# Методы искусственного интеллекта

Лекция 3. Методы рассуждений с учетом неопределенности

# Неопределенность

**В реальности информация может быть:**

- Неполной
- Противоречивой
- Ненадёжной

**Неопределённость** – нехватка точных знаний, которые бы позволили прийти к достоверному и надёжному заключению.

**Медицинская диагностика:** невозможность построения (на данном этапе развития медицинской науки) строгой, «математической» теории функционирования человеческого тела; невозможность ввода в интеллектуальную систему всех фактов, касающихся текущего состояния.

**Формальные методы обработки неопределенности** – инструмент, облегчающий рассуждение в таких «сложных» областях.

# Источники неопределенности

- Неизвестные данные
- Неточность естественного языка
- Слабые закономерности
- Сочетание взглядов различных экспертов

# Тема 6. Ранние методы (схема Шортлиффа-Бьюкенена)

# Коэффициенты уверенности

**…известные также как теория Шортлиффа-Бучанана (Shortliffe-Buchanan), схема Шортлиффа.**
Впервые применены в экспертной системе медицинской диагностики MYCIN.

**Причины:**

- Выражение экспертами уверенности в тех или иных закономерностях способом, не согласованным со строгой математической теорией
- Недостаток достоверных статистических данных (в частности, в области медицинской диагностики)

# Коэффициенты уверенности

**ЕСЛИ** *свидетельство E*
**ТО** *гипотеза H* {cf}

**Коэффициент уверенности cf** (certainty factor) представляет собой степень уверенности в том, что гипотеза *H* при условии свидетельства *E*.
Изменяется в пределах от -1 (при соблюдении всех условий существует полная уверенность в ошибочности заключения) до +1 (полная уверенность в правильности заключения).

**Пример:**

**ЕСЛИ** *Высокая_температура* & *Кашель* & *Эпидемия_гриппа*
**ТО** *Грипп* {cf = 0.8}

# Возможная интерпретация коэффициентов уверенности

| Характеристика | Коэффициент уверенности |
|---|---|
| Definitely not | -1.0 |
| Almost certainly not | -0.8 |
| Probably not | -0.6 |
| Maybe not | -0.4 |
| Unknown | -0.2 до 0.2 |
| Maybe | +0.4 |
| Probably | +0.6 |
| Almost certainly | +0.8 |
| Definitely | +1.0 |

# Меры доверия и недоверия

***MB(H,E)*** – (measure of belief, мера доверия) – характеризует степень, в которой *доверие* к гипотезе *H* увеличится, при поступлении свидетельства *E*.
***MD(H,E)*** – (measure of disbelieve, мера недоверия) – характеризует степень, в которой увеличится *недоверие* к гипотезе *H* при поступлении свидетельства *E*.
Обе меры принадлежат диапазону [0; 1].

$$cf = \frac{MB(H,E) - MD(H,E)}{1 - \min[MB(H,E), MD(H,E)]}$$

# Распространение коэффициентов уверенности

Базовый принцип определения степени уверенности в консеквенте правила при его срабатывании:

$$cf(H,E) = cf(E) \times cf$$

**Правило:**
ЕСЛИ *ЛасточкиЛетаютНизко*
ТО *БудетДождь* { *cf=0.85* }

**При этом:**
*cf(ЛасточкиЛетаютНизко)* = 0.9

**Заключаем:**

$$cf(\text{БудетДождь}, \text{ЛасточкиЛетаютНизко}) = 0.9 \times 0.85 = 0.765$$

# Несколько свидетельств (конъюнкция)

ЕСЛИ $E_1$ И … И $E_n$
ТО $H$ { *cf* }

$$cf(H, E_1 \cap E_2 \cap \cdots \cap E_n) = \min[cf(E_1), \dots, cf(E_n)] \times cf$$

**Правило:**
ЕСЛИ *ЛасточкиЛетаютНизко* И *ДымСтелетсяПоЗемле*
ТО *БудетДождь* { *cf=0.9* }

**При этом:**
*cf(ЛасточкиЛетаютНизко)* = 0.9
*cf(ДымСтелетсяПоЗемле)* = 0.8

**Заключаем:**

$$cf(\text{БудетДождь}, \text{ЛасточкиЛетаютНизко} \cap \text{ДымСтелетсяПоЗемле}) = \min[0.9, 0.8] \times 0.9$$
$$= 0.8 \times 0.9 = 0.72$$

# Несколько свидетельств (дизъюнкция)

ЕСЛИ $E_1$ ИЛИ … ИЛИ $E_n$
ТО $H$ { *cf* }

$$cf(H, E_1 \cup E_2 \cup \dots \cup E_n) = \max[cf(E_1), \dots, cf(E_n)] \times cf$$

**Правило:**
ЕСЛИ *ЛасточкиЛетаютНизко* ИЛИ *ДымСтелетсяПоЗемле*
ТО *БудетДождь* { *cf = 0.9* }

**При этом:**
*cf(ЛасточкиЛетаютНизко)* = 0.9
*cf(ДымСтелетсяПоЗемле)* = 0.8

**Заключаем:**

$$cf(\text{БудетДождь}, \text{ЛасточкиЛетаютНизко} \cup \text{ДымСтелетсяПоЗемле}) = \max[0.9, 0.8] \times 0.9$$
$$= 0.9 \times 0.9 = 0.81$$

# Несколько правил

Два правила с коэффициентами уверенности $cf_1$ и $cf_2$ относительно одной гипотезы:

$$cf(cf_1, cf_2) = \begin{cases} cf_1 + cf_2 \times (1 - cf_1) & \text{если } cf_1 > 0 \text{ и } cf_2 > 0 \\ \dfrac{cf_1 + cf_2}{1 - \min[|cf_1| + |cf_2|]} & \text{если } cf_1 < 0 \text{ или } cf_2 < 0 \\ cf_1 + cf_2 \times (1 - cf_1) & \text{если } cf_1 < 0 \text{ и } cf_2 < 0 \end{cases}$$

**Правила:**

ЕСЛИ *ЛасточкиЛетаютНизко* ТО *БудетДождь* { *cf = 0.85* }

ЕСЛИ *ДымСтелетсяПоЗемле* ТО *БудетДождь* { *cf = 0.9* }

**При этом:**

*cf(ЛасточкиЛетаютНизко)* = 1.0

*cf(ДымСтелетсяПоЗемле)* = 0.7

**Заключаем:**

*cf(БудетДождь)* = (1 x 0.85) + (0.7 x 0.9) x (1 – (1 x 0.85)) = 0.9445

# Тема 7. Байесовские сети

# Представление неопределённости с использованием теории вероятностей

Знания агента позволяют сформировать относящиеся к делу высказывания только с определенной **степенью уверенности (degree of belief)**.

Вероятности предоставляют способ *суммарного учета неопределенности*, возникающей по причинам *экономии усилий и отсутствия знаний*.

**Степень уверенности ≠ степень истинности:**
Высказывание истинно или ложно, степень уверенности выражает лишь уверенность агента в истинности высказывания с учётом имеющейся у него информации.

**Основа уверенности:** статистические данные, правила, комбинация сведений, полученных из разных источников.

# Свидетельства

Вероятность высказывания ≈ утверждение о том, что высказывание следует из БЗ.

Оценка этого *может изменяться* по мере добавления в базу знаний новых высказываний.

Во всех вероятностных утверждениях должно быть указано **свидетельство**, с учётом которого оценивалась данная вероятность.

**Априорные (безусловные)** вероятности – до получения свидетельств.
**Апостериорные (условные) вероятности** – после получения свидетельств.

# Язык теории вероятностей

Вероятность применяется к **высказываниям** – утверждениям о том, что имеет место некоторое состояние мира. Язык высказываний – язык теории вероятностей.

**Случайная величина (переменная)** – обозначение некоторой части мира, значение которой неизвестно. (*OilDeposit*, *Cavity*, *Rain*)

**Область определения** – множество тех значений, которые может принимать случайная величина. (<*true*, *false*>)

**Элементарные высказывания** – утверждения о том, что случайная величина принимает одно из значений своей области определения (*OilDeposit* = *true*).

**Сложные высказывания** – комбинации элементарных высказываний с использованием логических связок ($Cavity = true \wedge Rain = false$ или, по-другому, $cavity \wedge \neg rain$).

# Априорная и апостериорная вероятности

**Безусловная**, или **априорная**, вероятность P(*a*), связанная с высказыванием *a*, представляет собой степень уверенности, относящуюся к этому высказыванию в *отсутствии любой другой информации*.
P(*Cavity = true*) = 0.1 или P(*cavity*) = 0.1

Для всех возможных значений СВ – **распределение априорных вероятностей СВ**.
Для нескольких СВ – **совместные распределения вероятностей**.
Для всех СВ модели – **полное совместное распределение вероятностей**.

**Условная**, или **апостериорная**, вероятность P(*a*|*b*) представляет собой степень уверенности в том, что истинно высказывание *a*, если *всё, что нам известно*, это *b*.

# Логический вывод с использованием полных совместных распределений

**Вероятностный вывод** - вычисление апостериорных вероятностей для высказываний, заданных в виде запросов, на основании наблюдаемых свидетельств.

Простейший вариант: БЗ – полное совместное распределение.

$$P(X|e) = \alpha P(X,e) = \alpha \sum_{y} (X,e,y)$$

$X$ – переменная запроса;
$E$ – множество переменных свидетельства;
$e$ – наблюдаемые значения переменных свидетельства;
$y$ – возможные значения ненаблюдаемых переменных из множества $Y$;
P($X|e$) – запрос;
$\alpha$ – константа нормализации ($1/P(e)$).

# Пример вывода с использованием полного совместного распределения (1)

| | *influenzaEpidemic* | | *not influenzaEpidemic* | |
|---|---|---|---|---|
| | *fever* | *not fever* | *fever* | *not fever* |
| *influenza* | 0.05 | 0.001 | 0.02 | 0.001 |
| *not influenza* | 0.009 | 0.04 | 0.33 | 0.549 |

3 логические переменные:
- *Infuenza*
- *InfluenzaEpidemic*
- *Fever*

$\Sigma = 1$

# Пример вывода с использованием полного совместного распределения (2)

| | *influenzaEpidemic* | | *not influenzaEpidemic* | |
|---|---|---|---|---|
| | *fever* | *not fever* | *fever* | *not fever* |
| *influenza* | 0.05 | 0.001 | 0.02 | 0.001 |
| *not influenza* | 0.009 | 0.04 | 0.33 | 0.549 |

→ 0.072

Маргинальная вероятность

P(*influenza*) = 0.05 + 0.001 + 0.02 + 0.001 = 0.072

# Пример вывода с использованием полного совместного распределения (3)

| | *influenzaEpidemic* | | *not influenzaEpidemic* | |
|---|---|---|---|---|
| | *fever* | *not fever* | *fever* | *not fever* |
| *influenza* | 0.05 | 0.001 | 0.02 | 0.001 |
| *not influenza* | 0.009 | 0.04 | 0.33 | 0.549 |

P(*influenza*) = 0.05 + 0.001 + 0.02 + 0.001 = 0.072

P(*Influenza* | *influenzaEpidemic*) = $\alpha$P(*Influenza*, *influenzaEpidemic*) =

= $\alpha$<0.051, 0.049> = <0.51, 0.49>

# Пример вывода с использованием полного совместного распределения (4)

| | *influenzaEpidemic* | | *not influenzaEpidemic* | |
|---|---|---|---|---|
| | *fever* | *not fever* | *fever* | *not fever* |
| *influenza* | 0.05 | 0.001 | 0.02 | 0.001 |
| *not influenza* | 0.009 | 0.04 | 0.33 | 0.549 |

P(*influenza*) = 0.05 + 0.001 + 0.02 + 0.001 = 0.072

P(*Influenza* | *influenzaEpidemic*) = $\alpha$P(*Influenza*, *influenzaEpidemic*) =

  = $\alpha<0.051, 0.049> = <0.51, 0.49>$

P(*Influenza* | *influenzaEpidemic* ∩ *fever*) =

  = $\alpha$P(*Influenza*, *influenzaEpidemic*, *fever*) =

  = $\alpha<0.05, 0.009> \approx <0.85, 0.15>$

# Краткий анализ

**Полезный вывод:** полное совместное распределение *может* быть использовано для получения ответа на *любые* вероятностные запросы.

