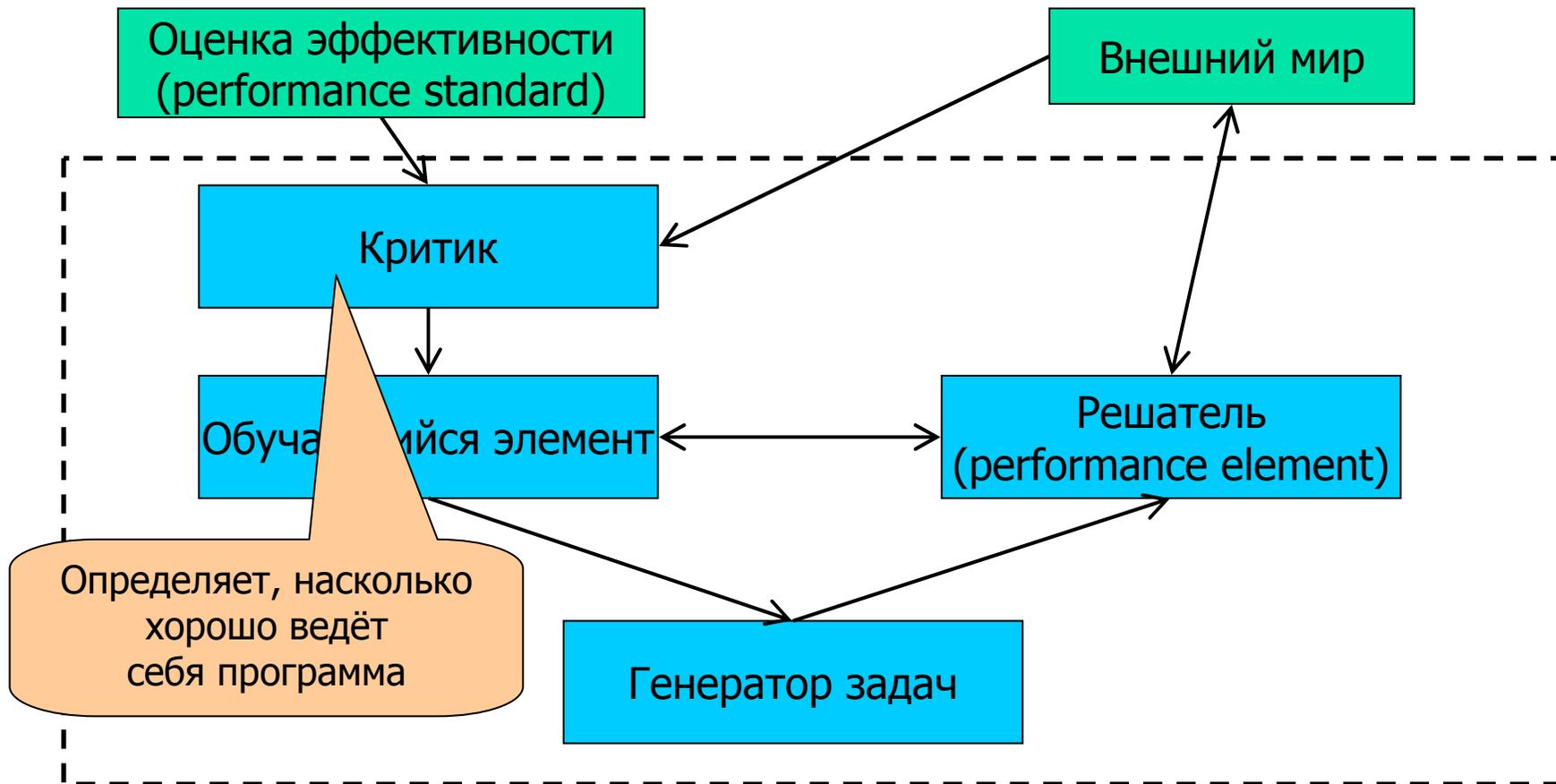
# Самообучающийся агент

- Общая схема самообучающегося агента:



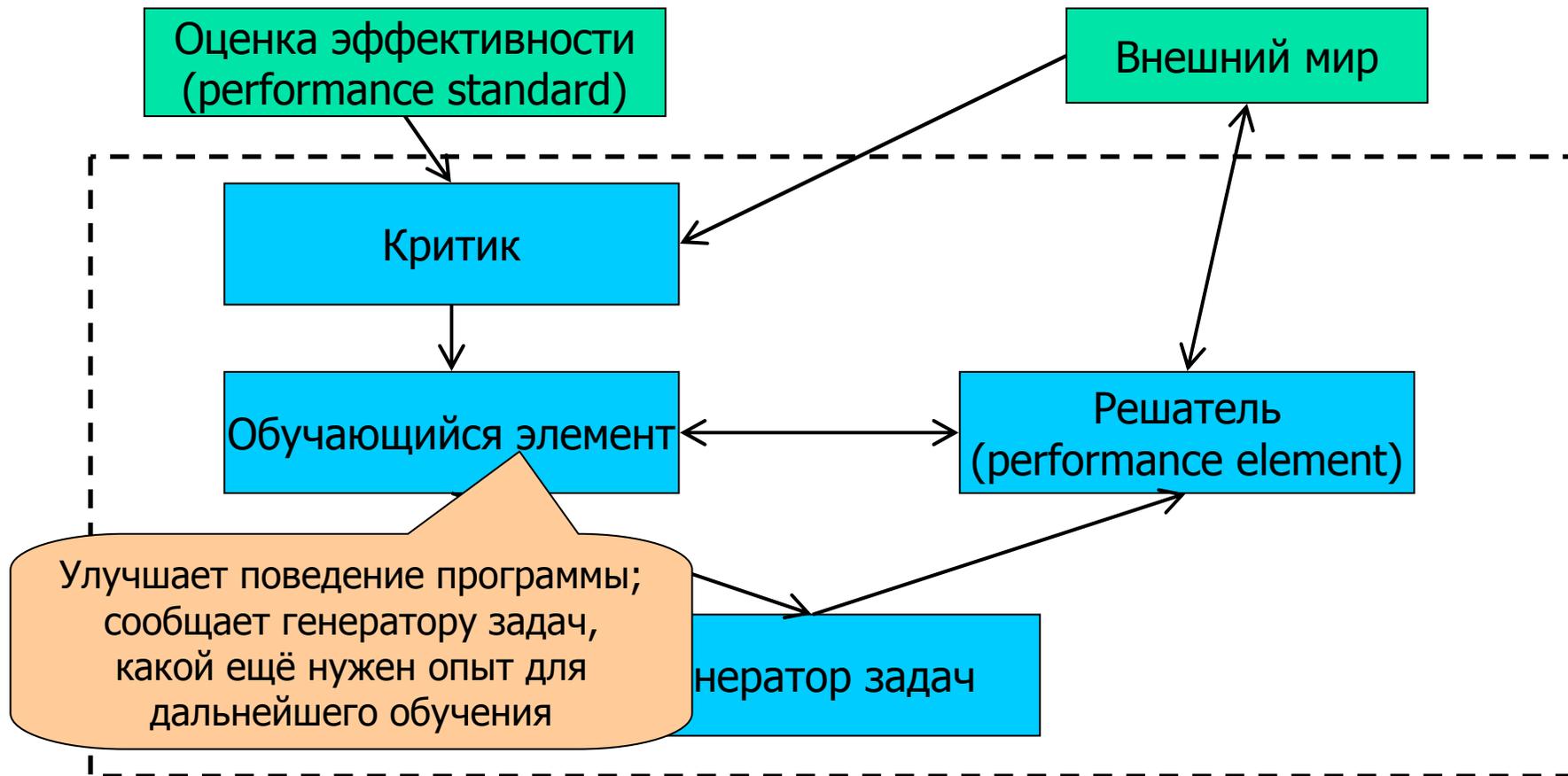
# Самообучающийся агент

- Общая схема самообучающегося агента:



