Decision Trees - Influence Diagrams

Descriptive Decision Theory

Descriptive Decision Theory tries to simulate human behavior in finding the right or best decision for a given problem

Example:

- Company can chose one of two places for a new store
- Option 1: 125.000 EUR profit per year
- Option 2: 150.000 EUR profit per year

Company should take Option 2, because it maximized the profit.

In real world not everything is known, so there are uncertainties in the model

Example:

- There are plans for restructure the local traffic, which changes the predicted profit
- Option 1: 125.000 EUR profit per year
- Option 2: 80.000 EUR profit per year

With modification Option 1 is the better one and without modification Option 2 is the better one

To model these variations in the environment we use so called Decision Tables

	z_1 (no modification)	z_2 (restructure)
a_1 (Option 1)	$125.000 = e_{11}$	$125.000 = e_{12}$
a_2 (Option 2)	$150.000 = e_{21}$	$80.000 = e_{22}$

An alternative a_1 dominates a_2
iff the value of a_1 is always greater of (or equal to) the value of a_2
That means, for all j: $e_{1j} \geq e_{2j}$

Example:

	z_1	z_2
a_1	150.000 = e_{11}	90.000 = e_{12}
a_2	125.000 = e_{21}	80.000 = e_{22}

Alternative a_2 could be dropped

Domination - Example

Some more alternatives:

	z_1	z_2	z_3	z_4	z_5	
a_1	0	20	10	60	25	dominated by a_3
a_2	-20	80	10	10	60	
a_3	20	60	20	60	50	
a_4	55	40	60	10	40	
a_5	50	10	30	5	20	dominated by a_4

- a_3 dominated a_1
- a_4 dominated a_5

  Alternatives a_1 and a_5 could be dropped

Decision Making Rules

	z_1	z_2	z_3	z_4
a_1	60	30	50	60
a_2	10	10	10	140
a_3	-30	100	120	130

Various Decision Rules are available

- Maximin - Rule
- Maximax - Rule
- Hurwicz - Rule
- Laplace - Rule

Maximin - Rule

	z_1	z_2	z_3	z_4	Minimum
a_1	60	30	50	60	30
a_2	10	10	10	140	10
a_3	-30	100	120	130	-30

Choose the one with the highest minimum

Contra To pessimistic, only focus on one column

Example

	z_1	z_2	z_3	z_4	Minimum
a_1	1,000,000	1,000,000	0.99	1,000,000	0.99
a_2	1	1	1	1	1

Maximax - Rule

	z_1	z_2	z_3	z_4	Maximum
a_1	60	30	50	60	60
a_2	10	10	10	140	140
a_3	-30	100	120	130	130

Choose the one with the highest maximum

Contra To optimistic, only focus on one column

Example

	z_1	z_2	z_3	z_4	Maximum
a_1	1,000,000	1,000,000	1,000,000	1,000,000	1,000,000
a_2	1,000,001	1	1	1	1,000,001

	z_1	z_2	z_3	z_4	Max	Min	$\Phi(a_i)$
a_1	60	30	50	60	60	30	$0.4 \cdot 60 + 0.6 \cdot 30 = 42$
a_2	10	10	10	140	140	10	$0.4 \cdot 140 + 0.6 \cdot 10 = 62$
a_3	-30	100	120	130	130	-30	$0.4 \cdot 130 + 0.6 \cdot (-30) = 34$

Combination of Maximin and Maximax -Rule

$\Phi(a) = \lambda \cdot \max(e_i) + (1 - \lambda) \cdot \min(e_i)$

λ represents readiness to assume risk

Contra Only focus on two column

Example ($\min(a_1) < \min(a_2), \max(a_1) < \max(a_2) \Rightarrow$ chose a_2)

	z_1	z_2	z_3	z_4	Max	Min
a_1	1,000,000	1,000,000	1,000,000	0.99	1,000,000	0.99
a_2	1,000,001	1	1	1	1,000,001	1