Но **совершенно непрактично:**
- Объем $2^n$ значений, где n – количество переменных (СВ);
- Обработка таблицы – $O(2^n)$.

# Правило Байеса

$$P(b|a) = \frac{P(a|b)P(b)}{P(a)}$$

*Три* вероятности для вычисления *одной*. В чём выигрыш?

# Виды информации (в системах диагностики)

**Диагностическая информация** – наличие симптома *a* свидетельствует в пользу (заболевания, неисправности) *b*. Встречается *редко*, подвержена влиянию внешних условий.

**Причинная информация** – (заболевание, неисправность) *b* с определённой вероятностью вызывает симптом *a*. Встречается *часто*, менее подвержена влиянию внешних условий.

Причинная

$$P(\text{заболевание}|\text{симптом}) = \frac{P(\text{симптом}|\text{заболевание})P(\text{заболевание})}{P(\text{симптом})}$$

Диагностическая

# Комбинирование свидетельств

Что, если нам известно несколько свидетельств (симптомов)?

$$P(Flu|fever = true \wedge sorethroat = true)$$

Применение правила Байеса даёт:

$$P(Flu|fever \wedge sorethroat) = \frac{P(fever \wedge sorethroat|Flu)P(Flu)}{P(fever \wedge sorethroat)}$$

**Но! Непосредственное применение этого правила требует определения условных вероятностей для всех возможных сочетаний свидетельств!**

# Комбинирование свидетельств. Независимость

**Независимость** СВ:

$$P(a|b) = P(a)$$
$$P(a \wedge b) = P(a)P(b)$$

**Условная независимость** СВ:

$$P(a \wedge b|c) = P(a|c)P(b|c)$$

т.е. *a* и *b* являются независимыми, при условии, что задано значение некоторой третьей СВ.
Чаще всего, и *a*, и *b*, зависят от *c* но не зависят друг от друга.

Из предположения об условной независимости *fever* и *sorethroat*:

$$P(Flu|fever \wedge sorethroat) = \alpha P(fever|Flu)P(sorethroat|Flu)P(Flu)$$

# Наивная байесовская модель

$$P(Cause, E_1, \dots, E_n) = P(Cause)\prod_i P(E_i|Cause)$$

Часто применяется и в том случае, когда $E_i$ не являются условно-независимыми.

Например, спам-фильтрация:

$Cause$ – спам или не спам, $E_i$ - наличие определенного слова в словаре.

# Байесовская сеть

**Байесовская сеть** — это ориентированный граф, в котором каждая вершина помечена количественной вероятностной информацией.

1. Вершинами сети является множество случайных переменных. Переменные могут быть дискретными или непрерывными.

2. Вершины соединяются попарно ориентированными ребрами; ребра образуют множество ребер. Если стрелка направлена от вершины $X$ к вершине $Y$, то вершина $X$ называется родительской вершиной вершины $Y$.

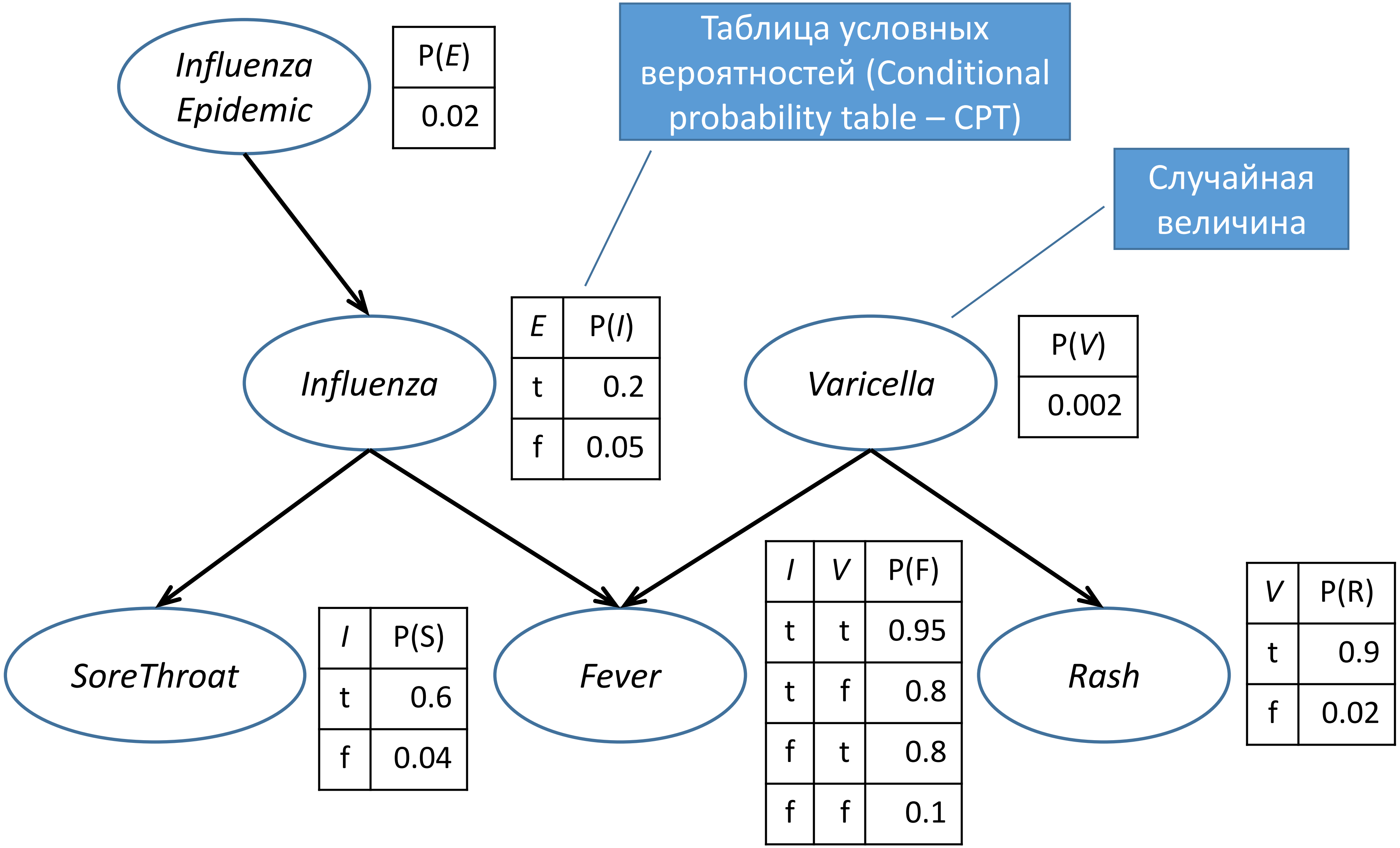
3. Каждая вершина $X_i$ характеризуется распределением условных вероятностей $P(X_i \mid Parents(X_i))$, которое количественно оценивает влияние родительских вершин на эту вершину.

4. Граф не имеет контуров (Directed Acyclic Graph — DAG).

# Байесовская сеть

**Топология** сети показывает отношения, определяющие условную независимость, которые проявляются в данной проблемной области. *Интуитивный* смысл стрелки в правильно составленной сети обычно состоит в том, что вершина X оказывает *непосредственное* влияние на вершину Y.

# Пример байесовской сети




| P(*E*) |
|---|
| 0.02 |

| *E* | P(*I*) |
|---|---|
| t | 0.2 |
| f | 0.05 |

| P(*V*) |
|---|
| 0.002 |

| *I* | P(S) |
|---|---|
| t | 0.6 |
| f | 0.04 |

| *I* | *V* | P(F) |
|---|---|---|
| t | t | 0.95 |
| t | f | 0.8 |
| f | t | 0.8 |
| f | f | 0.1 |

| *V* | P(R) |
|---|---|
| t | 0.9 |
| f | 0.02 |

# Представление полного совместного распределения

Каждый элемент в полном совместном распределении вероятностей может быть рассчитан на основании информации, представленной в байесовской сети.

$$P(x_1, \dots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

где *parents($X_i$)* обозначает конкретные значения переменных в множестве вершин *Parents($X_i$)*. Поэтому каждый элемент в совместном распределении представлен в виде произведения соответствующих элементов в таблицах условных вероятностей (Conditional Probability Table — CPT) байесовской сети.

Таким образом, таблицы CPT обеспечивают **декомпонованное представление совместного распределения**.

# Составление байесовской сети

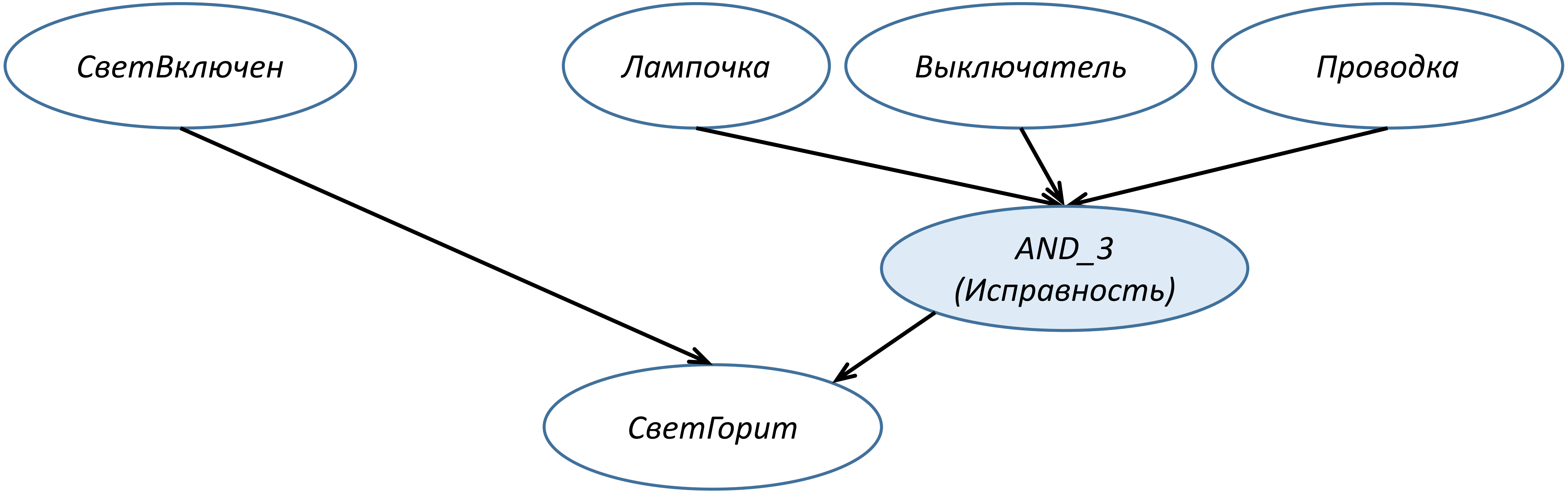
Байесовская сеть служит правильным представлением проблемной области, только если *каждая вершина в ней условно независима от ее предшественников в конкретном упорядочении вершин, после того как заданы ее родительские вершины*.

Необходимо выбрать для каждой вершины родительские вершины так, чтобы соблюдалось это свойство. Интуитивно ясно, что множество родительских вершин вершины $X_i$ должно включать все такие вершины из множества $X_1,...,X_{i-1}$, которые *непосредственно* влияют на $X_i$.

Правильный порядок вершин (от причин к следствиям). Правила должны быть *причинными*, а не *диагностическими*.

# Составление байесовской сети. Детерминированные узлы

Например, логические связки, функции. Многие инструменты позволяют определять вставлять такие узлы напрямую, сокращая количество параметров.



# Точный вероятностный вывод в байесовских сетях

Основной задачей для любой системы вероятностного вывода является вычисление распределения апостериорных вероятностей для множества **переменных запроса**, если дано некоторое наблюдаемое **событие**, т.е. если выполнено некоторое присваивание значений множеству **переменных свидетельства**.

**Система обозначений:**
$X$ – обозначает переменную запроса;
$E$ — множество переменных свидетельства, $E_1,..., E_m$;
$e$ — конкретное наблюдаемое событие;
$Y$ обозначает переменные, отличные от переменных свидетельства, $Y_1, ..., Y_l$ (иногда называемые **скрытыми переменными**).
Полное множество переменных определяется выражением $X=\{X\} \cup E \cup Y$.