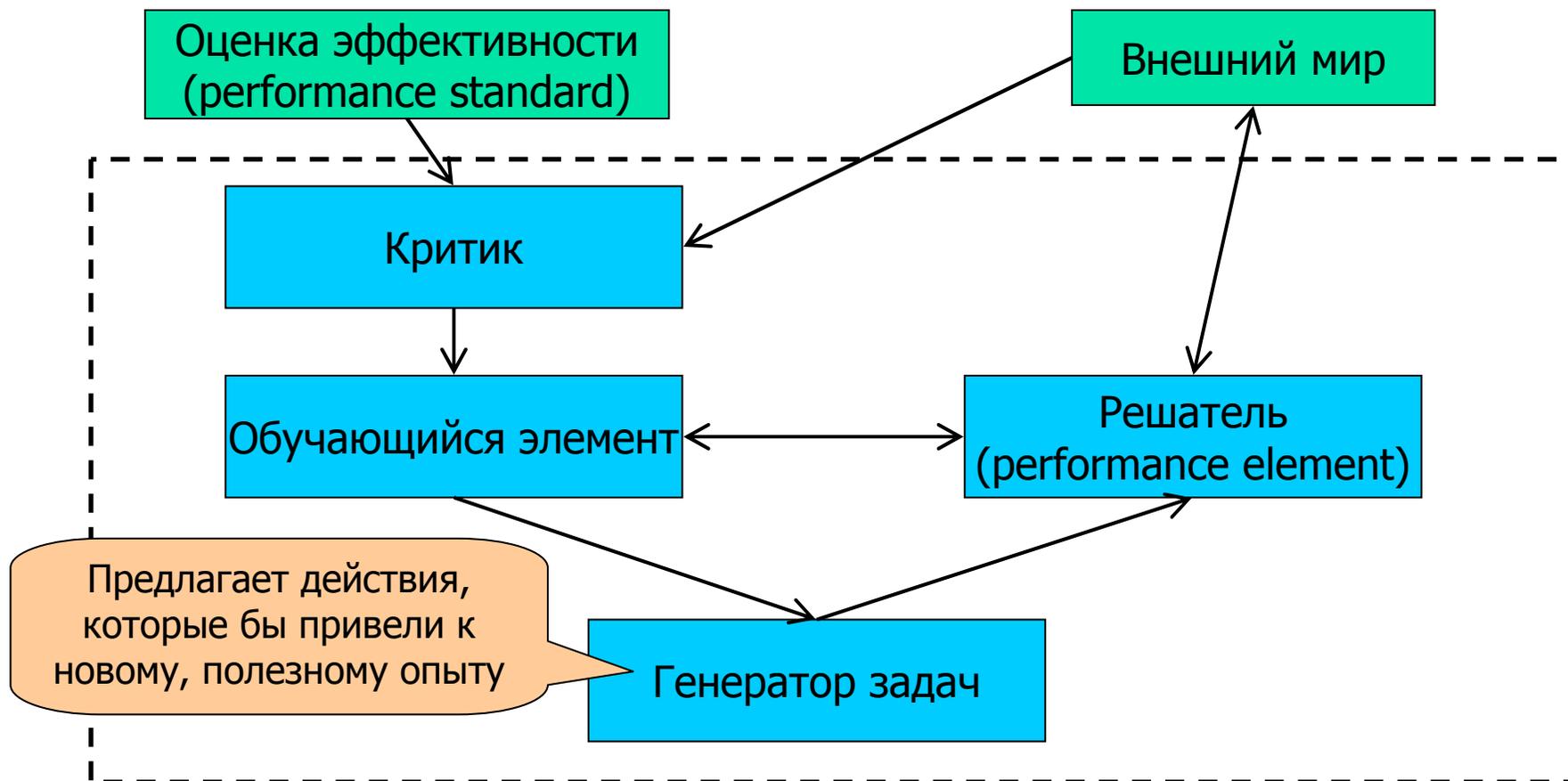
# Самообучающийся агент

- Общая схема самообучающегося агента:



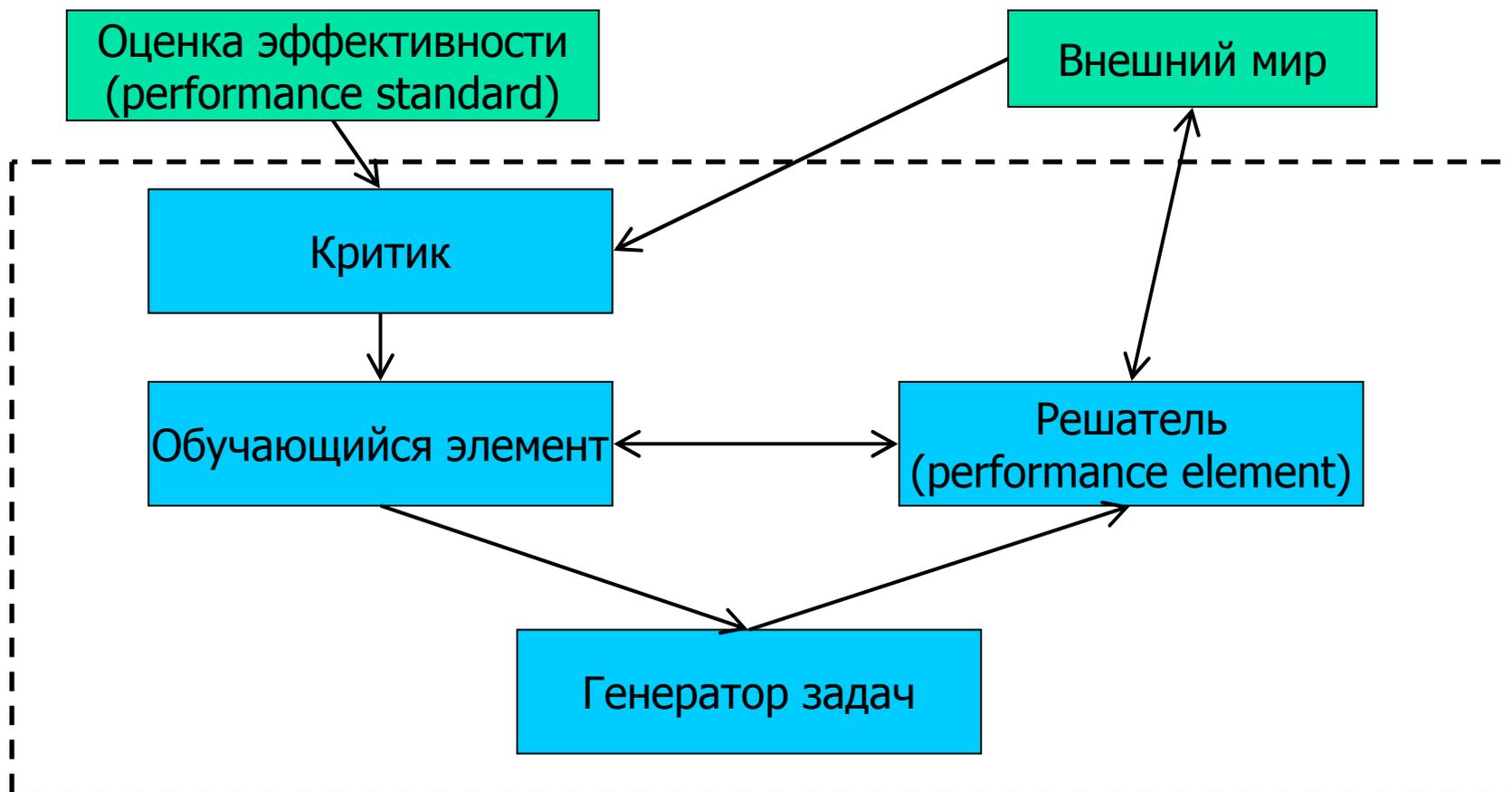
# Самообучающийся агент

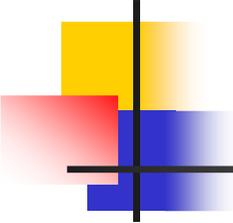
- Общая схема самообучающегося агента:



# Самообучающийся агент

- Общая схема самообучающегося агента:

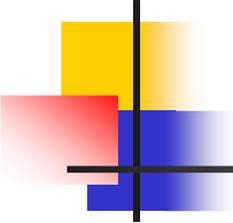




# План лекции

---

- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



# Decision trees

---

- Знания представляются в виде дерева (decision tree), в листьях которого – правила действия в отдельных случаях;
- Задача – аппроксимировать дискретную функцию при помощи такого дерева;
- Обучение – это процесс построения дерева; затем дерево используется для принятия решений; при поступлении новых данных дерево модифицируется.

# Классический пример

- Ищем зависимость между тем, можно ли играть в гольф, и разными погодными условиями

Play golf dataset

Independent variables				Dep. var
OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
sunny	85	85	FALSE	Don't Play
sunny	80	90	TRUE	Don't Play
overcast	83	78	FALSE	Play
rain	70	96	FALSE	Play
rain	68	80	FALSE	Play
rain	65	70	TRUE	Don't Play
overcast	64	65	TRUE	Play
sunny	72	95	FALSE	Don't Play
sunny	69	70	FALSE	Play
rain	75	80	FALSE	Play
sunny	75	70	TRUE	Play
overcast	72	90	TRUE	Play
overcast	81	75	FALSE	Play
rain	71	80	TRUE	Don't Play

# Классический пример

- Сначала делаем переменные дискретными

Play golf dataset

Independent variables				Dep. var
OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
sunny	85	Влажно	FALSE	Don't Play
sunny	80		TRUE	Don't Play
overcast	83		FALSE	Play
rain	70		FALSE	Play
rain	68		FALSE	Play
rain	65	Сухо	TRUE	Don't Play
overcast	64		TRUE	Play
sunny	72	Влажно	FALSE	Don't Play
sunny	69	Сухо	FALSE	Play
rain	75	Влажно	FALSE	Play
sunny	75	Сухо	TRUE	Play
overcast	72	Влажно	TRUE	Play
overcast	81		FALSE	Play
rain	71		TRUE	Don't Play

# Классический пример

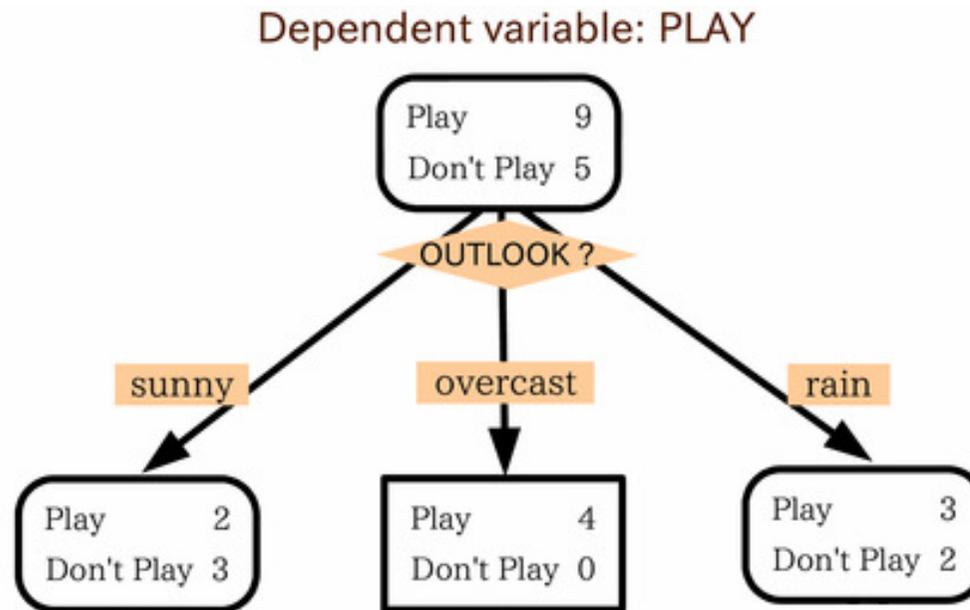
- Затем выбираем переменную, которая лучше всего классифицирует имеющиеся данные