Laplace - Rule

	z_1	z_2	z_3	z_4	Mean
a_1	60	30	50	60	50
a_2	10	10	10	140	42.5
a_3	-30	100	120	130	80

Choose the one with the highest mean value

Contra

- Not every condition has the same probability
- Duplication of one condition could change the result

Most people would also chose a_3 in this example

In many cases probabilities could be assigned to each option

Objective Probabilities based on mathematic or statistic background

Subjective Probabilities based on intuition or estimations

Example:

The management estimates the probability for the restructure to 30%

The decision can be chosen by analyzing the expected values

	z_1 (nomodification) $p_1 = 0.7$	z_2 (restructure) $p_2 = 0.3$	Expectation
a_1 (Option 1)	$125.000 = e_{11}$	$125.000 = e_{12}$	125.000
a_2 (Option 2)	$150.000 = e_{21}$	$80.000 = e_{22}$	129.000

Option 2 has the higher expectation and should be used

Stochastic Dominance

Example

	z_1	z_2	z_3	z_4
	$p_1 = 0.3$	$p_2 = 0.2$	$p_3 = 0.4$	$p_4 = 0.1$
A_1	20	40	10	50
A_2	60	30	50	20

Here the options A_1 and A_2 can be considered as random variables, we have e.g. $P(A_1(z_1)=20)=0.3$.

A random variable A stochastically dominates B if for any outcome*,* A gives at least as high a probability of receiving at least *z* as does B, and for some *z*, A gives a higher probability of receiving at least *z*.

Formally A dominates B iff $p(A \geqslant x)$ for all x is always higher oder equal than $p(B \geqslant x)$, and it is strictly higher for at least one x . This definition of dominance gives a partial ordering of sets of options.

If A dominates B then the expected output for A is higher.

Note that there are different concepts of stochastic dominance in literature, here the so called first-order stochastic dominance is use

Stochastic Dominance

Example

Order payout by value in a decreasing order: Reward z, in brackets: $p(A_i = z)$

$$A_1: \quad 50(0.1) \quad 40(0.2) \quad 20(0.3) \quad 10(0.4)$$

$$A_2: \quad 60(0.3) \quad 50(0.4) \quad 30(0.2) \quad 20(0.1)$$

Determine probabilities $p(A_i \geqslant z)$: Reward z, in brackets: $p(A_i \geqslant z)$

$$A_1: \quad 50(0.1) \quad 40(0.3) \quad 20(0.6) \quad 10(1)$$

$$A_2: \quad 60(0.3) \quad 50(0.7) \quad 30(0.9) \quad 20(1)$$

Compare probabilities $p(A \geqslant x) \geqslant p(B \geqslant x)$ for all x

A_2 dominates A_1, because $p(A_2 \geqslant x) \geqslant p(A_1 \geqslant x)$ is valid for all real numbers x and (e.g.) $p(A_2 \geqslant 50) > p(A_1 \geqslant 50)$ holds

	Sales e_1	Profit e_2	Environment Pollution e_3
a_1	800	7000	-4
a_2	600	7000	-2
a_3	400	6000	0
a_4	200	4000	0

Efficient Alternatives

- Only focus on alternatives which are not dominated by others
- Example: Drop a_4

Finding a Decision

- If multiple alternatives are effective we need an algorithm to choose the preferred one

Goal find a function $U(e_1, e_2, \ldots, e_n)$ as a combination of all targets/criteria , which could be optimized

Linear combination

- Simplest variant: Linear combination of all targets using **weights** for criteria
- $U(e_1, e_2, \ldots, e_i) = \sum_{i=1}^{n} \omega_i \cdot e_i$

Example

- $\omega_1 = 10, \quad \omega_2 = 1, \quad \omega_3 = 500$

	Sales e_1	Profit e_2	Environment Pollution e_3	$U(e_1, e_2, e_3)$	
a_1	800	7000	-4	13000	⇐ Optimum
a_2	600	7000	-2	12000	
a_3	400	6000	0	10000	