В типичном запросе  содержится просьба определить распределение апостериорных вероятностей $P(X \mid e)$.

# Вероятностный вывод с помощью перебора

**Предпосылки:**

1. Условную вероятность можно вычислить суммируя элементы из полного совместного распределения:

$$P(X|e) = \alpha P(X,e) = \alpha \sum_{y} P(X,e,y)$$

2. Байесовская сеть является представлением полного совместного распределения:

$$P(x_1, \dots, x_n) = \prod_{i=1}^{n} P(x_i|parents(X_i))$$

**Идея:**

Вычисляем результат запроса, суммируя произведения условных вероятностей из сети.

# Алгоритм перебора для получения ответов на запросы в байесовских сетях

```
function EnumerationAsk(X, e, bn) returns распределение по X
  inputs: X, переменная запроса
          e, наблюдаемые значения переменных E
          bn, байесовская сеть с переменными
              {X} ∪ E ∪ Y /* Y - скрытые переменные */
  Q(X) <– распределение по X, первоначально пустое
  for each значение x_i переменной X do
    дополнить e значением x_i переменной X
    Q(x_i) <– EnumerateAll(Vars[bn], e)
  return Normalize(Q(X))

function EnumerateAll(vars, e) returns действительное число
  if Empty?(vars) then return 1.0
  Y <– First(vars)
  if переменная Y имеет значение y в множестве e
    then return P(y|parents(Y))EnumerateAll(Rest(vars),e)
    else return ∑_y P(y|parents(Y)) EnumerateAll(Rest(vars),e_y),
         где e_y представляет собой множество e, дополненное
         значением Y = y
```

# Сложность точного вероятностного вывода

**Односвязная сеть** (**полидерево**) – сеть, в которой имеется самое большее один неориентированный путь между любыми двумя вершинами.

Временная и пространственная сложность точного вероятностного вывода в полидеревьях *линейно зависят от размера сети* (количество элементов таблиц CPT).
Например, алгоритм устранения переменной.

В общем случае (в многосвязных сетях) *вероятностный вывод является NP-трудным, поскольку включает вывод в пропозициональной логике как частный случай.*

# Приближённый вероятностный вывод

Например, применение методов выборки (**Монте-Карло**) для вычисления апостериорных вероятностей.
Приближенный алгоритм => точность ответа зависит от количества сформированных выборок.

**Идея метода непосредственной выборки:** выборка должна формироваться последовательно по каждой переменной, в топологическом порядке. Распределение вероятностей, из которого берется выборка значения, обуславливается значениями, уже присвоенными родительским переменным.

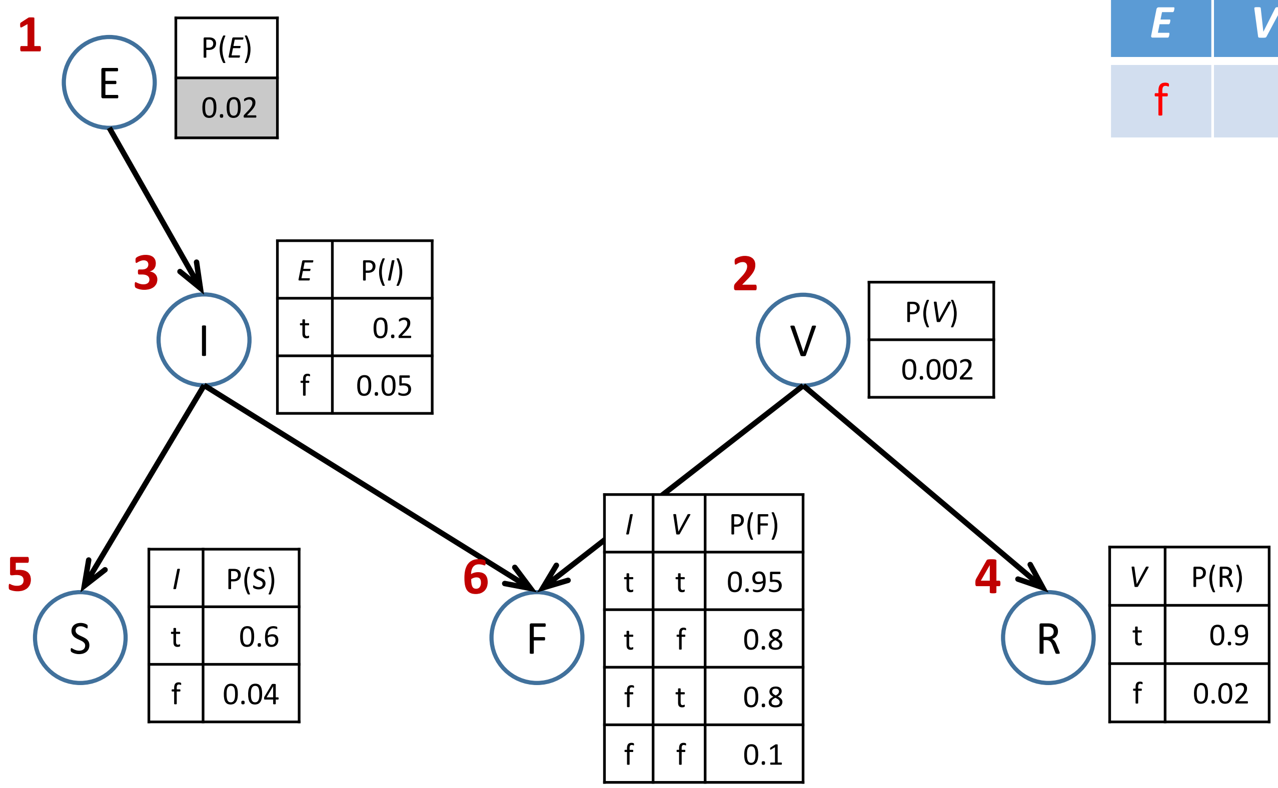
Ответы вычисляются путем подсчета фактически сформированных выборок.

# Формирование выборки из априорного распределения

```
function PriorSample(bn) returns событие, выработанное путем
         применения операции формирования выборки к априорному
         распределению, заданному в виде сети bn
  inputs: bn, байесовская сеть, задающая совместное
              распределение P(X1,...,Xn)
    x <– событие с n элементами
    for i = 1 to n do
      xi <– случайная выборка из P(Xi|parents(Xi))
    return x
```

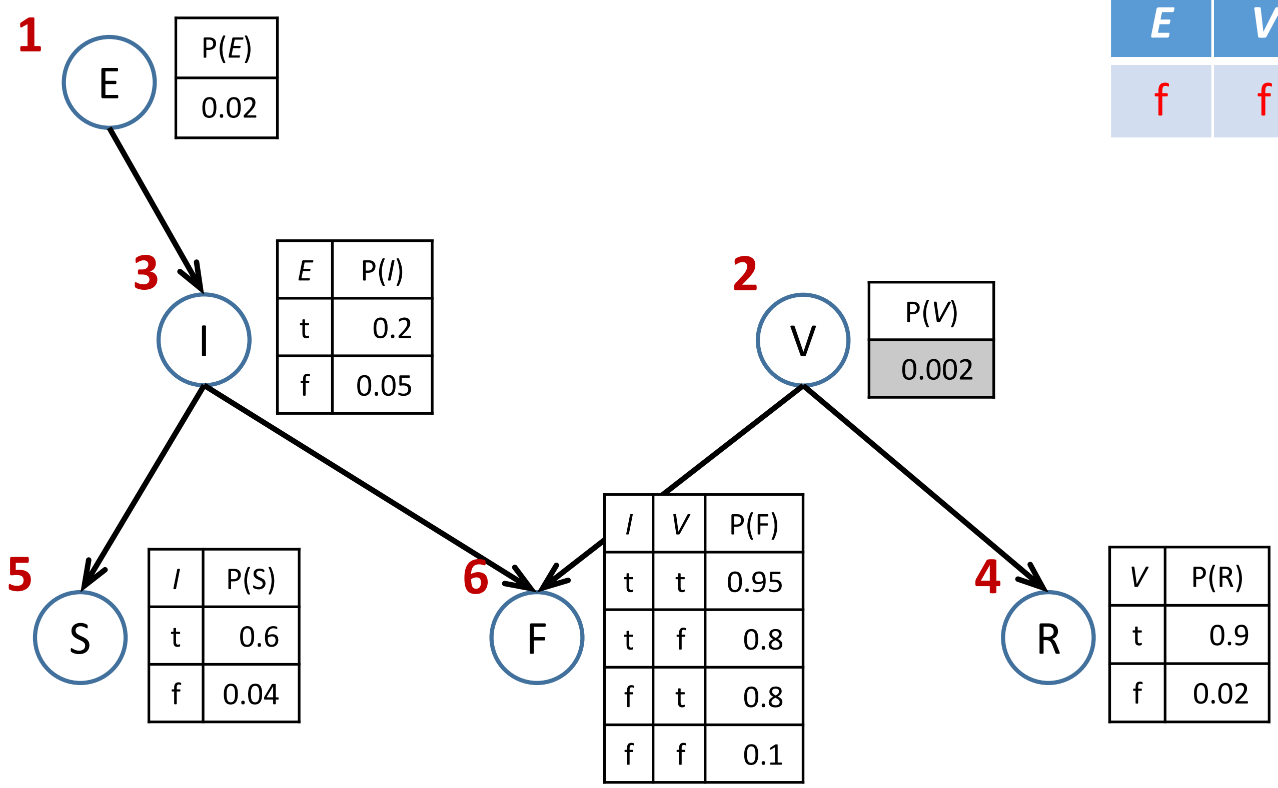
# Формирование выборки из априорного распределения

| *E* | *V* | *I* | *R* | *S* | *F* |
|---|---|---|---|---|---|
| f | | | | | |



# Формирование выборки из априорного распределения

| *E* | *V* | *I* | *R* | *S* | *F* |
|---|---|---|---|---|---|
| f | f | | | | |




| P(*E*) |
|---|
| 0.02 |

| *E* | P(*I*) |
|---|---|
| t | 0.2 |
| f | 0.05 |

| P(*V*) |
|---|
| 0.002 |

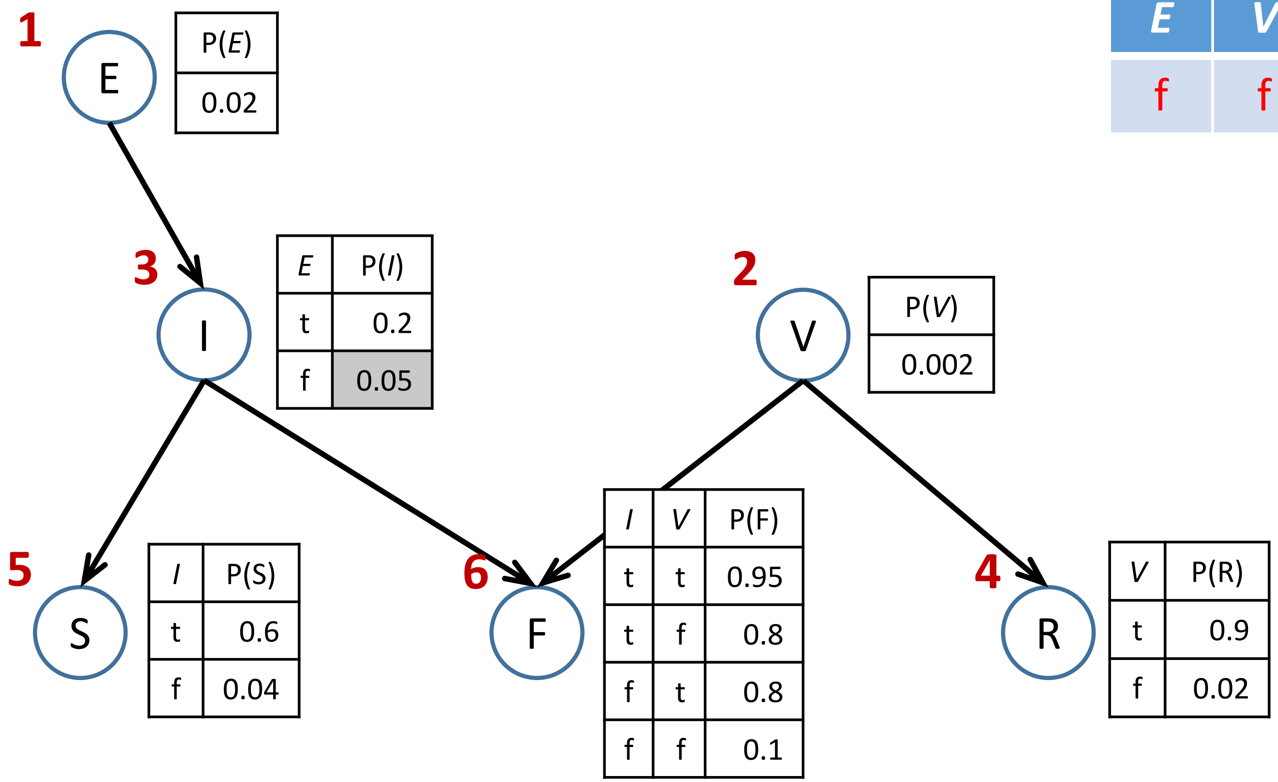
| *I* | P(S) |
|---|---|
| t | 0.6 |
| f | 0.04 |