Play golf dataset

Independent variables				Dep. var
OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
sunny	85	Влажно	FALSE	Don't Play
sunny	80		TRUE	Don't Play
overcast	83		FALSE	Play
rain	70		FALSE	Play
rain	68	Сухо	FALSE	Play
rain	65		TRUE	Don't Play
overcast	64	TRUE	Play	
sunny	72	Влажно	FALSE	Don't Play
sunny	69	Сухо	FALSE	Play
rain	75	Влажно	FALSE	Play
sunny	75	Сухо	TRUE	Play
overcast	72	Влажно	TRUE	Play
overcast	81		FALSE	Play
rain	71		TRUE	Don't Play

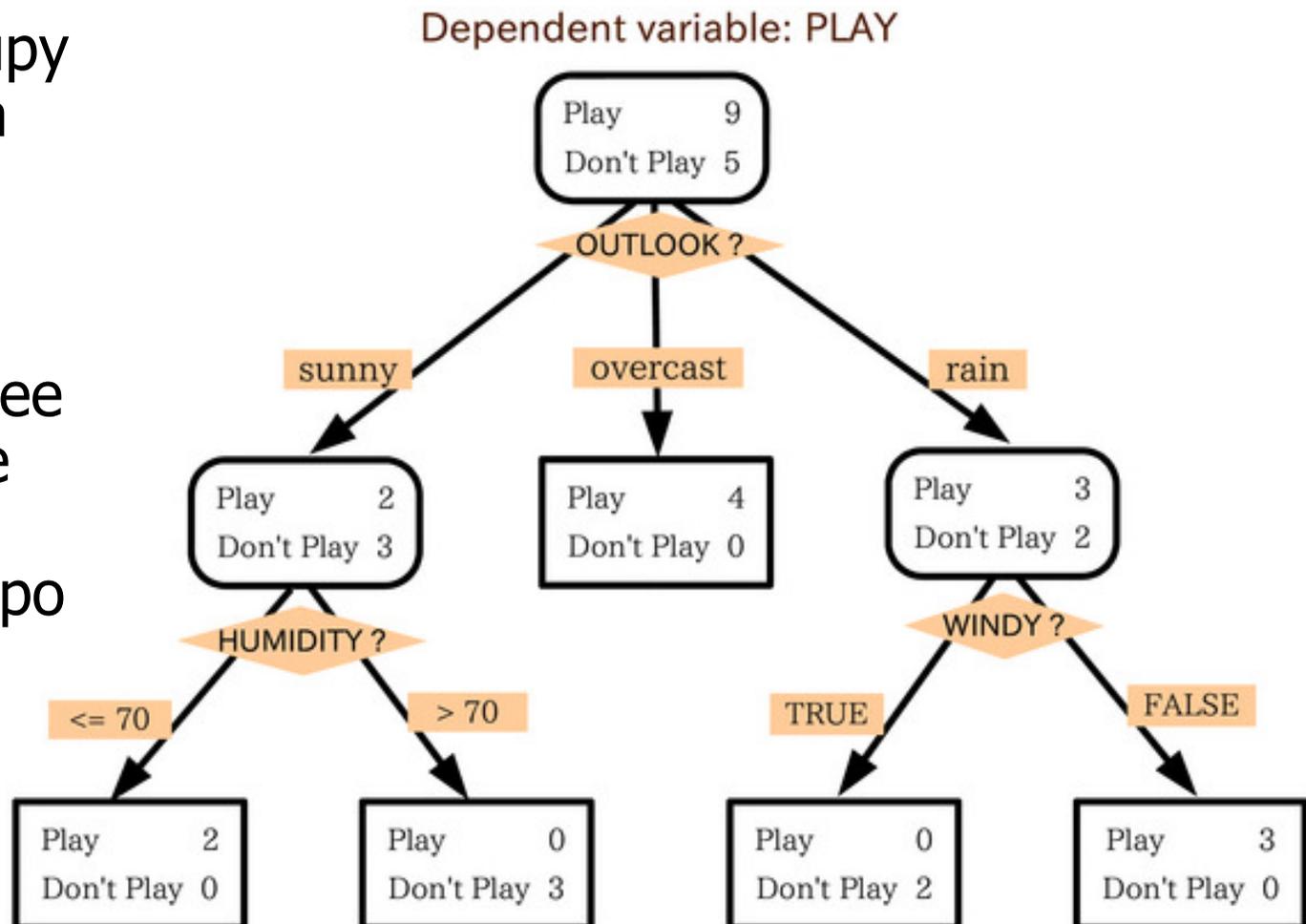
# Классический пример

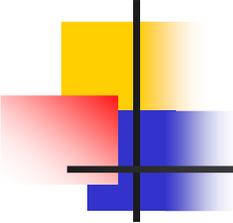
- Классифицируем по ней, а затем – по следующей переменной



# Классический пример

- Классифицируем по ней, а затем – по следующей переменной
- И decision tree ГОТОВО – все данные классифицированы

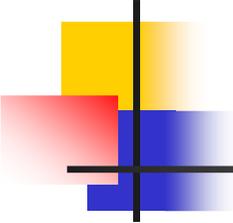




# План лекции

---

- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - [Нейронные сети](#)
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



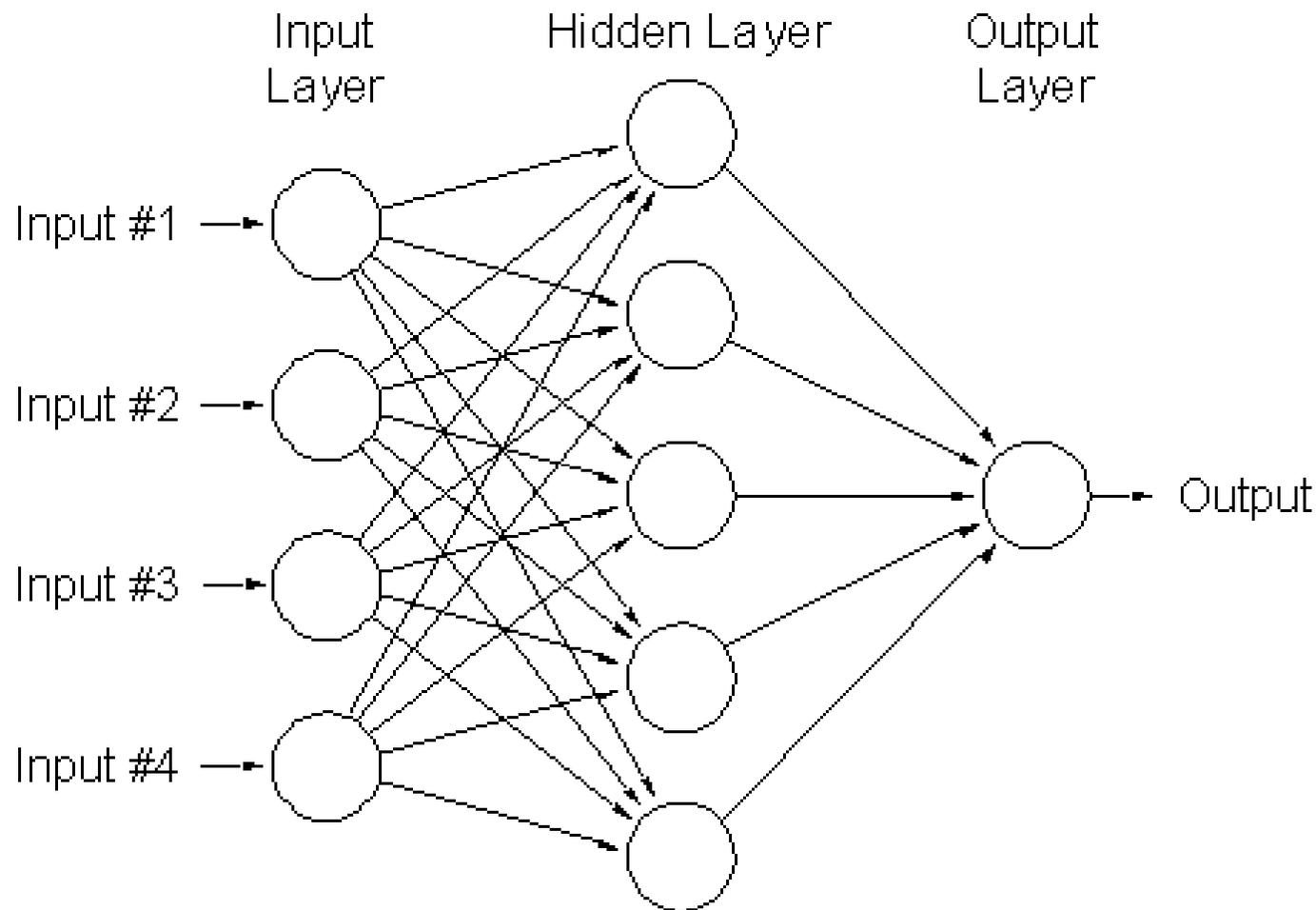
# Нейронные сети

---

- Сеть «нейронов» со связями между ними;
- Связи (рёбра графа) имеют веса;
- Сначала входы сети активируются в соответствии с ситуацией;
- То, как активируются выходы сети, транслируется в принятое решение;
- Обучение происходит посредством оценки результатов и соответствующего изменения весов на рёбрах.