Preference Orderings

Note that transitivity is not always given in decision making

Consider the following set of dice (so called Efron-Dice)

- Dice A has sides: 2, 2, 4, 4, 9, 9
- Dice B has sides: 1, 1, 6, 6, 8, 8
- Dice C has sides: 3, 3, 5, 5, 7, 7

The probability that A rolls a higher number than B, the probability that B rolls higher than C, and the probability that C rolls higher than A are all $5/9$, so this set of dice lead to nontransitive decisions. In fact, it has the property that, for each dice in the set, there is another dice that rolls a higher number than in more than half the time.

Standard economic theory assumes that preferences are transitive.

In most real applications there are good arguments for imposing such “rationality requirements”, e.g. the **money pump** argument, or the **Dutch Book** argument.

Preference Orderings

A *preference ordering* $\preceq$ is a ranking of all possible states of affairs (worlds) S

- these could be outcomes of actions, truth assignments, states in a search problem, etc.

- $s \preceq t$: means that state ***t*** is *at least as good as* ***s***

- $s \succ t$: means that state s *is strictly preferred to* t

We insist that $\preceq$ is

- reflexive: i.e., $s \preceq s$ for all states s

- transitive: i.e., if $s \preceq t$ and $t \preceq w$, then $s \preceq w$

- connected: for all states s,t, either $s \preceq t$ or $t \preceq s$

Utilities

Rather than just ranking outcomes, we are often able to quantify our degree of preference

A *utility function* $U : S \to \mathbb{R}$ associates a realvalued *utility* with each outcome.

- $U(s)$ measures the *degree* of preference for s

Note: U induces a preference ordering $\leq_U$ over S defined as: s $\leq_U$ t iff $U(s) \leq U(t)$

- $\leq_U$ is reflexive, transitive, and connected

Expected Utility

Under conditions of uncertainty, each decision d induces a distribution P_d over possible outcomes