| *I* | *V* | P(F) |
|---|---|---|
| t | t | 0.95 |
| t | f | 0.8 |
| f | t | 0.8 |
| f | f | 0.1 |

| *V* | P(R) |
|---|---|
| t | 0.9 |
| f | 0.02 |

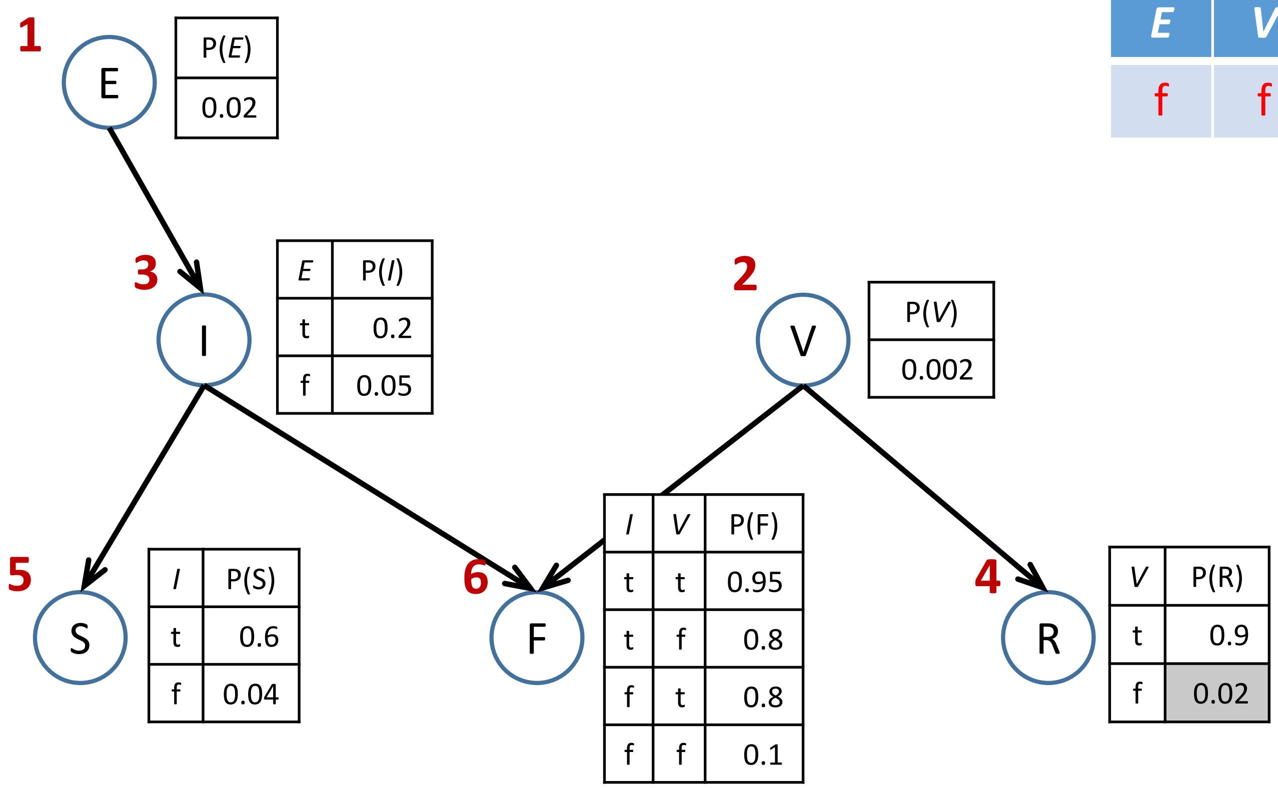
# Формирование выборки из априорного распределения

| *E* | *V* | *I* | *R* | *S* | *F* |
|---|---|---|---|---|---|
| f | f | f | | | |



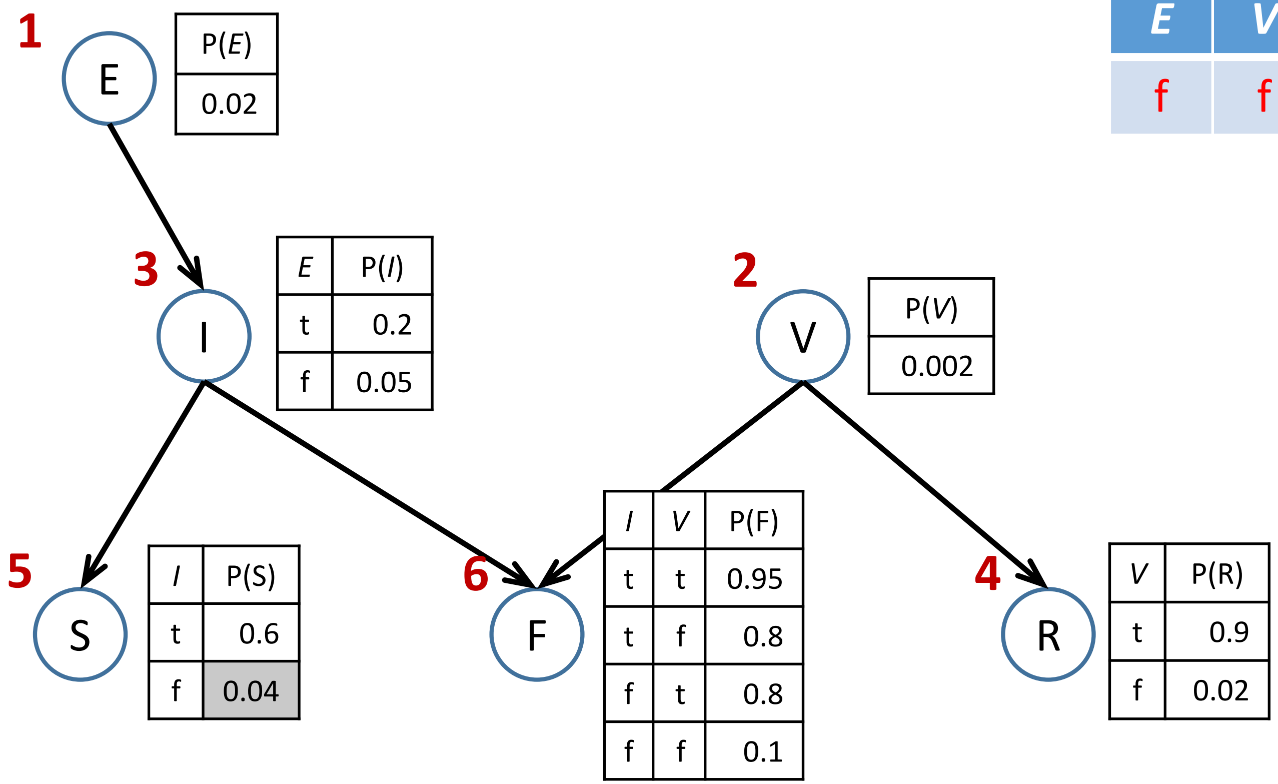
# Формирование выборки из априорного распределения

| *E* | *V* | *I* | *R* | *S* | *F* |
|---|---|---|---|---|---|
| f | f | f | f | | |



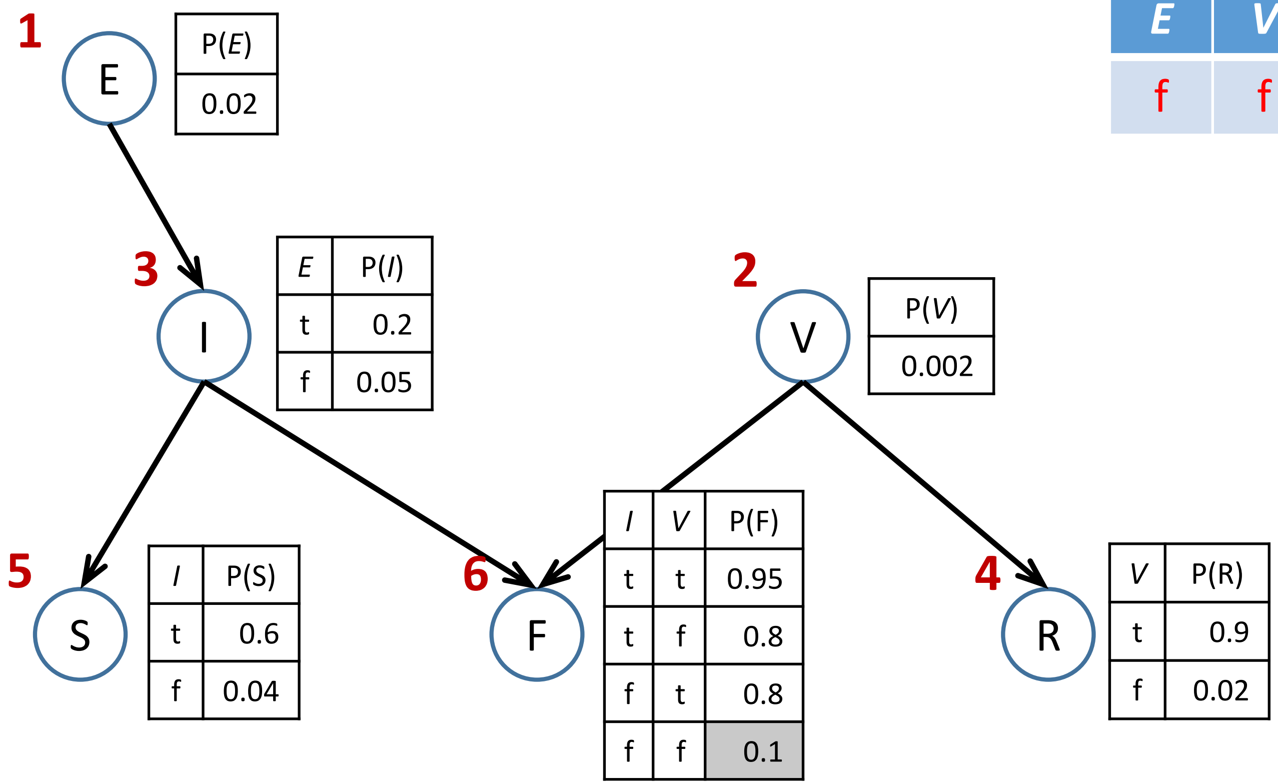
# Формирование выборки из априорного распределения

| *E* | *V* | *I* | *R* | *S* | *F* |
|---|---|---|---|---|---|
| f | f | f | f | f | |



# Формирование выборки из априорного распределения

| *E* | *V* | *I* | *R* | *S* | *F* |
|---|---|---|---|---|---|
| f | f | f | f | f | t |




| P(*E*) |
|---|
| 0.02 |

| *E* | P(*I*) |
|---|---|
| t | 0.2 |
| f | 0.05 |

| P(*V*) |
|---|
| 0.002 |

| *I* | P(S) |
|---|---|
| t | 0.6 |
| f | 0.04 |

| *I* | *V* | P(F) |
|---|---|---|
| t | t | 0.95 |
| t | f | 0.8 |
| f | t | 0.8 |
| f | f | 0.1 |

| *V* | P(R) |
|---|---|
| t | 0.9 |
| f | 0.02 |

# Формирование выборки с исключением

1) Формируем выборки из априорного распределения, определяемого сетью.
2) Исключаем все те выборки, которые не соответствуют свидетельству.
3) Формируем оценку *P(X=x|e)* путем подсчета того, насколько часто событие *X*=*x* встречалось в оставшихся выборках.

```
function Rejection-Sampling(X, e, bn, N) returns оценка
         значения P(X|e)
  inputs: X, переменная запроса
          e, свидетельство, определяемое как некоторое событие
          bn, байесовская сеть
          N, общее количество выборок, которые должны
             быть сформированы
  local variables: n, вектор результатов подсчетов по X,
                   первоначально равный нулю
  for j = 1 to N do
    x <- PriorSample(bn)
    if выборка x согласуется со свидетельством e then
      n[x] <- n[x]+1, где x представляет собой значение
                      переменной X в множестве x
    return Normalize(n[X])
```

# Формирование выборки с исключением

| *E* | *V* | *I* | *R* | *S* | *F* |
|---|---|---|---|---|---|
| f | f | f | f | f | t |
| f | f | f | f | f | f |
| f | f | f | f | f | t |
| t | f | t | f | t | t |
| f | f | f | f | f | f |
| t | f | f | f | f | f |
| f | t | f | t | f | t |
| f | f | f | f | f | f |

P(*Influenza* | *influenzaEpidemy* ∩ *fever*) = <1/1, 0/1> = <1.0, 0>

# Недостаток (Фатальный)

Приходится генерировать слишком много «лишних» выборок, доля выборок, согласованных со свидетельством *e*, уменьшается *экспоненциально* по мере увеличения количества переменных свидетельства.