# Нейронные сети

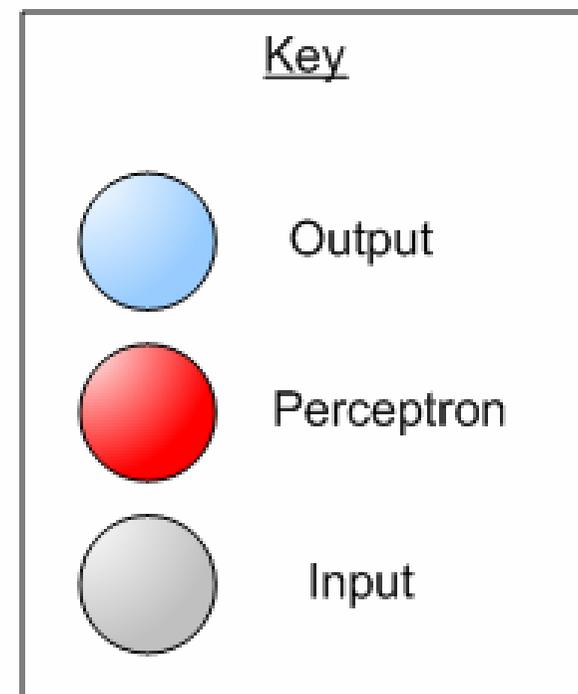
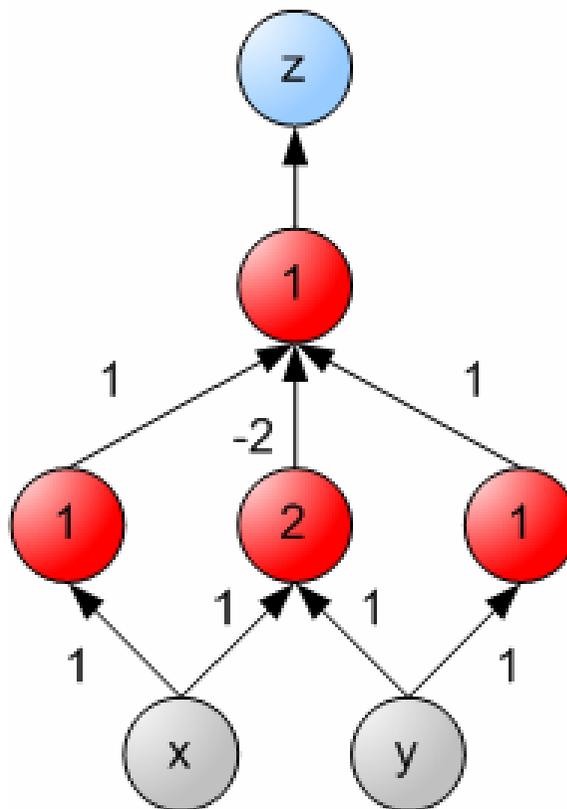
- Входы активируют нейроны, которые передают «возбуждение» дальше.
- Нейронная сеть как таковая просто представляет некоторую функцию.

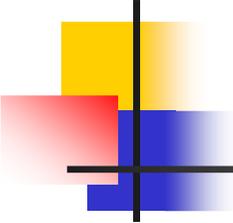


# Нейронные сети

- Входы активируют нейроны, которые передают «возбуждение» дальше.
- Нейронная сеть как таковая просто представляет некоторую функцию.

$$z = \text{XOR}(x, y)$$

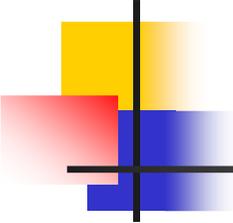




# Нейронные сети

---

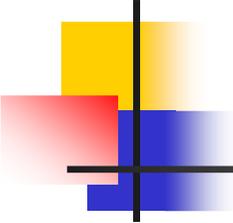
- Основная суть – в том, что сеть может учиться, изменяя свои параметры; например, у перцептрона могут меняться коэффициенты в функции, определяющей срабатывание.



# План лекции

---

- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры

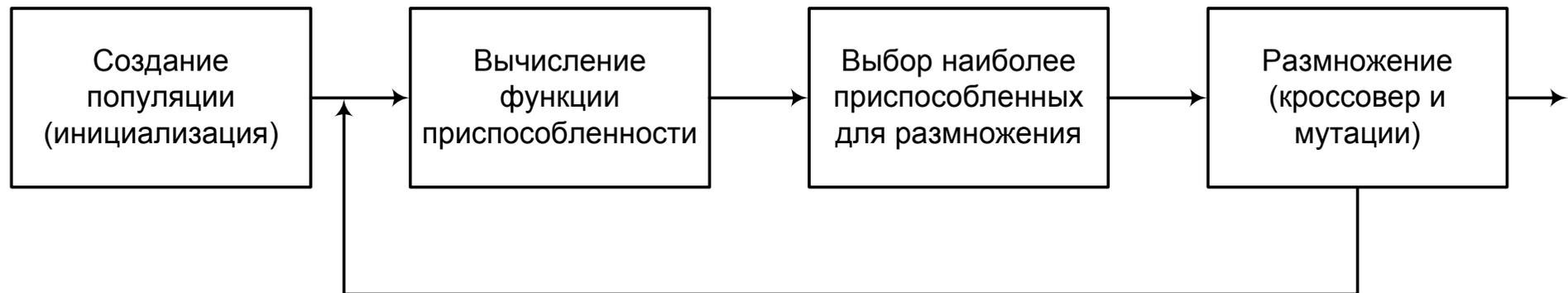


# Генетические алгоритмы

---

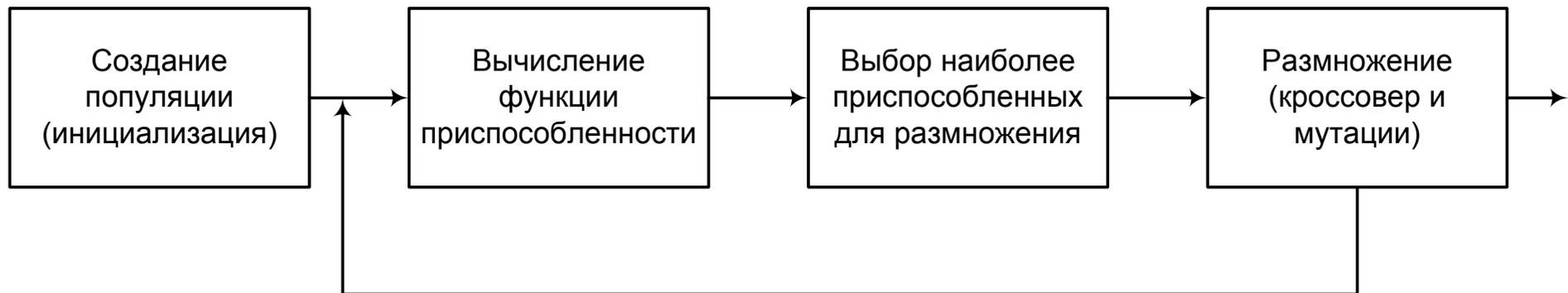
- База для работы алгоритма – множество гипотез (*популяция*);
- На каждом шаге возникают новые гипотезы, их проверяют на практике, происходит отбор, часть гипотез погибает;
- Наиболее вероятно, что на следующем шаге будут слегка видоизменённые уже имеющиеся гипотезы, а не совсем новые;
- Обучение – за счёт видоизменения популяции по мере накопления опыта.

# Генетические алгоритмы

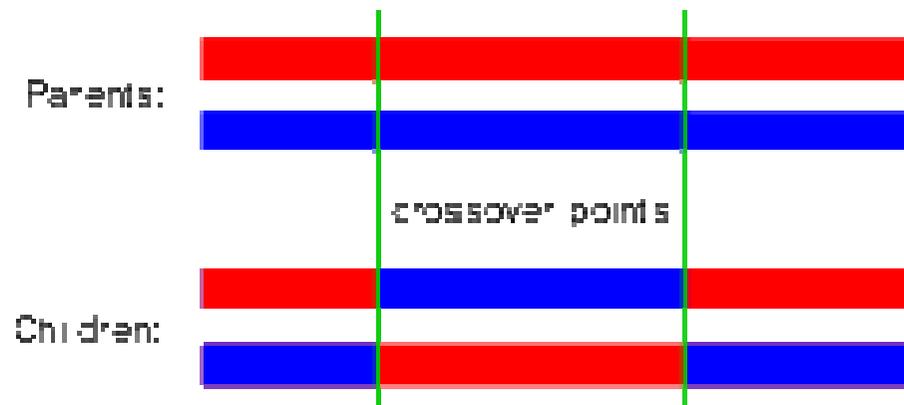
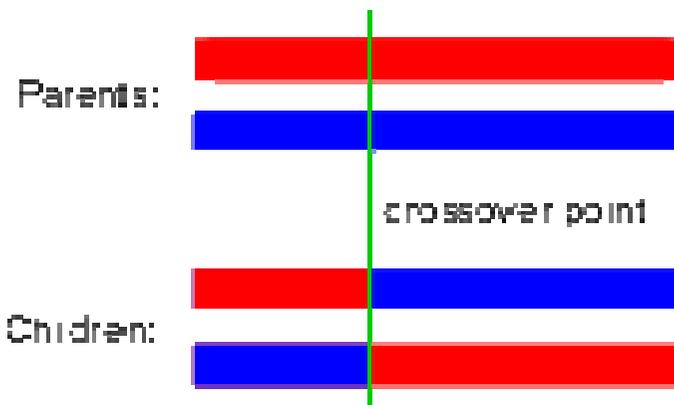


- Как правило, применяется для задач, в которых гипотезы (решения) можно удобно и естественно выразить строками битов. Так удобнее делать кроссовер.