- $P_d(s)$ is probability of outcome s under decision d

The *expected utility* of decision d is defined by

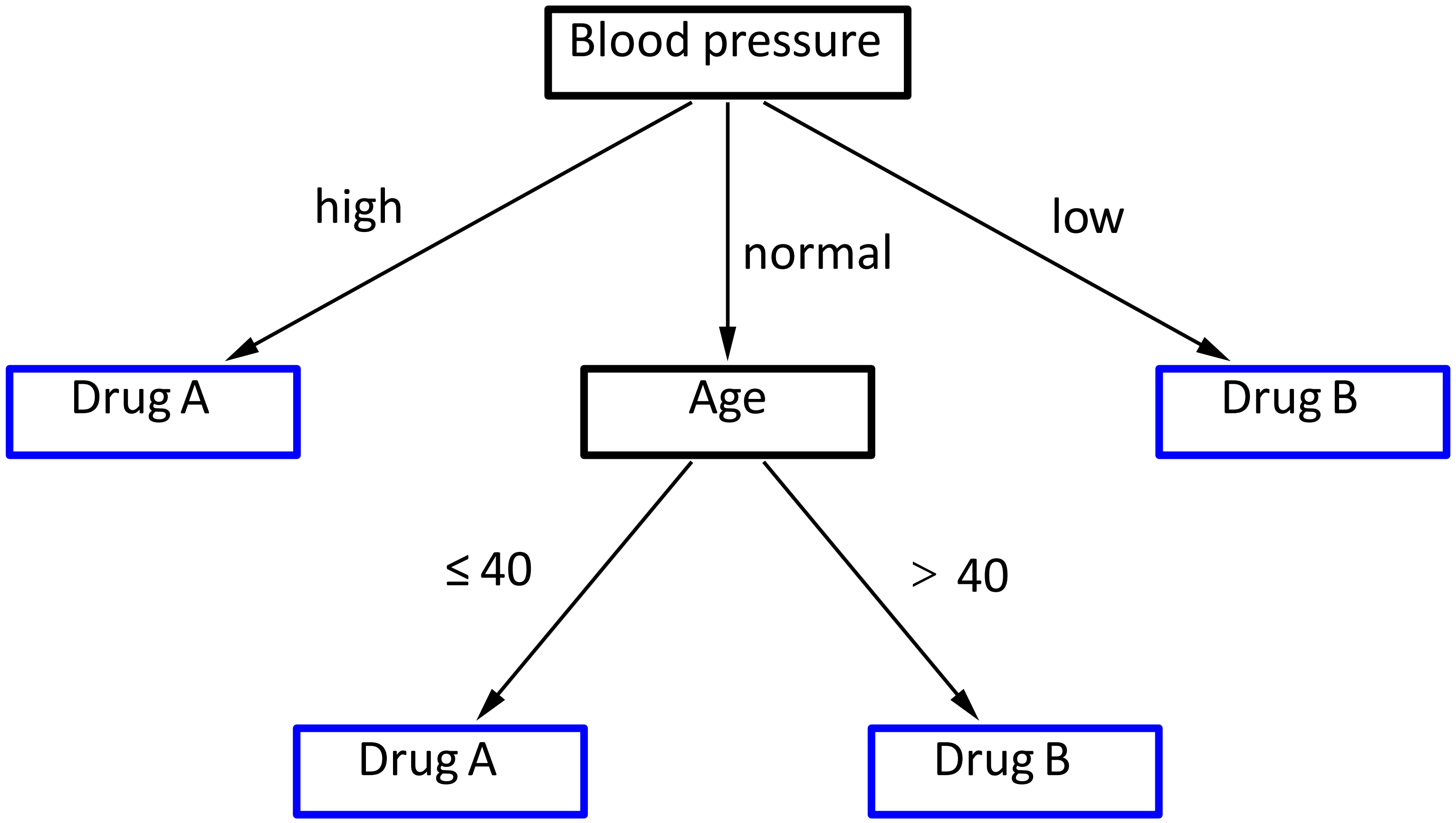
$$EU(d) = \sum_{s \in S} P_{d}(s)\, U(s)$$

The *principle of maximum expected utility (MEU)* states that the optimal decision under conditions of uncertainty is that with the greatest expected utility.

Decision Trees

in Machine Learning

Assignment of a drug to a patient:



Recursive Descent:

Start at the root node.

If the current node is an leaf node:

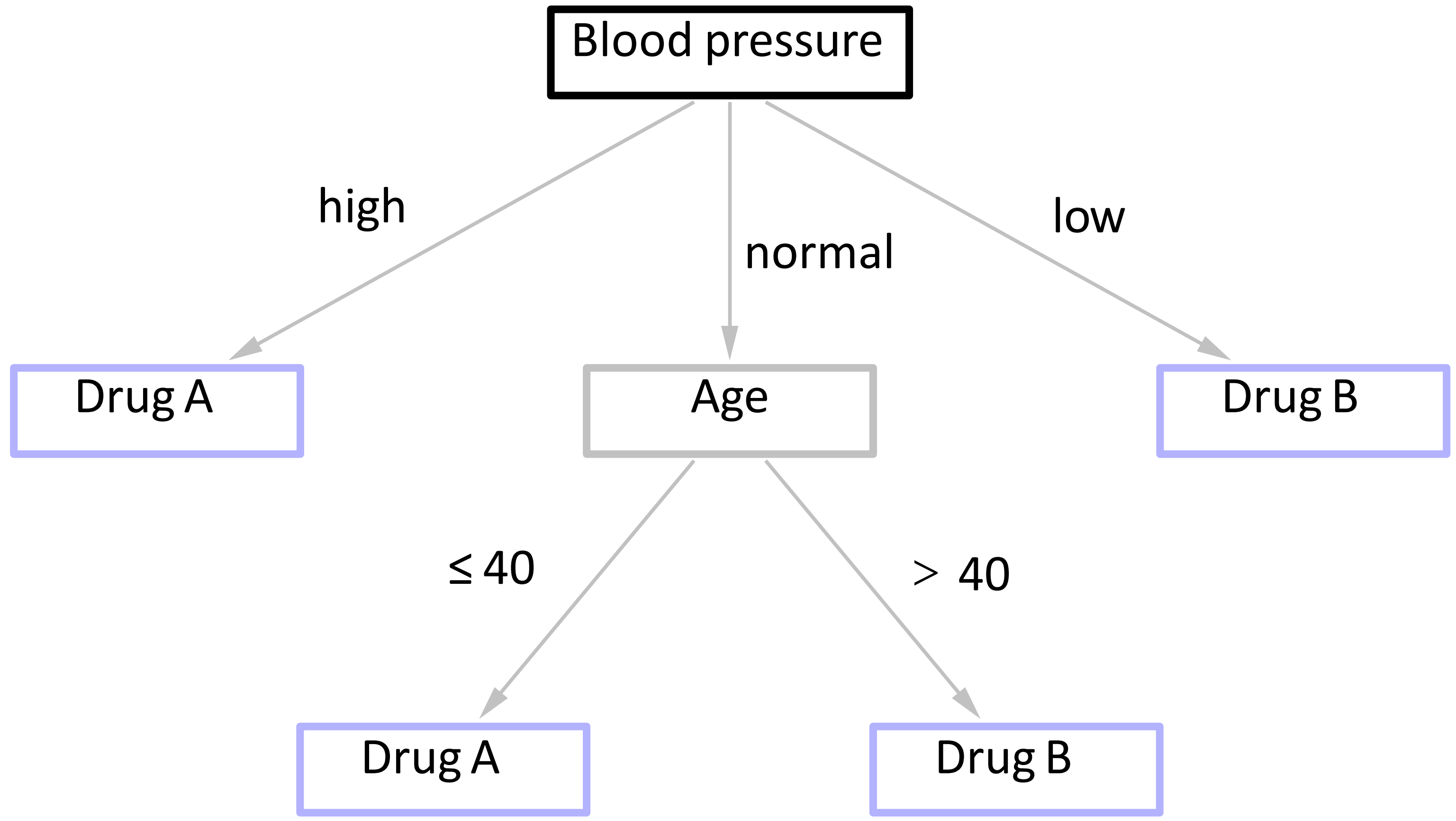
- Return the class assigned to the node.

If the current node is an inner node:

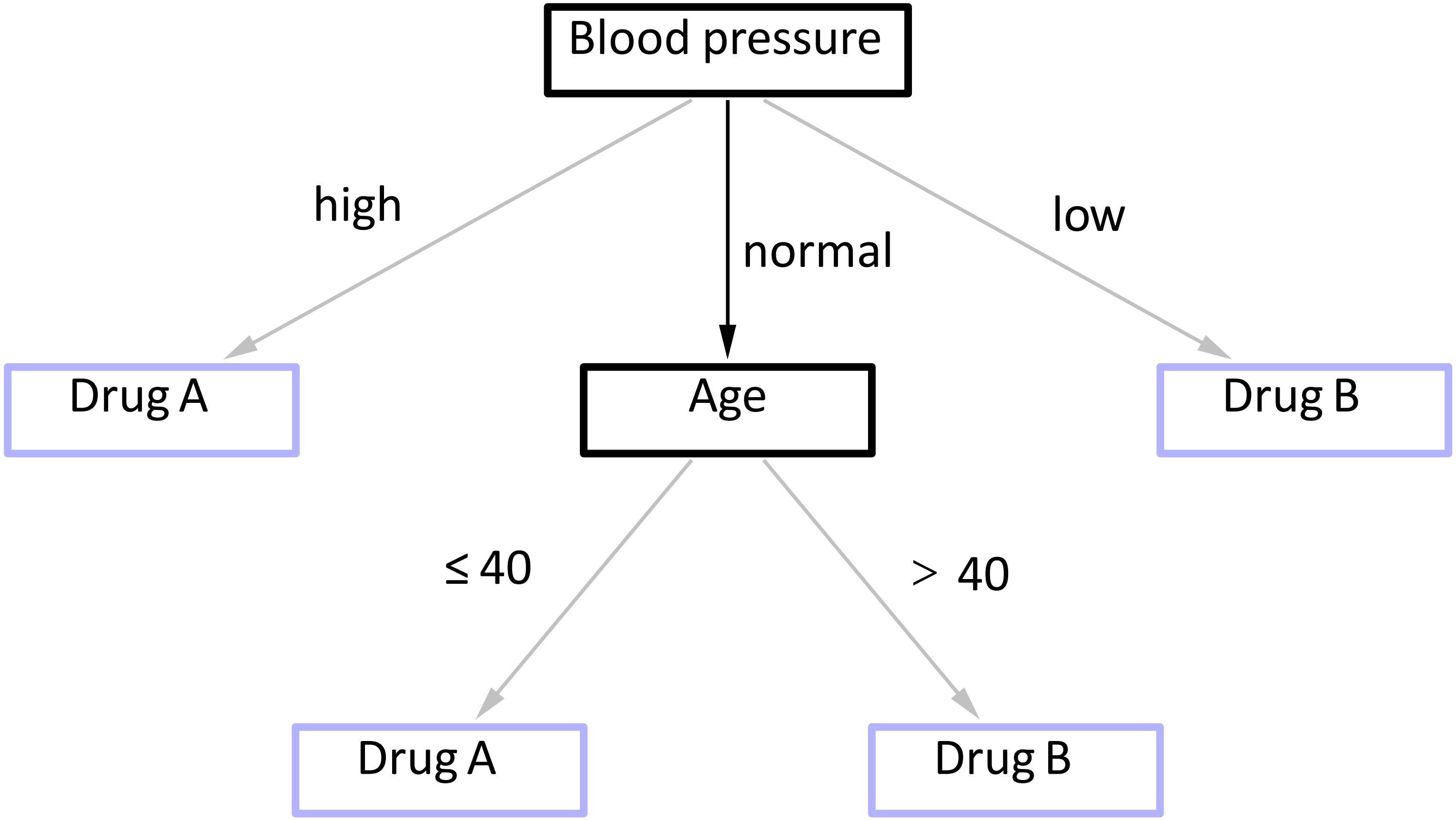
- Test the attribute associated with the node.
- Follow the branch labeled with the outcome of the test.
- Apply the algorithm recursively.

Intuitively: Follow the path corresponding to the case to bedecided.