# Оценка веса с учётом правдоподобия

1. Значения для переменных свидетельства *E* фиксируются и формируются выборки только для оставшихся переменных *X* и *Y*. Это позволяет гарантировать, что каждое выработанное событие будет согласованным со свидетельством.
2. Перед подведением итогов подсчетов в распределении для переменной запроса каждое событие взвешивается с учетом правдоподобия того, что событие согласуется со свидетельством.
3. Правдоподобие измеряется с помощью произведения условных вероятностей для каждой переменной свидетельства, если даны ее родительские переменные.

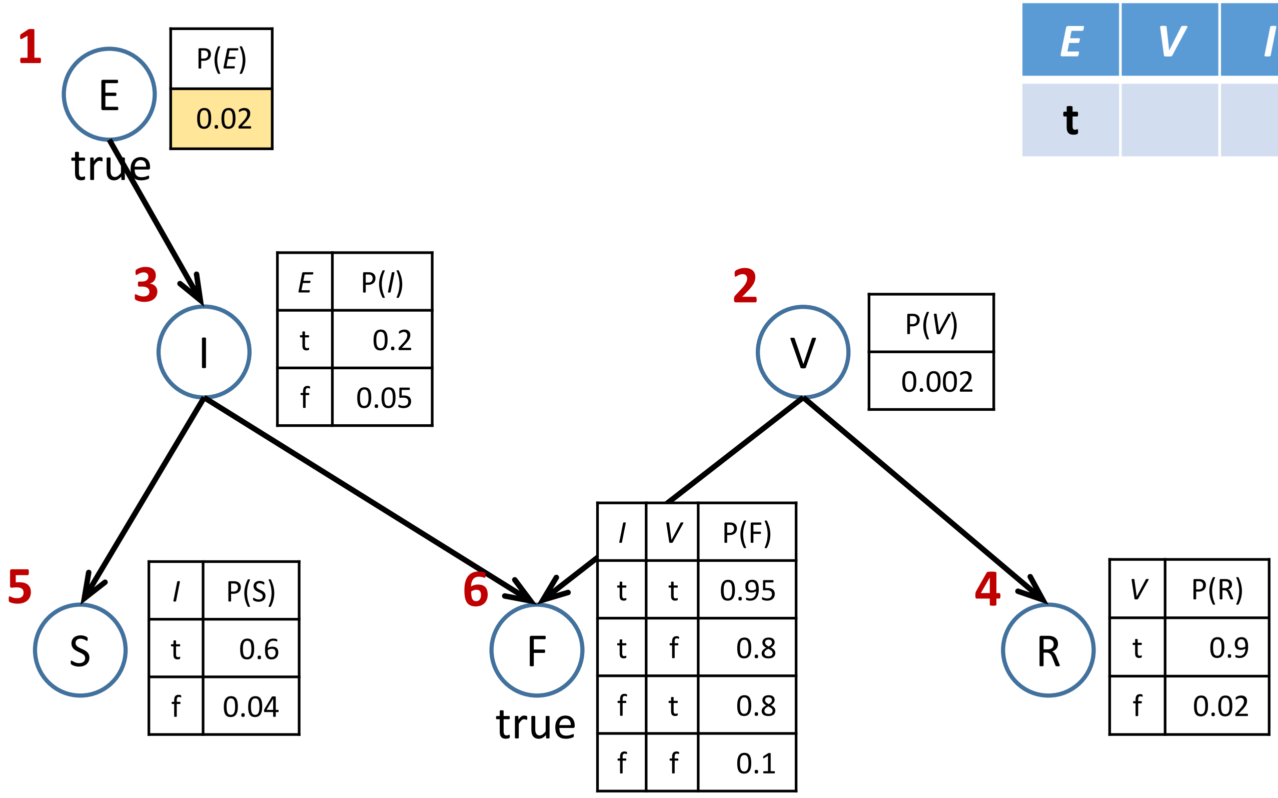
**Интуитивно**: события, в которых фактическое свидетельство кажется маловероятным, должны получать меньший вес.

# Оценка веса с учётом правдоподобия

```
function LikelihoodWeighting(X, e, bn, N) returns оценка
         значения P(X|e)
  inputs: X, переменная запроса
          e, свидетельство, определяемое как некоторое событие
          bn, байесовская сеть
          N, общее количество формируемых выборок
  local variables: W, вектор взвешенных результатов подсчетов по X,
                   первоначально равный нулю
  for j = 1 to N do
    x, w <– WeightedSample(bn)
    W[x] <– W[x] + w, где x представляет собой значение переменной X
                   в множестве x
  return Normalize(W[X])

function WeightedSample(bn, e) returns событие x и вес w
  x <– событие с n элементами; w <– 1
  for i = 1 to n do
    if Xi имеет значение xi in свидетельство e
    then w <– w * P(Xi=xi|parents(Xi))
    else xi <– случайная выборка из P(Xi|parents(Xi))
  return x, w
```

# Формирование взвешенной выборки



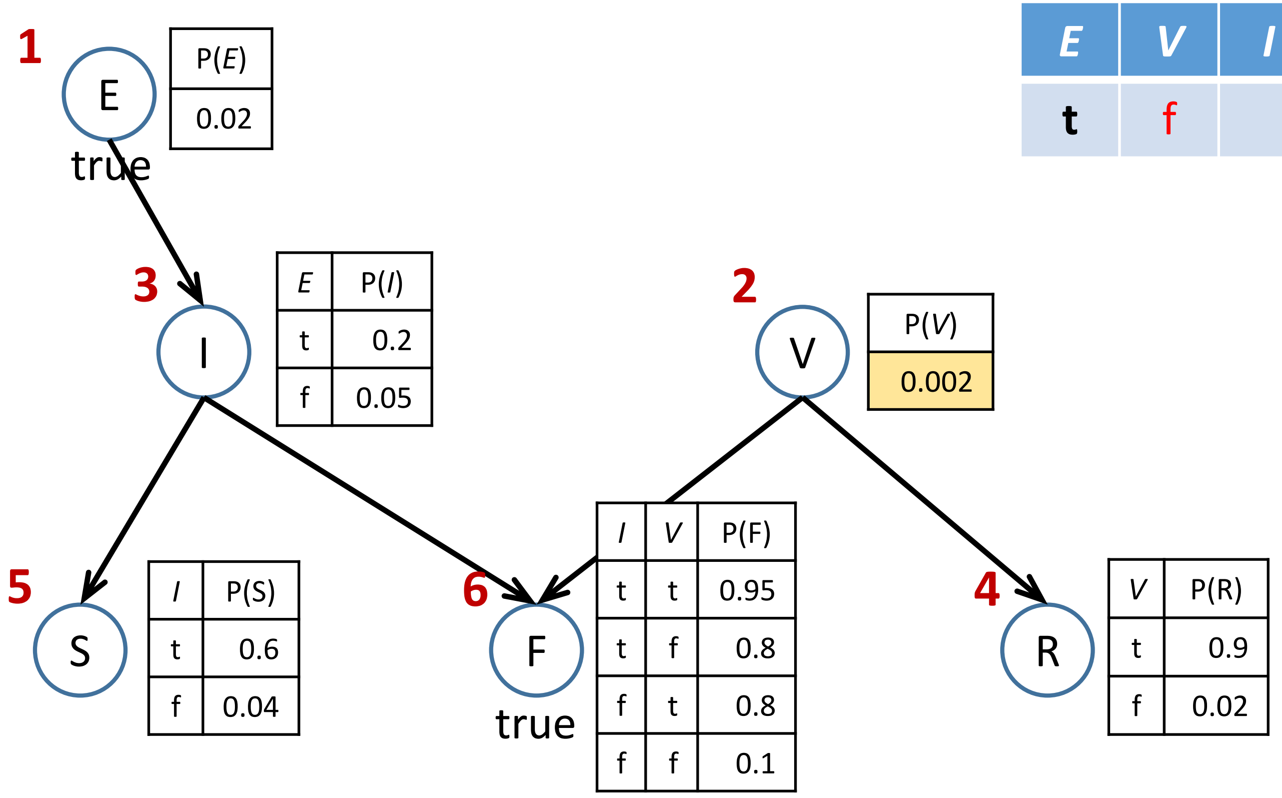

| *E* | *V* | *I* | *R* | *S* | *F* | *w* |
|---|---|---|---|---|---|---|
| **t** | | | | | **t** | |

w = 0.02

P(*Influenza* | *influenzaEpidemy* ∩ *fever*)

# Формирование взвешенной выборки

| *E* | *V* | *I* | *R* | *S* | *F* | *w* |
|---|---|---|---|---|---|---|
| **t** | f | | | | **t** | |

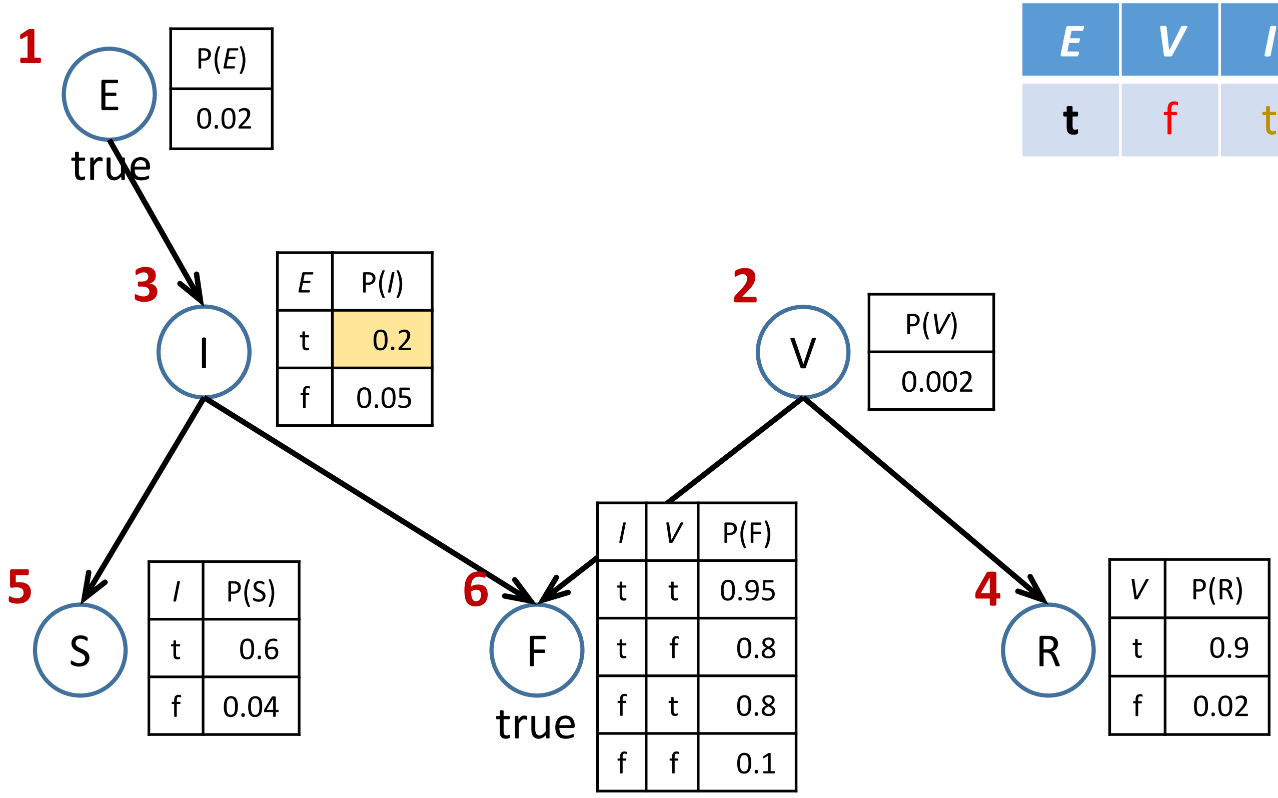



w = 0.02

P(*Influenza* | *influenzaEpidemy* ∩ *fever*)