# Генетические алгоритмы

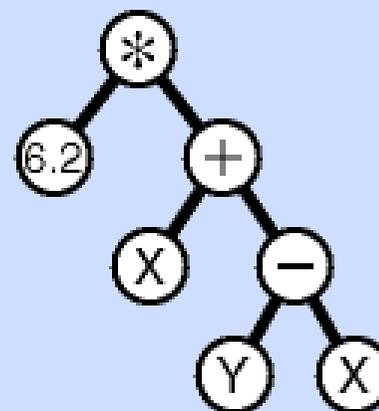
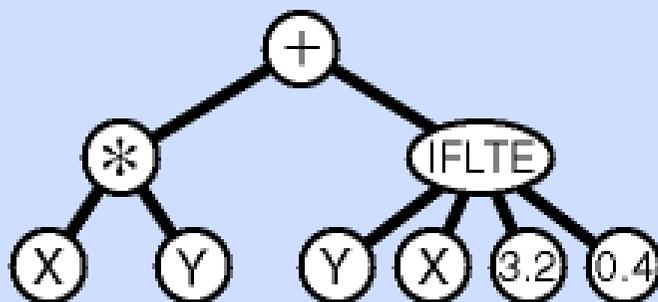


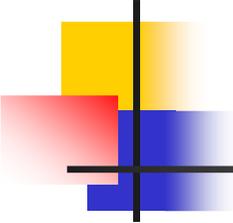
- Вот как обычно происходит кроссовер:



# Генетические алгоритмы

- Ещё одна интересная смежная область – *генетическое программирование*. В популяции участвуют программы! Кроссовер:

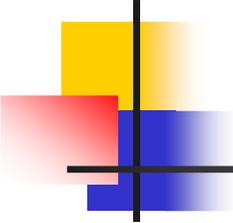




# План лекции

---

- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



# Байесовское обучение

---

- Теперь подключаем теорию вероятностей.
- Пусть есть набор гипотез  $H$  и данные  $D$ . Лучшая гипотеза – та, что наиболее вероятна при условии  $D$ . Эту вероятность можно вычислять по *теореме Байеса*:

$$p(h | D) = \frac{p(D | h)p(h)}{p(D)}$$

# Байесовское обучение

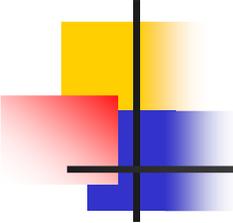
- Теперь подключаем теорию вероятностей.
- Пусть есть набор гипотез  $H$  и данные  $D$ . Лучшая гипотеза – та, что наиболее вероятна при условии  $D$ . Эту вероятность можно вычислять по *теореме Байеса*:

Апостериорная  
вероятность  $h$

$$p(h | D) = \frac{p(D | h)p(h)}{p(D)}$$

Априорная  
вероятность  
 $h$

Априорная  
вероятность  
 $D$



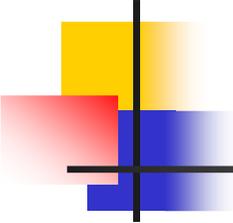
# Байесовское обучение

---

- Проблема: надо знать кучу вероятностей, в том числе совместных и условных при условии целого набора свидетельств.
- Решение: предполагать некие условия независимости или условной независимости, чтобы выполнялось

$$\begin{aligned} p(D = \{d_1, d_2, \dots, d_k\} | h) &= \\ &= p(d_1 | h) p(d_2 | h) \dots p(d_k | h) \end{aligned}$$

или другие удобные условия.

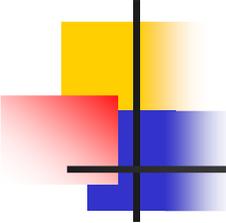


# Байесовское обучение

---

- Naïve Bayes classifier:
  - Каждое данное  $x$  – конъюнкция атрибутов, целевая функция  $f$  принимает конечное число значений  $v_1, v_2, \dots, v_m$ .
  - Атрибуты условно независимы при условии данного значения целевой функции:
$$p(a_1, a_2, \dots, a_n | v_j) = p(a_1 | v_j) p(a_2 | v_j) \dots p(a_n | v_j)$$
  - $p(v_j)$  и  $p(a_i | v_j)$  оцениваются из данных (это и есть обучение!), затем новые значения подсчитываются так:

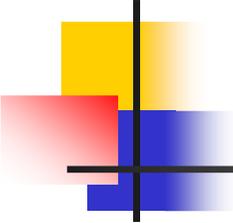
$$v_{NB} = \operatorname{argmax}_{v_j, j=1..m} p(v_j) \prod_i p(a_i | v_j)$$



# Байесовское обучение

---

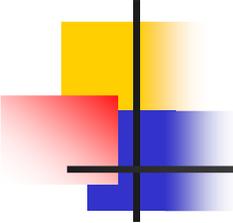
- Naïve Bayes classifier работает на удивление хорошо, но требуются и более сложные формализмы, которые могут отразить нетривиальную структуру условных независимостей



# План лекции

---

- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



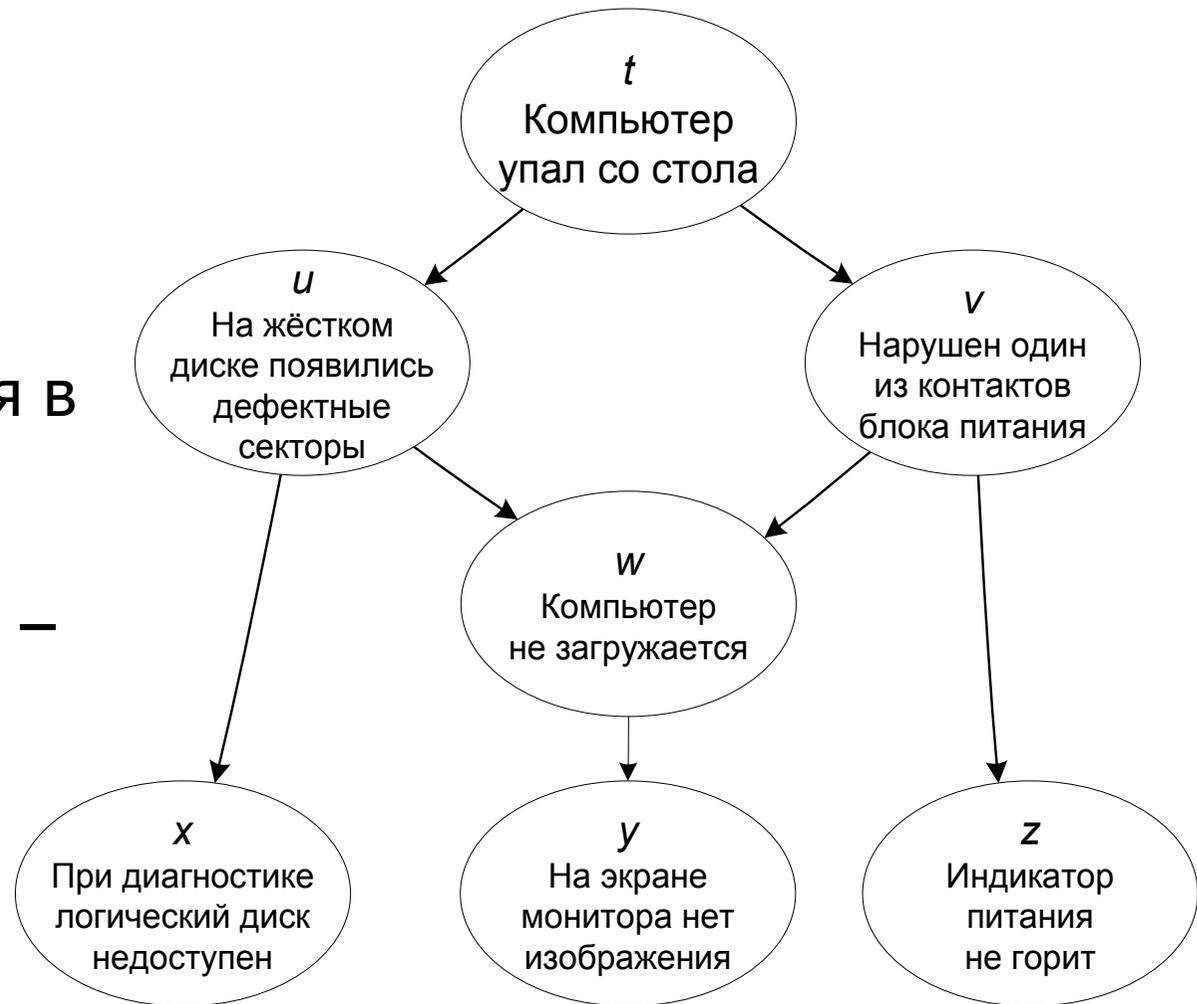
# Байесовские сети доверия

---

- Знания представляются в виде байесовской сети доверия: вершины графа – события, рёбра – зависимости; на рёбрах указаны условные вероятности;
- Принятие решений – байесовский вывод: по произошедшим событиям пересчитываются вероятности других событий;
- Обучение – построение сети либо поиск параметров (условных вероятностей) по уже заданной структуре сети.