# Формирование взвешенной выборки

| *E* | *V* | *I* | *R* | *S* | *F* | *w* |
|---|---|---|---|---|---|---|
| **t** | f | t | | | **t** | |

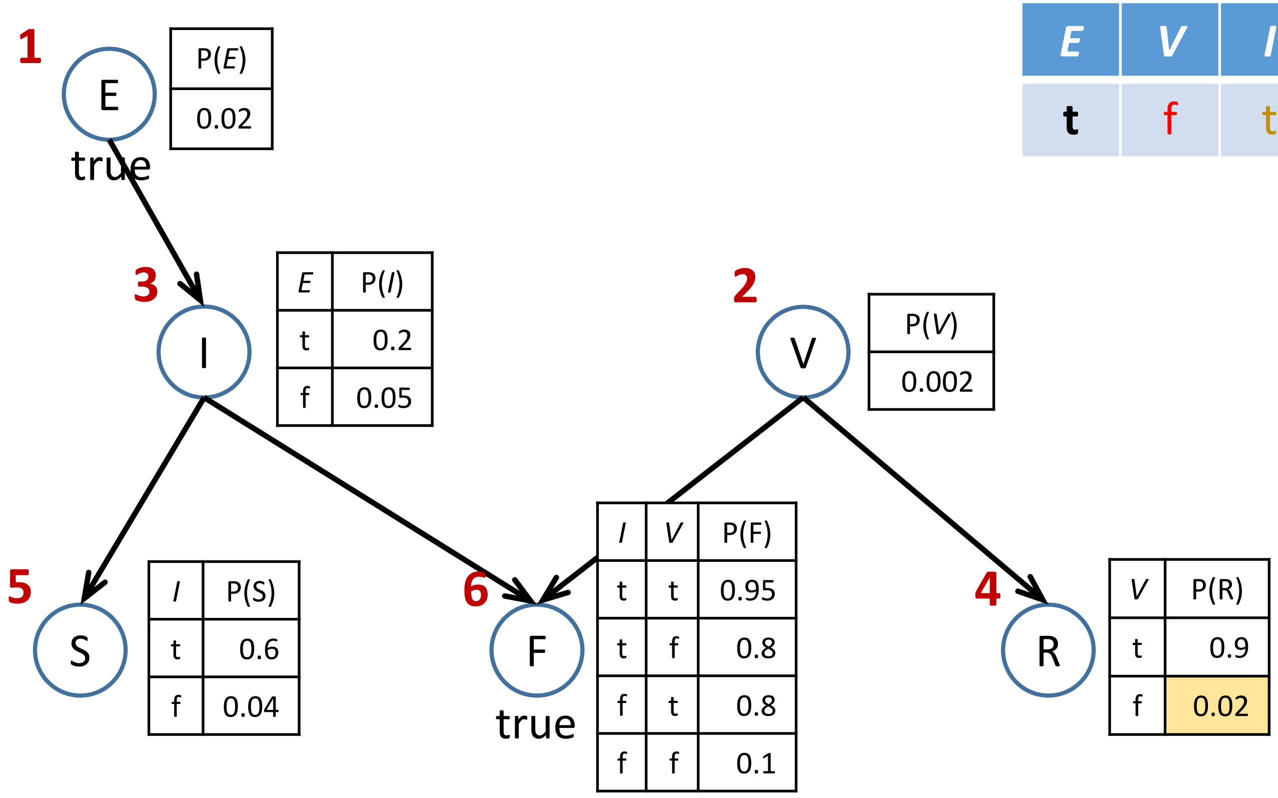



w = 0.02

P(*Influenza* | *influenzaEpidemy* ∩ *fever*)

# Формирование взвешенной выборки

| *E* | *V* | *I* | *R* | *S* | *F* | *w* |
|---|---|---|---|---|---|---|
| **t** | f | t | f | | **t** | |

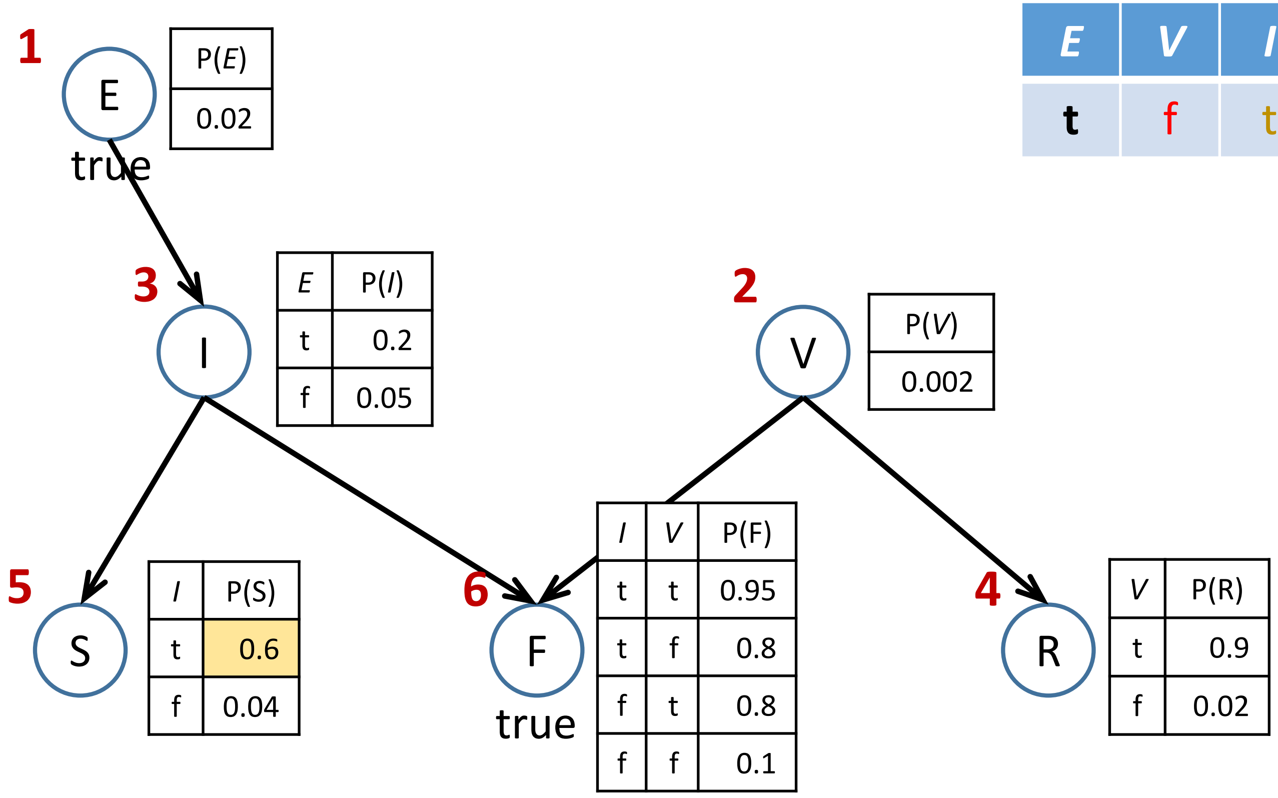



w = 0.02

P(*Influenza* | *influenzaEpidemy* ∩ *fever*)

# Формирование взвешенной выборки

| *E* | *V* | *I* | *R* | *S* | *F* | *w* |
|---|---|---|---|---|---|---|
| **t** | f | t | f | t | **t** | |




| P(*E*) |
|---|
| 0.02 |

| *E* | P(*I*) |
|---|---|
| t | 0.2 |
| f | 0.05 |

| P(*V*) |
|---|
| 0.002 |

| *I* | P(S) |
|---|---|
| t | 0.6 |
| f | 0.04 |

| *I* | *V* | P(F) |
|---|---|---|
| t | t | 0.95 |
| t | f | 0.8 |
| f | t | 0.8 |
| f | f | 0.1 |

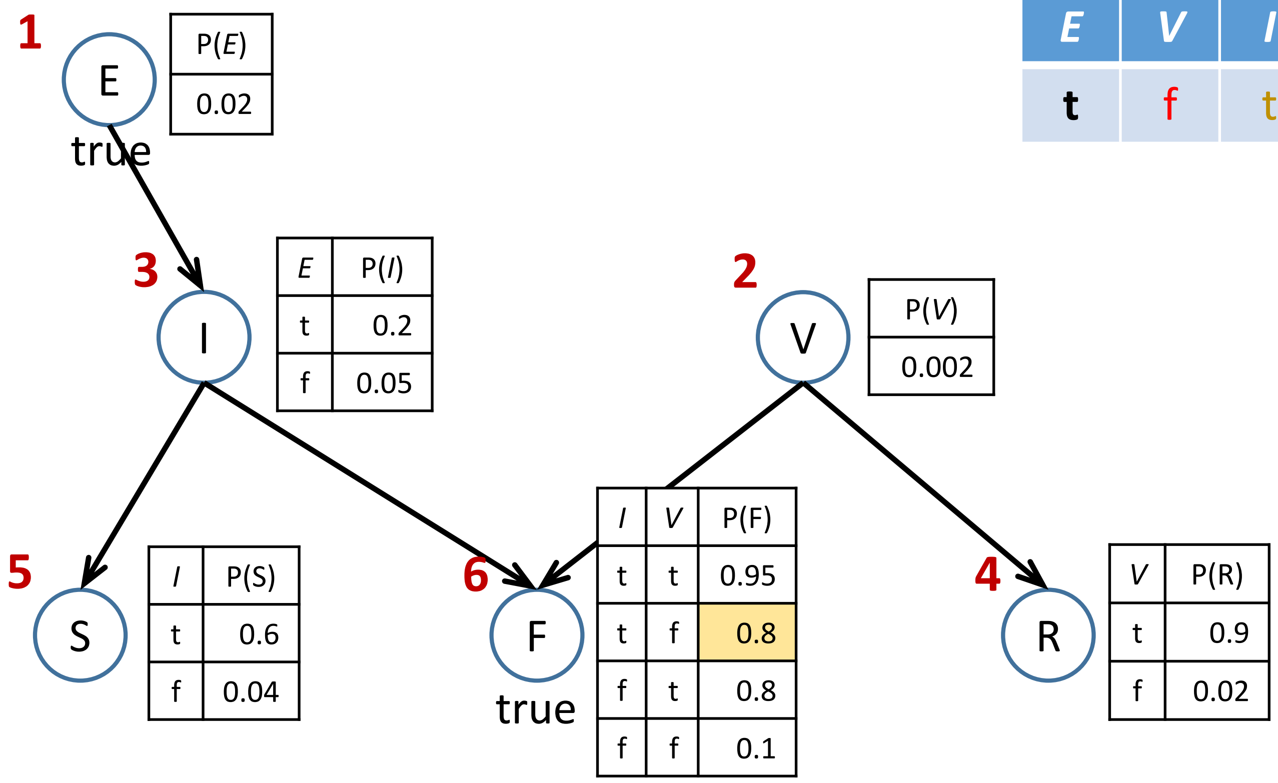
| *V* | P(R) |
|---|---|
| t | 0.9 |
| f | 0.02 |

w = 0.02

P(*Influenza* | *influenzaEpidemy* ∩ *fever*)