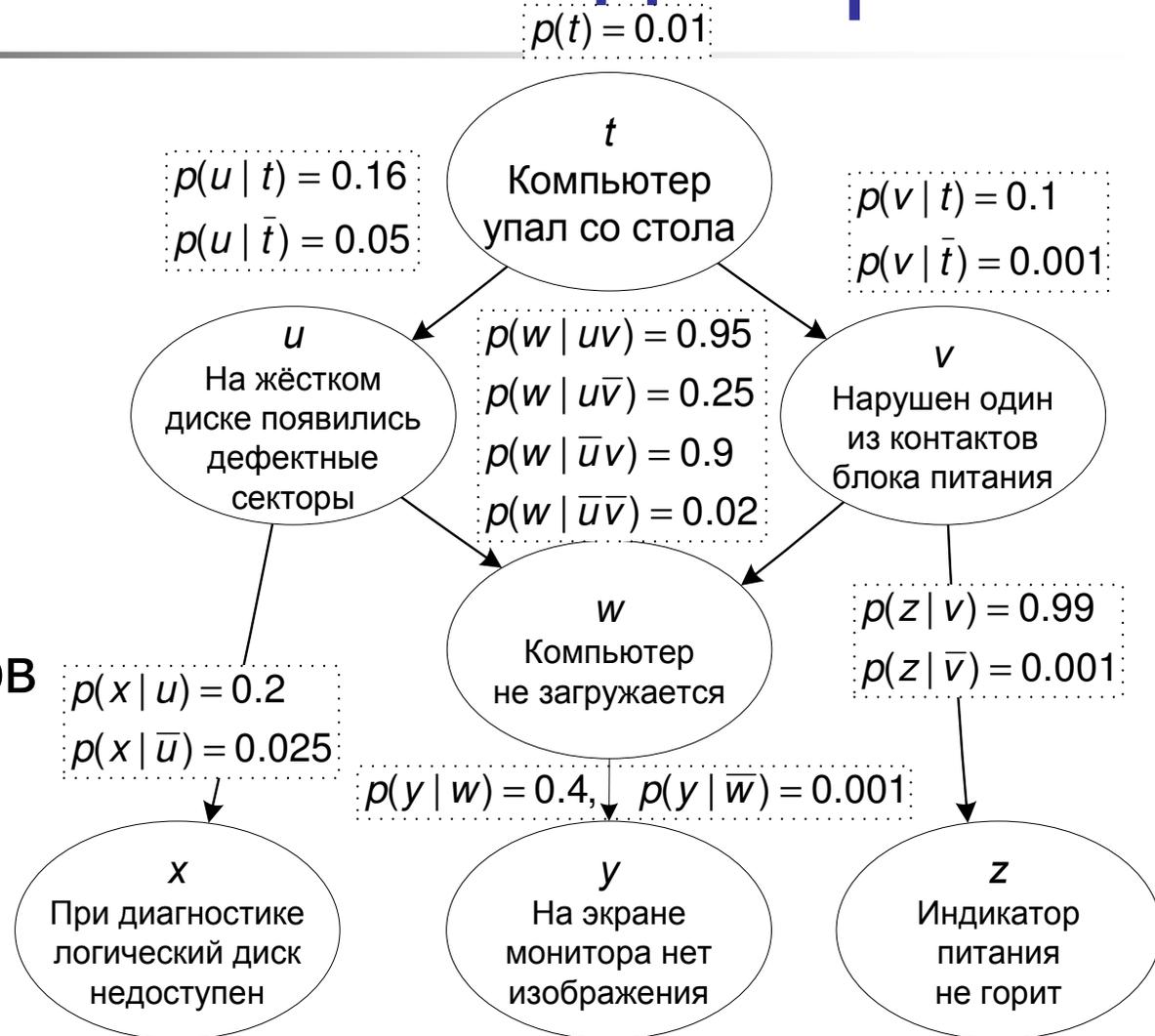
# Байесовские сети доверия

- Знания представляются в виде графа; вершины – события, рёбра – зависимости.



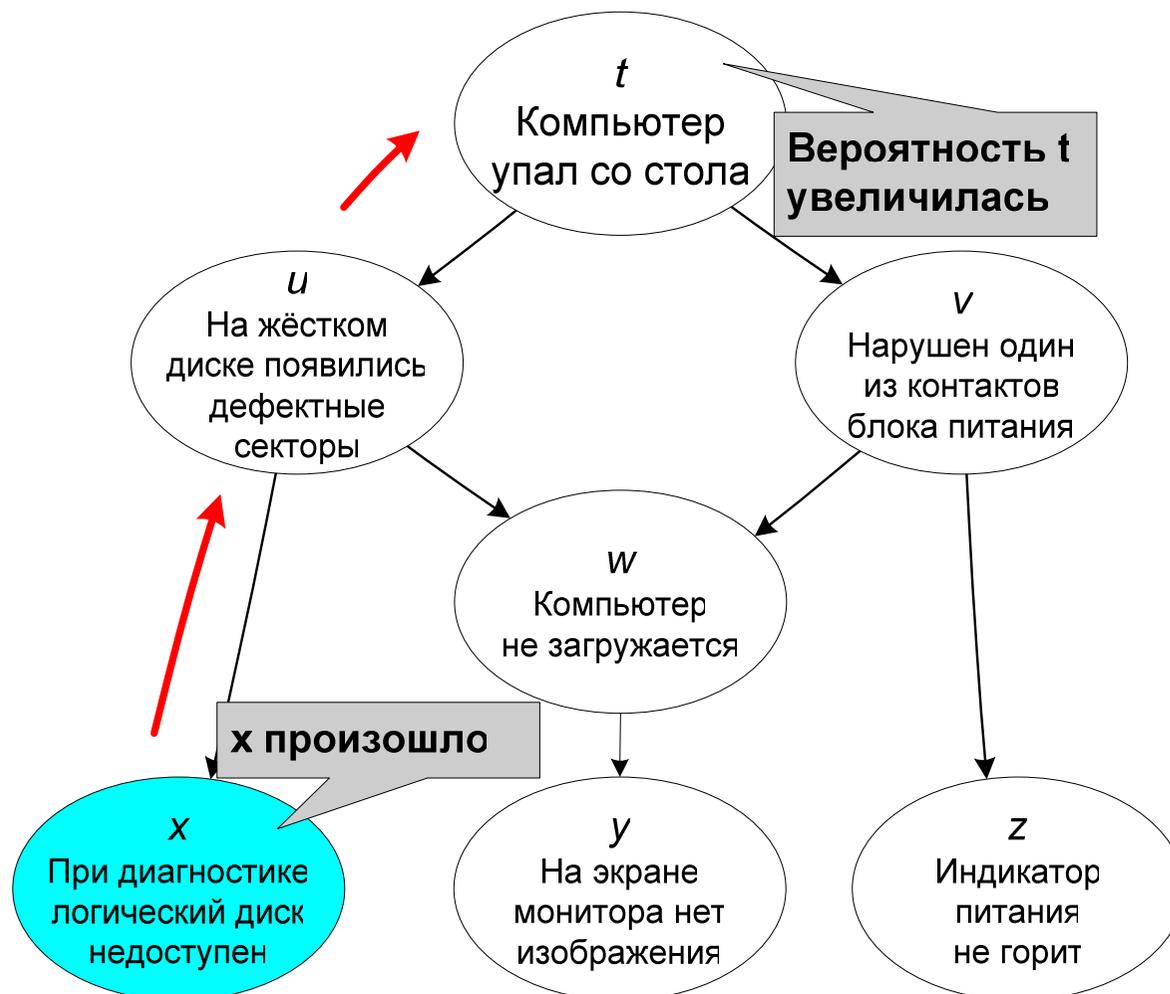
# Байесовские сети доверия

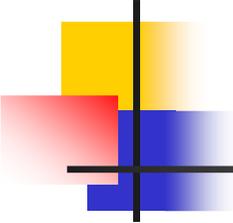
- Известны условные вероятности потомков при условии предков



# Байесовские сети доверия

- Суть – по поступившим свидетельствам пересчитать вероятности причин

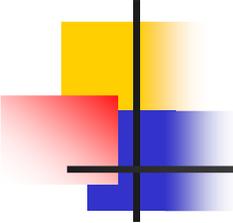




# Байесовские сети доверия

---

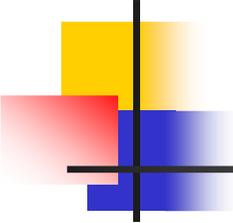
- Обучение – двух видов:
  - узнать вероятности на рёбрах, уже зная структуру сети (внешний вид графа);
  - узнать и вероятности, и структуру сети по имеющимся статистическим данным.



# План лекции

---

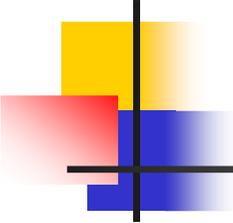
- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



# Нечёткость

---

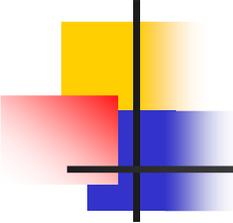
- Основная идея – формализовать такие понятия, как «немного», «приблизительно», определения вроде «высокий», «молодой», а также такие рассуждения, как «если пешеход близко, быстро дави на тормоз».



# Нечёткость

---

- Что такое «высокий человек»?



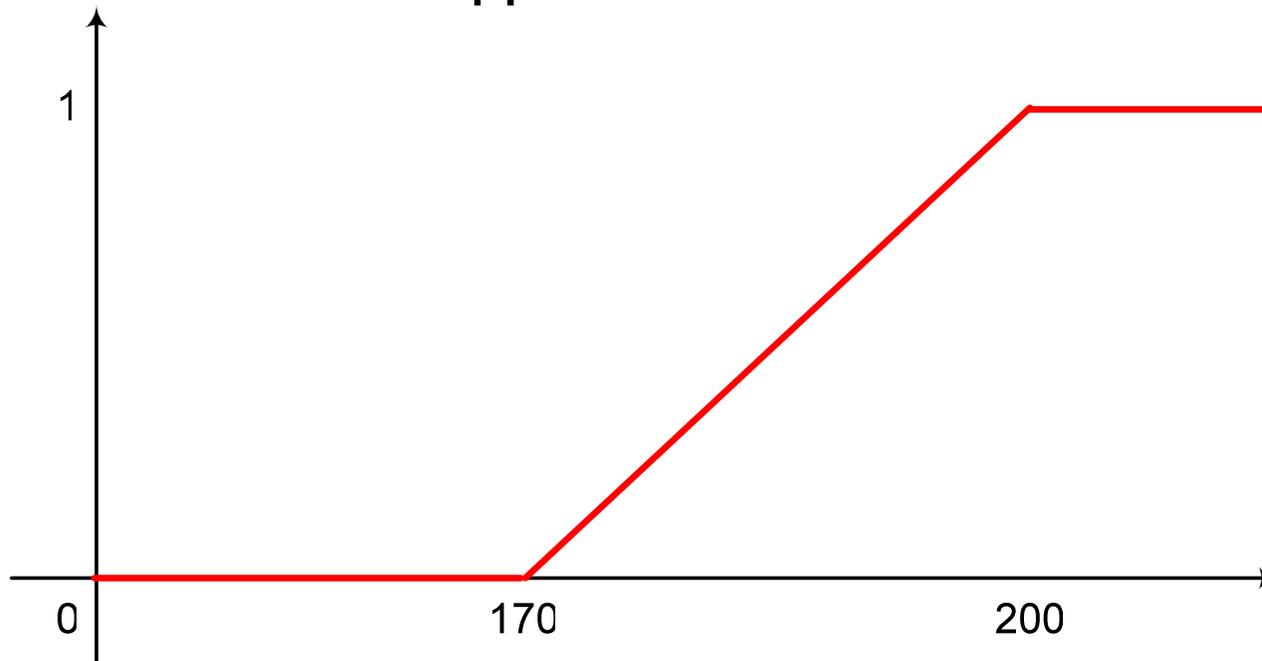
# Нечёткость

---

- Что такое «высокий человек»?
- Предположим, все согласны, что выше 2 метров – это высокий, а ниже 170 см – это не высокий.
- Что между этими границами?

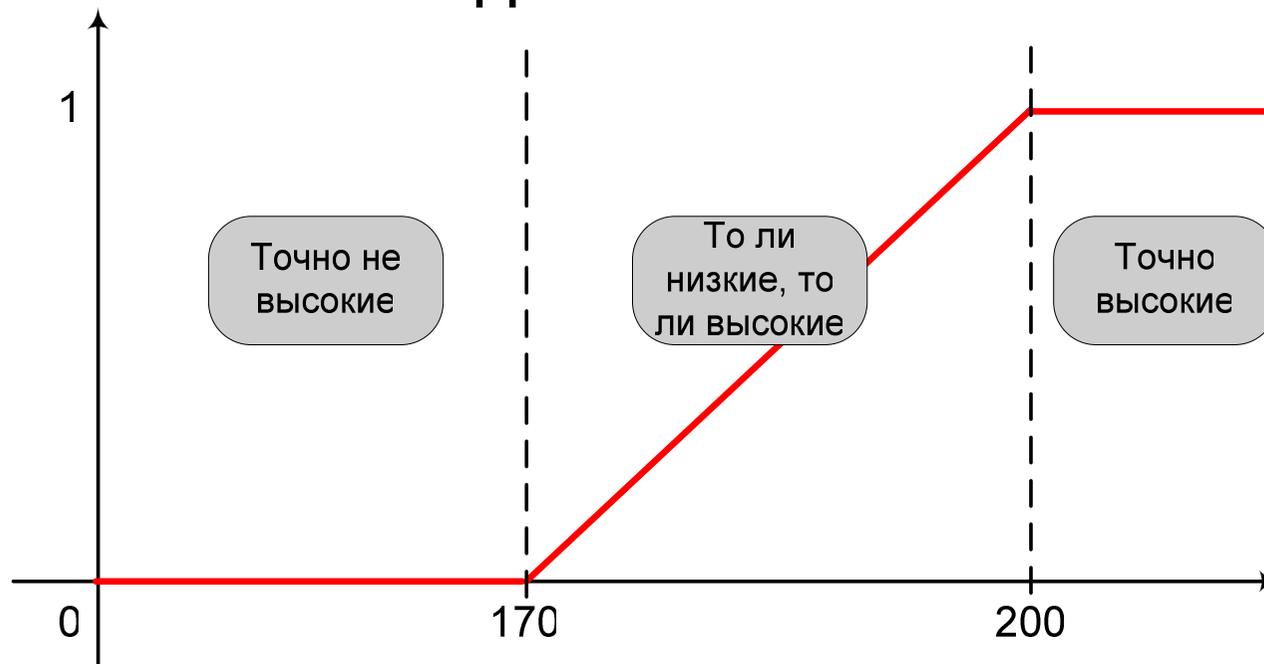
# Нечёткость

- Нечёткость предлагает «размазанную» функцию принадлежности множеству «высокие люди»:



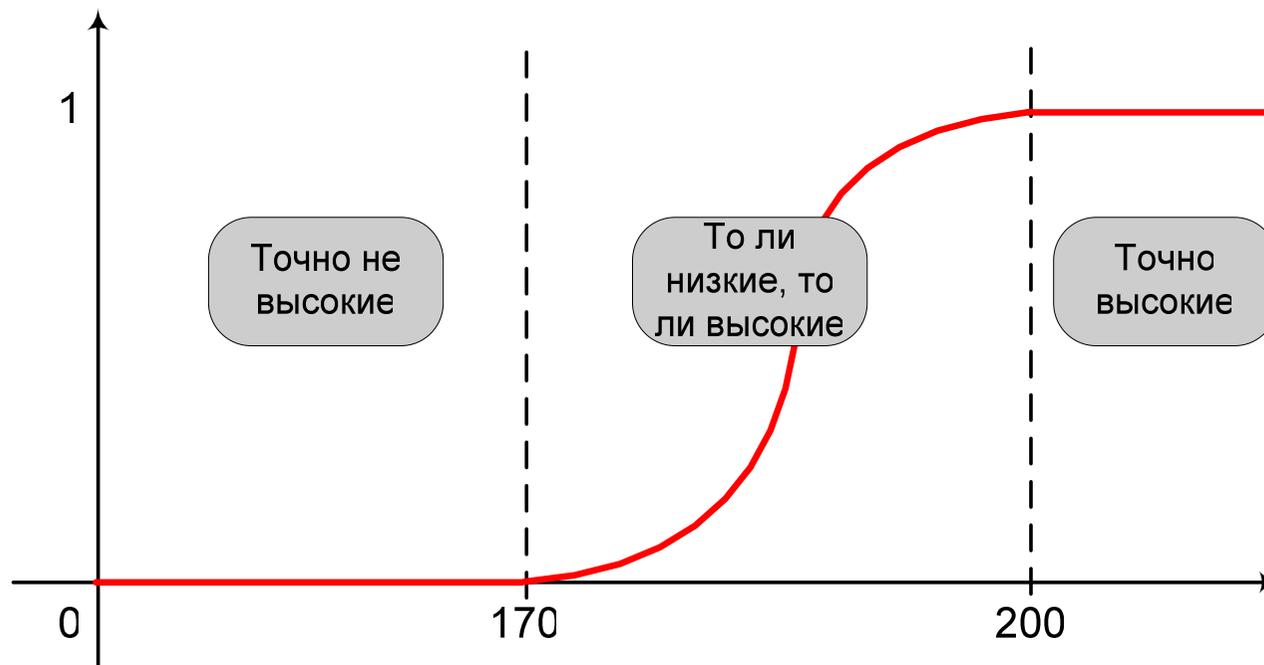
# Нечёткость

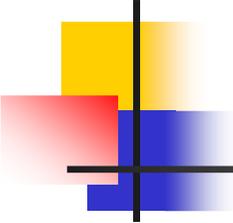
- Нечёткость предлагает «размазанную» функцию принадлежности множеству «высокие люди»:



# Нечёткость

- Функция не обязательно кусочно-линейная:

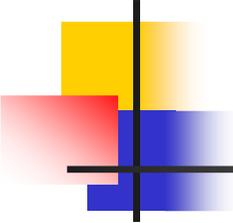




# Нечёткость

---

- Дисциплина называется «нечёткая логика», но логика – лишь часть
- На самом деле это:
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



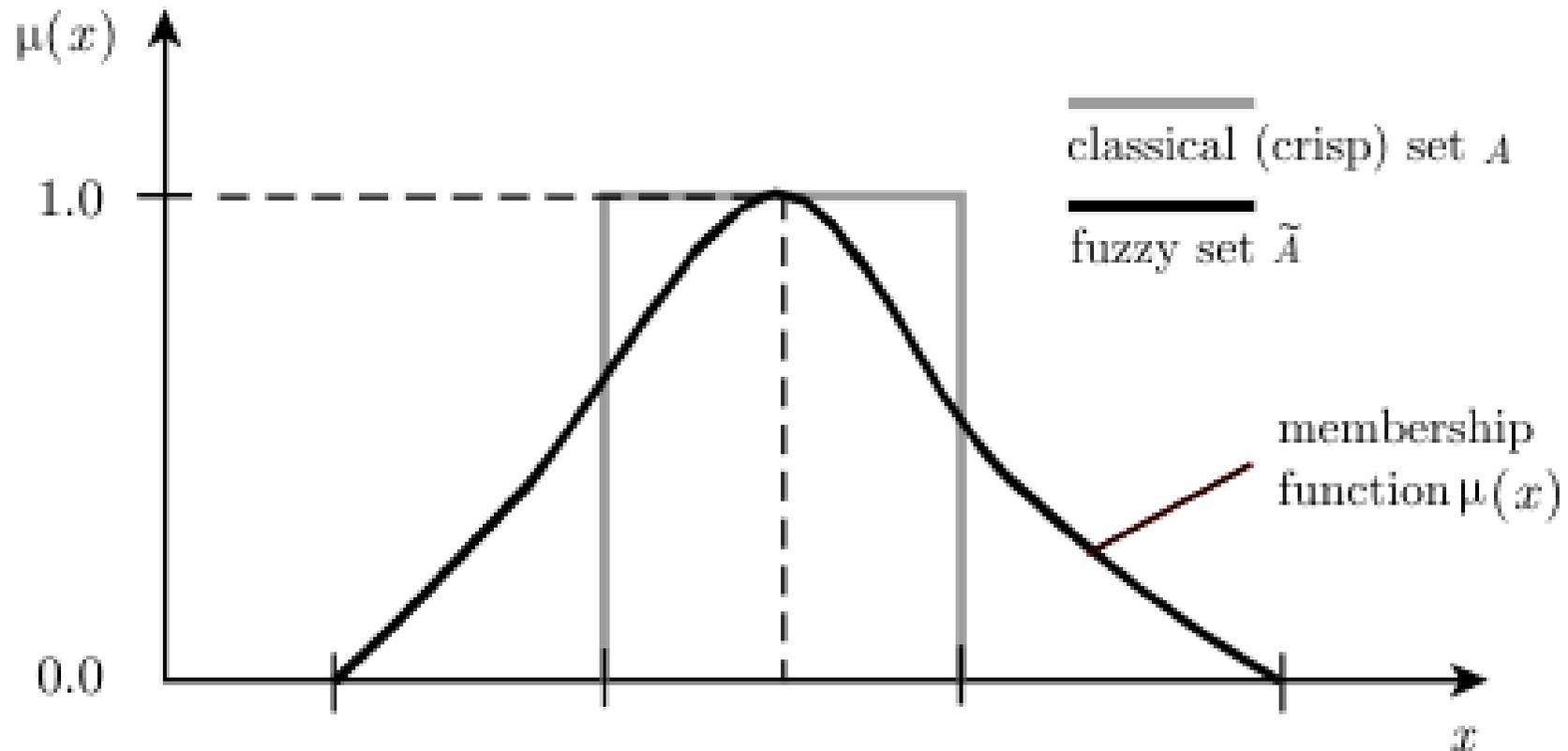
# План лекции

---

- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры

# Нечёткая теория множеств

## ■ Нечёткое множество:



# Нечёткая теория множеств

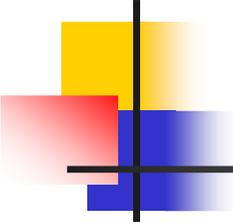
- Нечёткое число – нечёткое множество, у которого функция принадлежности достигает единицы только в одной точке.