# Формирование взвешенной выборки

| *E* | *V* | *I* | *R* | *S* | *F* | *w* |
|---|---|---|---|---|---|---|
| t | f | t | f | t | t | .016 |




| P(*E*) |
|---|
| 0.02 |

| *E* | P(*I*) |
|---|---|
| t | 0.2 |
| f | 0.05 |

| P(*V*) |
|---|
| 0.002 |

| *I* | P(S) |
|---|---|
| t | 0.6 |
| f | 0.04 |

| *I* | *V* | P(F) |
|---|---|---|
| t | t | 0.95 |
| t | f | 0.8 |
| f | t | 0.8 |
| f | f | 0.1 |

| *V* | P(R) |
|---|---|
| t | 0.9 |
| f | 0.02 |

w = 0.02 * 0.8 = 0.016

P(*Influenza* | *influenzaEpidemy* ∩ *fever*)

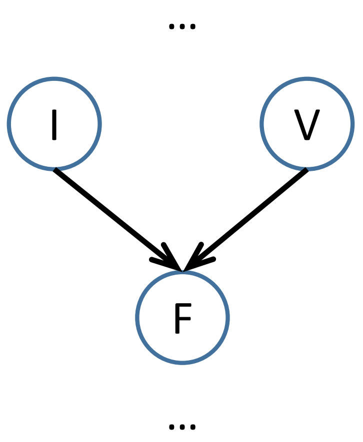
# Обучение байесовских сетей

1. Параметрическое обучение
    1. При полных данных
        1. На основе метода максимального правдоподобия
        2. Байесовские методы (распределение параметров)
    2. При неполных данных
2. Структурное обучение

# Обучение байесовских сетей. Максимальное правдоподобие

Набор данных по пациентам

| … | I | V | F | … |
|---|---|---|---|---|
| | 1 | 1 | 1 | |
| | 0 | 1 | 0 | |
| | 0 | 1 | 0 | |
| | 1 | 1 | 1 | |




$$\hat{P}(F = 1 \mid I = 1, V = 1) = \frac{\#(F=1, I=1, V=1)}{\#(I=1, V=1)}$$

$$\hat{P}(F = 1 \mid I = 1, V = 0) = \frac{\#(F=1, I=1, V=0)}{\#(I=1, V=0)}$$

$$\hat{P}(F = 1 \mid I = 0, V = 1) = \frac{\#(F=1, I=0, V=1)}{\#(I=0, V=1)}$$

$$\hat{P}(F = 1 \mid I = 0, V = 0) = \frac{\#(F=1, I=0, V=0)}{\#(I=0, V=0)}$$

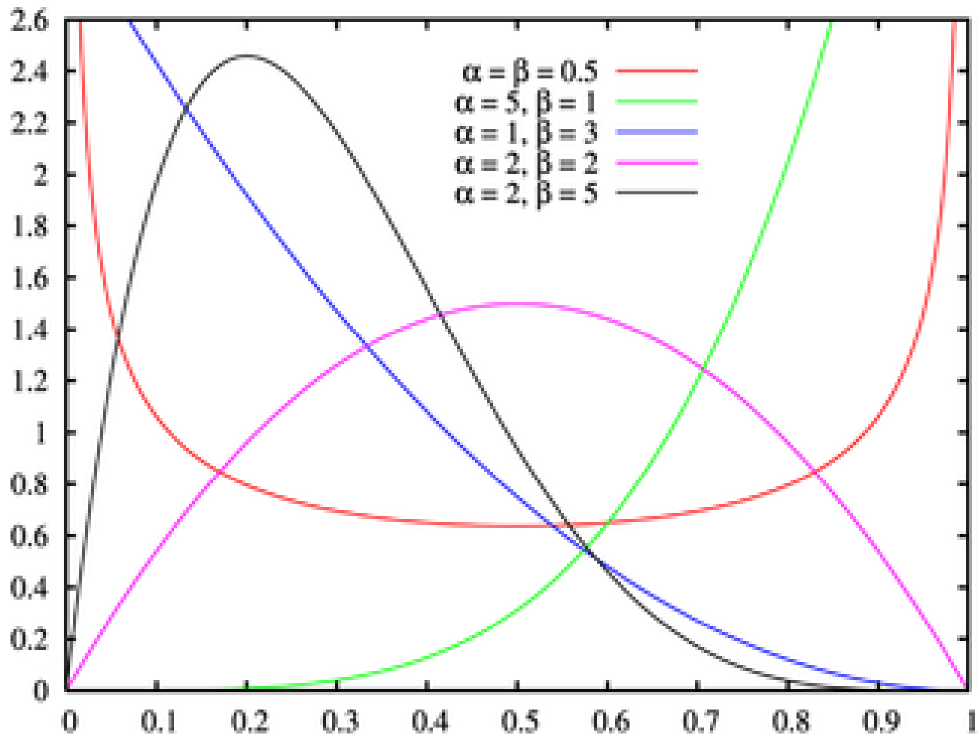
$$\hat{P}(F = 0 \mid I = i, V = v) = 1 - \frac{\#(F=1, I=i, V=v)}{\#(I=i, V=v)}, \;\; i, v \in \{0,1\}$$

# Обучение байесовских сетей. Байесовские оценки – общая идея

Распределение Бернулли – количество событий, имеющих вероятность *p* в *n* испытаниях.

Бета-распределение (сопряженное по отношению к распределению Бернулли) распределение вероятности одного события (*p*) если в схеме Бернулли оно случилось $\alpha$-1 раз и не случилось $\beta$-1 раз.

Псевдосчетчики – сочетание знаний эксперта и экспериментальных данных.



# Краткое резюме по байесовским сетям

- **Достоинства:**
  1. Надёжные теоретические основания (теория вероятностей)
  2. Можно комбинировать знания экспертов и наблюдения
- **Недостатки:**
  1. Может быть проблематично оценить вероятность (так, чтобы выполнялась вся аксиоматика) – для человека это не очень привычно

- **Практическое применение:**
  - Широко используются в системах диагностики (медицина, технические системы) и поддержки принятия решений

# Тема 8. Нечеткая логика

# Теория нечёткких множеств

**Теория нечётких множеств** – один из подходов к определению того, насколько хорошо некоторый объект подходит под расплывчатое описание.

**Нечёткая логика** – метод формирования рассуждений с помощью логических выражений, описывающих принадлежность элементов к нечётким множествам.

Неопределенность *другого* плана.

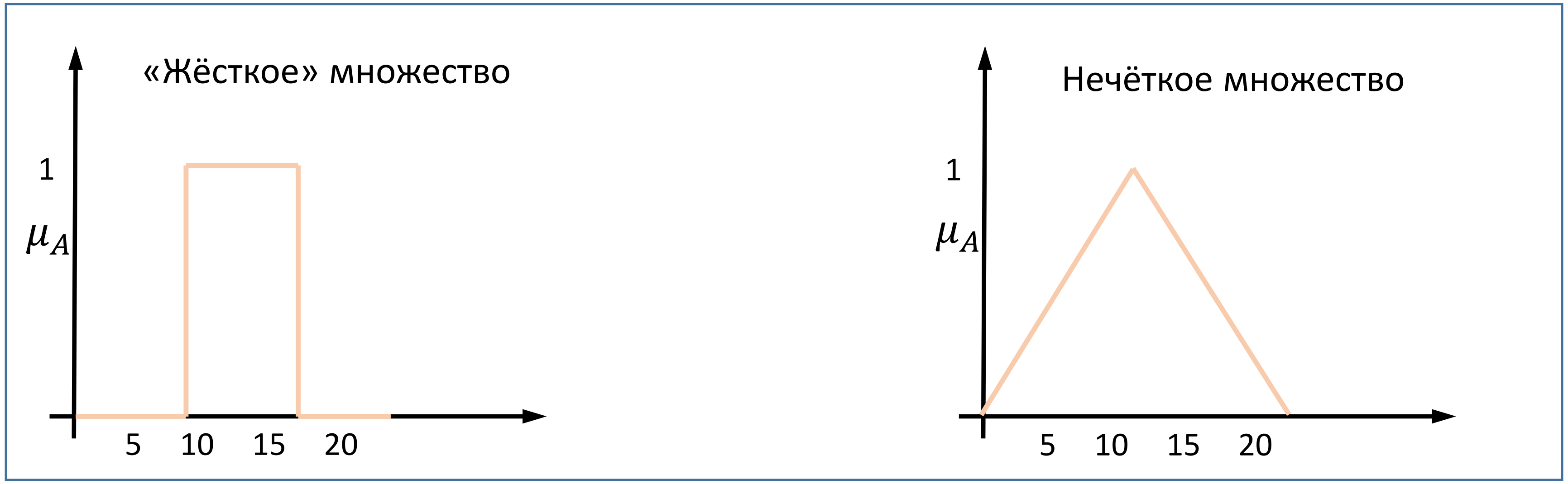
Лотфи Заде

# Нечёткие множества

**Нечёткое множество** *A* – совокупность упорядоченных пар, составленных из элементов *x* универсального множества *X* и соответствующих степеней принадлежности $\mu_A(x)$:

$$A = \{(x, \mu_A(x)) | x \in X\}$$

$\mu_A(x) \in [0,1]$ - функция принадлежности (характеристическая функция), показывающая, в какой степени элемент *x* принадлежит нечёткому множеству *A*.



# Пример нечёткого множества

Универсальное множество – рост человека.
Подмножества – высокие люди (Tall), среднего роста (Average), низкие (Short).

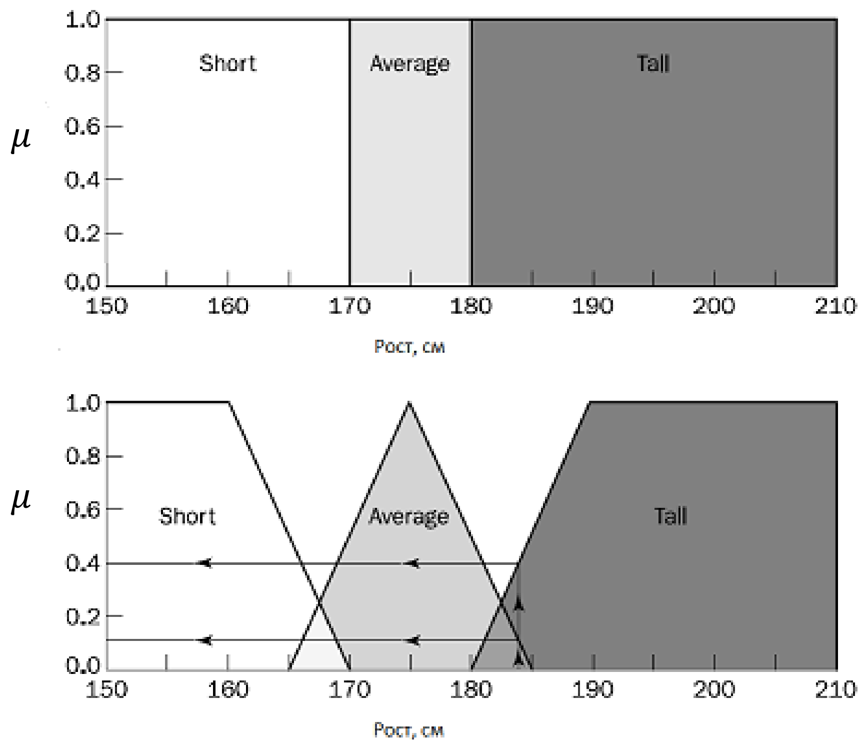



Иллюстрация из *Negnevitsky M. Artificial Intelligence. A Guide to Intelligent Systems. 2nd edition.*

# Представление нечётких множеств

Задание функции принадлежности:
- сигмоида, гауссиана… (усложняет вычисления)
- линеаризация: Tall = (0/180, 0.5/185, 1/190)
    - треугольные
    - трапециевидные

# Операции над нечёткими множествами

**Включение** (*A* содержится в *B*):

$$A \subset B \;\Leftrightarrow \mu_A(x) \leq \mu_B(x)$$

**Дополнение**:

$$\mu_{\neg A}(x) = 1 - \mu_A(x)$$

**Пересечение**:

$$\mu_{A \cap B}(x) = \min[\mu_A(x), \mu_B(x)], \text{ где } x \in X$$

**Объединение**:

$$\mu_{A \cap B}(x) = \max[\mu_A(x), \mu_B(x)], \text{ где } x \in X$$

# Нечёткие рассуждения (метод Мамдани)

ЕСЛИ x is A3
ИЛИ y is B1
ТО z is C1

ЕСЛИ x is A2
И y is B2
ТО z is C2

ЕСЛИ x is A1
ТО z is C3

ЕСЛИ project_funding is adequate
ИЛИ project_staffing is small
ТО risk is low

ЕСЛИ project_funding is marginal
ИЛИ project_staffing is large
ТО risk is normal

ЕСЛИ project_funding is small
ТО risk is high

И. Мамдани

# Фаззификация

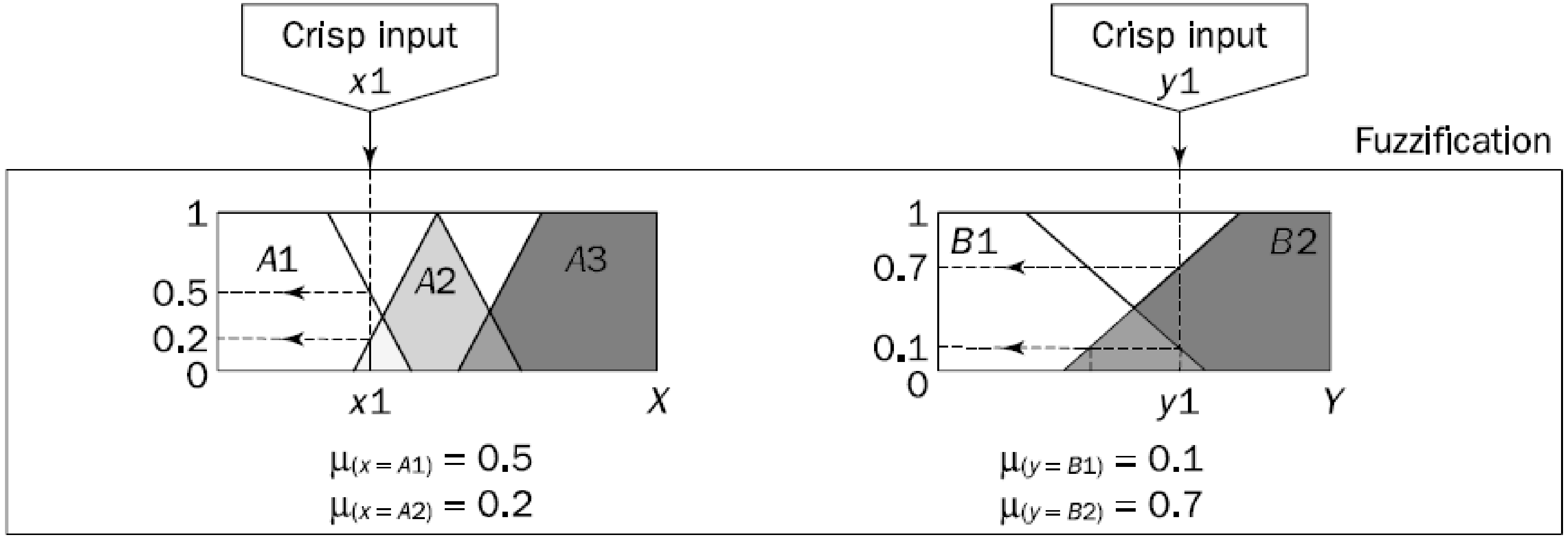



Иллюстрация из *Negnevitsky M. Artificial Intelligence. A Guide to Intelligent Systems. 2nd edition.*

# Вычисление правил

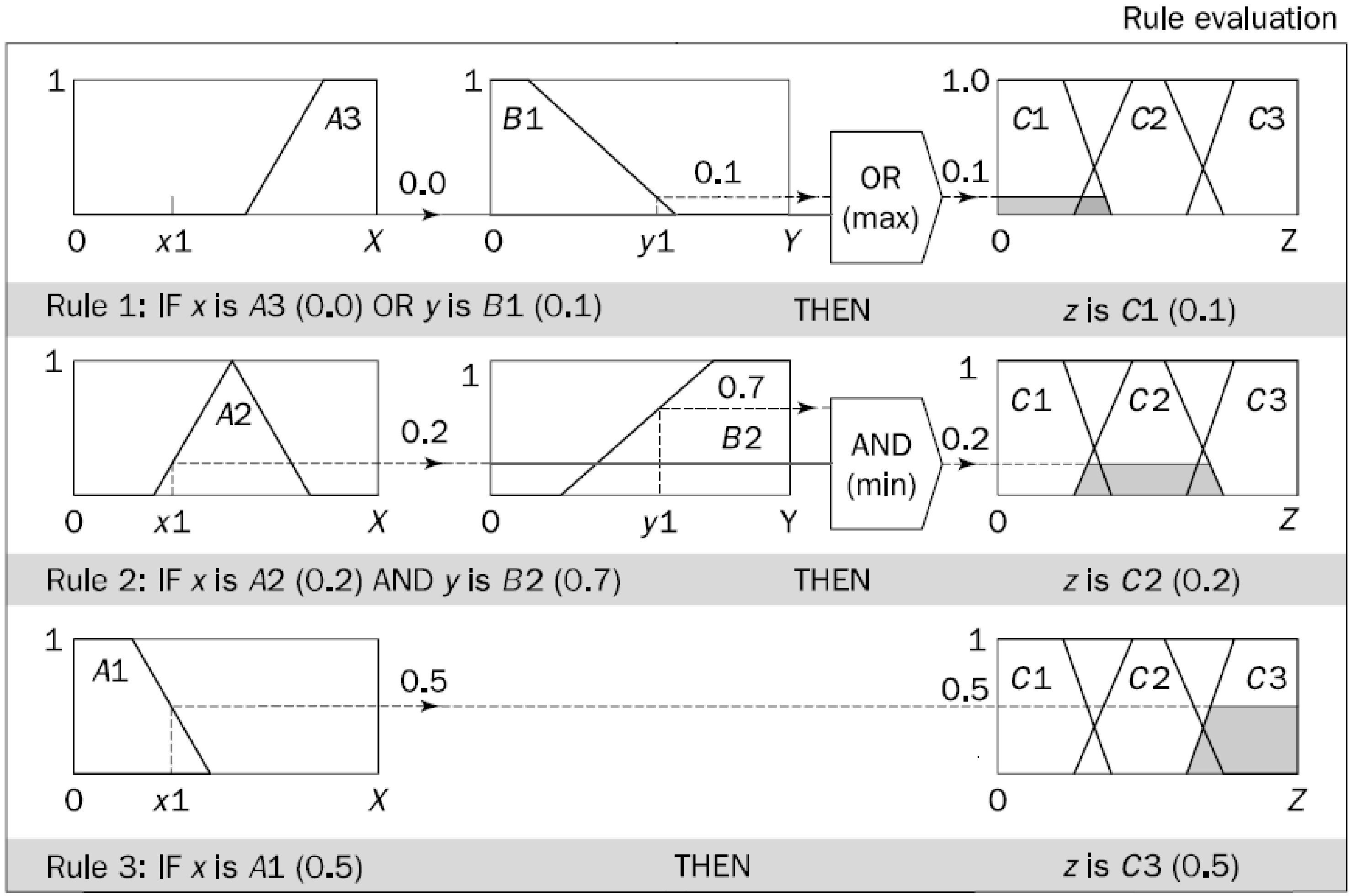



Иллюстрация из *Negnevitsky M. Artificial Intelligence. A Guide to Intelligent Systems. 2nd edition.*

# Агрегация заключений

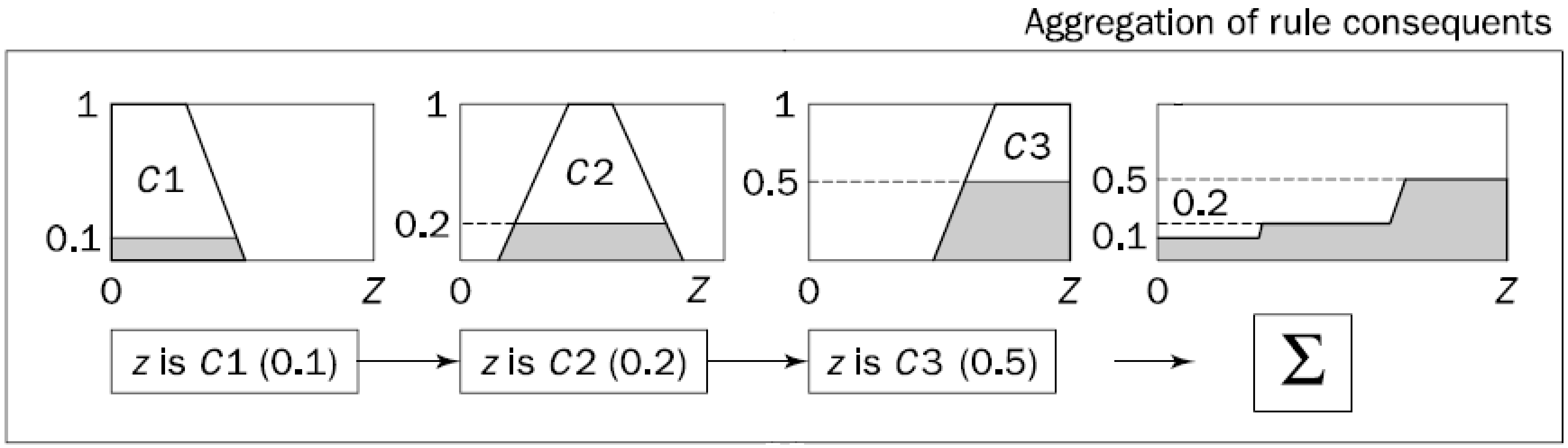



Иллюстрация из *Negnevitsky M. Artificial Intelligence. A Guide to Intelligent Systems. 2nd edition.*

# Дефаззификация

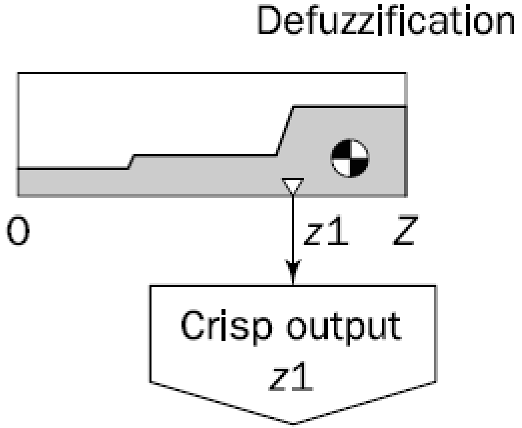



Иллюстрация из *Negnevitsky M. Artificial Intelligence. A Guide to Intelligent Systems. 2nd edition.*

# Резюме

- **Почти любому интеллектуальному агенту, действующему в реальном мире приходится иметь дело с неопределенностью**
- **Существуют разные способы формализации неопределенности (и разных видов неопределенности)**
- **Рассмотрели:**
  - Схему Шортлиффа-Бьюкенена (исторический интерес)
  - Байесовские сети (активно используются)
  - Нечёткую логику (активно используется, особенно в задачах автоматического управления)

# Литература

**Схема Шортлиффа:**
1) Джексон П. Введение в экспертные системы. 3-е издание (Гл. 9).
2) Negnevitsky M. Artificial Intelligence. A Guide to Intelligent Systems. 2nd edition (Ch. 3, 4).

**Байесовские сети:**
1) Рассел С., Норвиг П. Искусственный интеллект. Современный подход. 2-е издание (Гл. 13, 14).
2) Koller D., Friedman N. Probabilistic Graphical Models. Principles and Techniques. MIT Press, 2009. - 1270 p.

**Теория нечётких множеств:**
1) Джексон П. Введение в экспертные системы. 3-е издание (Гл. 9).
2) Negnevitsky M. Artificial Intelligence. A Guide to Intelligent Systems. 2nd edition (Ch. 4).