# Нечёткая теория множеств

- Нечёткий интервал – нечёткое множество, у функции принадлежности которого есть центральный сегмент, равный 1.

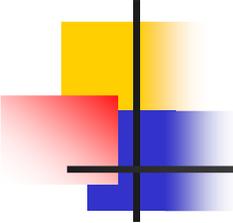




# Нечёткая теория множеств

---

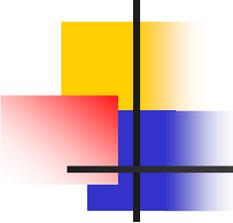
- Нужно научиться:
  - пересекать и объединять нечёткие множества, брать дополнения;
  - складывать и умножать нечёткие числа;
  - работать со множествами уровня и сечениями нечётких множеств.



# План лекции

---

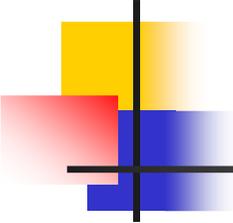
- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - **Нечёткая логика**
  - Нечёткая теория меры
  - Нечёткие контроллеры



# Нечёткая логика

---

- В обычной логике формула либо истинна, либо нет – множество значений ф-ции истинности  $\{0,1\}$ .
- В нечёткой логике формула может быть истинна на  $0,342$  – множество значений ф-ции истинности  $[0,1]$ .



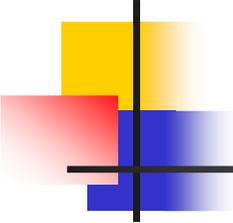
# Нечёткая логика

---

- Что такое логическая связка AND?
- В классической логике:

$\wedge$	0	1
0	0	0
1	0	1

$$\wedge : \{0,1\} \times \{0,1\} \rightarrow \{0,1\}$$



# Нечёткая логика

---

- Что такое логическая связка AND?
- В нечёткой логике:

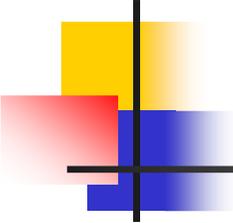
$$\wedge : [0,1] \times [0,1] \rightarrow [0,1]$$

- Свойства:

$$1 \wedge x = x \qquad (x \wedge y) \wedge z = x \wedge (y \wedge z)$$

$$x \wedge y = y \wedge x \qquad v \leq w, x \leq y \Rightarrow v \wedge x \leq w \wedge y$$

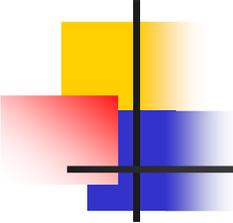
- Такие функции называются *триангулярными нормами* (t-нормами)



# Нечёткая логика

---

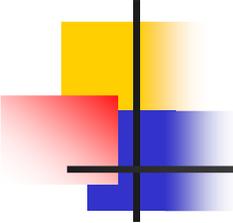
- Дуальное понятие – *триангулярная конорма* (t-конорма) – это нечёткий OR
- Отрицание – это инволюция на  $[0,1]$ :
  - $\eta(0) = 1, \quad \eta(1) = 0$  ;
  - $\eta$  не возрастает;
  - $\eta(\eta(x)) = x$  .
- Нужно изучить их свойства, установить нечёткие законы де Моргана и т.д.



# План лекции

---

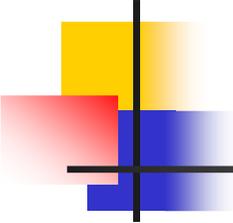
- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



# Нечёткая теория меры

---

- Допустим, мы хотим узнать, принадлежит ли данная точка  $x$  данному множеству  $A$ .
- Эксперты говорят «вроде да», присваивают некоторую степень доверия утверждению « $x$  лежит в  $A$ » для каждого множества  $A$ .
- Вот и получается мера  $\mu(A)$ .



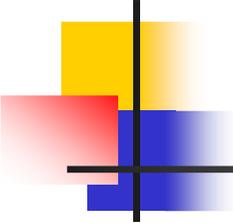
# Нечёткая теория меры

---

- Это похоже на вероятностную меру, но нечёткая мера не обязана быть аддитивной. Она может быть *МОНОТОННОЙ*, т.е. удовлетворять ф-ле включений-исключений; напр. монотонность порядка 2:

$$\mu(A \cup B) \geq \mu(A) + \mu(B) - \mu(A \cap B)$$

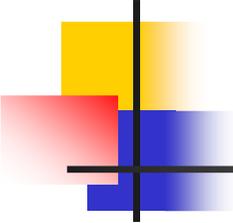
- И, наконец, ради чего создавались эти меры...



# Нечёткая теория меры

---

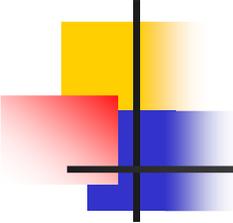
- ...нечёткие интегралы!
- Интеграл Сугено (Sugeno)
  - Максимизация функций при ограничениях, заданных как нечеткие множества
- Интеграл Шоке (Choquet)
  - Поиск нижней границы «мат. ожидания» заданной функции при нечётких ограничениях



# Нечёткая теория меры

---

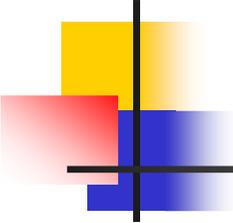
- Также мы, возможно, рассмотрим частные случаи нечётких мер:
  - меры доверия и правдоподобия (теория Демпстера-Шеффера);
  - теория возможности (possibility theory).
- Ключевые слова – *обработка знаний с неопределённостью.*



# План лекции

---

- Немного об AI
- Machine Learning
  - Деревья принятия решений
  - Нейронные сети
  - Генетические алгоритмы
  - Байесовские самообучающиеся системы
  - Байесовские сети доверия
- Нечеткая логика
  - Нечёткая теория множеств
  - Нечёткая логика
  - Нечёткая теория меры
  - Нечёткие контроллеры



# Нечёткие контроллеры

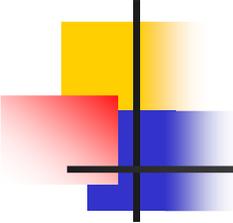
---

- Обычные контроллеры, как правило, решают дифференциальные уравнения, а система описывается как динамическая система:

$$\dot{x} = f(x, u, t),$$

где  $u$  – воздействие контроллера на систему.

- Надо найти такую функцию  $u(x, t)$ , при которой достигается заданная цель.



# Нечёткие контроллеры

---

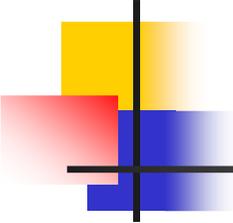
- Простой пример: машина едет по прямой.
  - $u_1$  – газ,
  - $u_2$  – тормоз.

- Получаем систему диффузов:

$$\dot{x}_1 = x_2,$$

$$\dot{x}_2 = u_1 - u_2,$$

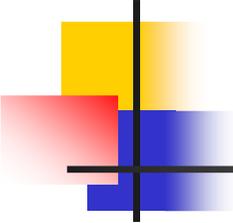
где  $x_1$  – положение машины,  $x_2$  – скорость.



# Нечёткие контроллеры

---

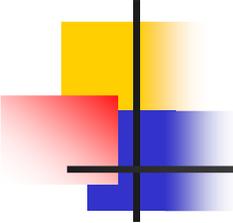
- Проблемы:
  - может быть не понятно, как аналитически описать целевую функцию (например, цель – припарковать машину);
  - могут быть не полностью или неточно заданы начальные условия (положение машины определяется по данным зрительных сенсоров);
  - могут быть неочевидны законы, по которым контрольное воздействие влияет на систему.



# Нечёткие контроллеры

---

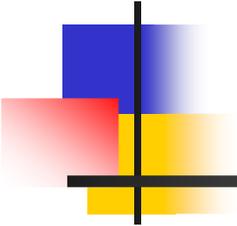
- Решение – нечёткие контроллеры.
- Вместо диффузов – нечёткие утверждения:
  - «Если машина близко к стене, надо жать на тормоз»;  
«близко» - нечёткость!
- Получается набор нечётких правил «если... то...», из которых методами нечёткой логики выводятся нечёткие заключения, а затем из них (обычно интегрированием) получаются конкретные «чёткие» ответы – насколько сильно надо нажать на тормоз.



# Нечёткие контроллеры

---

- Применяются очень много где
  - Автомшины
  - Бытовая техника
  - Мультиагентные системы (MASSIVE)
- Но об этом не очень любят говорить



Спасибо за внимание!

---