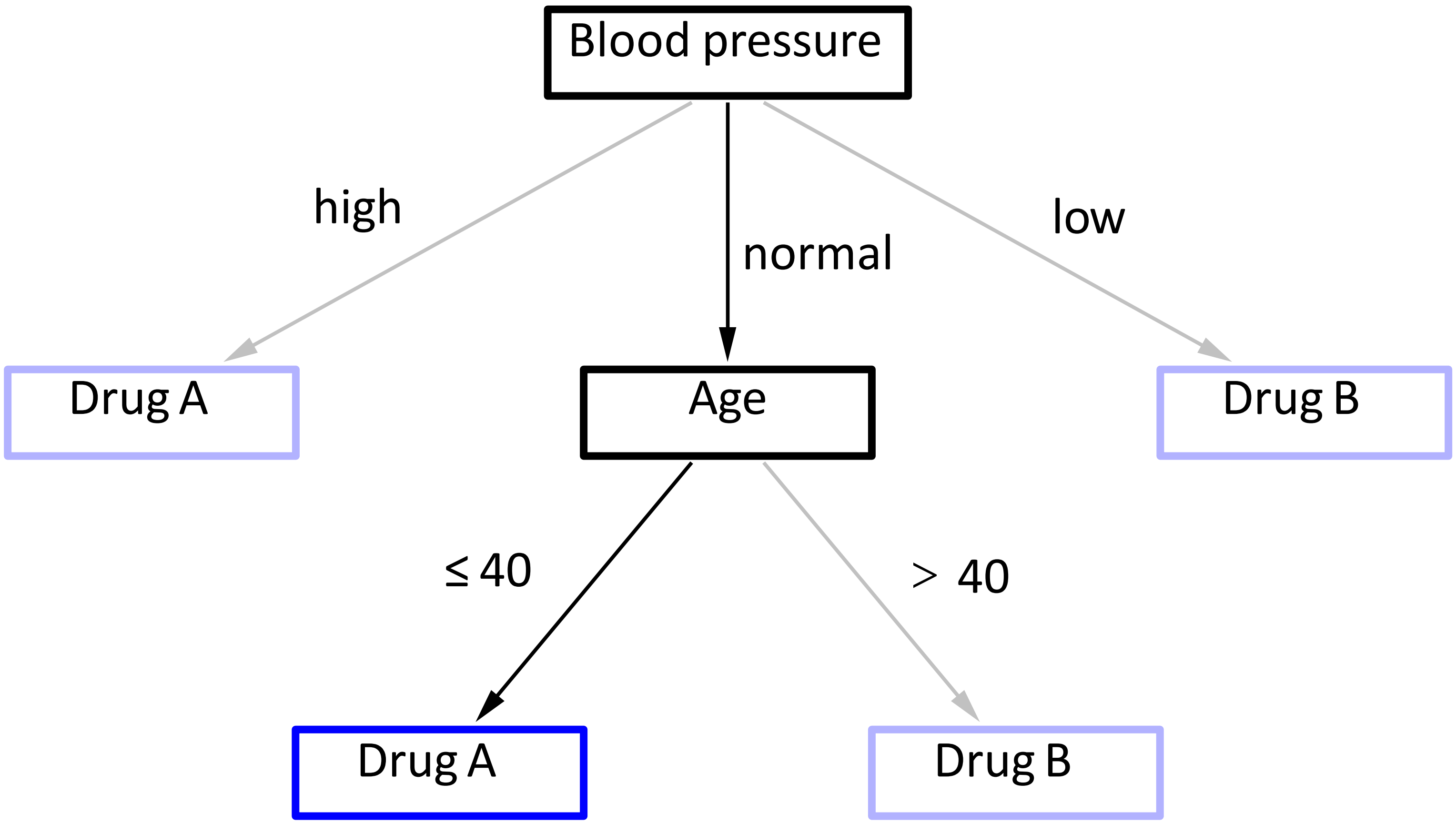
Example: Assignment of a drug to a patient



Assignment of a drug to a patient:



Assignment of a drug to a patient:



Learning of Decision Trees

Top- down approach

- Build the decision tree from top to bottom (from the root to the leaves).

Greedy Selection of a Test Attribute

- Compute an evaluation measure for all attributes.
- Select the attribute with the best evaluation.

Divide and Conquer / Recursive Descent

- Divide the example cases according to the values of the test attribute.
- Apply the procedure recursively to the subsets.
- Terminate the recursion if
  – all cases belong to the same class
  – no more test attributes are available

Induction of a Decision Tree: Example

Patient database

12 example cases

3 descriptive attributes

1 class attribute

No	Sex	Age	Blood pr.	Drug
1	male	20	normal	A
2	female	73	normal	B
3	female	37	high	A
4	male	33	low	B
5	female	48	high	A
6	male	29	normal	A
7	female	52	normal	B
8	male	42	low	B
9	male	61	normal	B
10	female	30	normal	A
11	female	26	low	B
12	male	54	high	A

Assignment of drug

(without patient attributes)

always drug A or always drug B:

50% correct (in 6 of 12 cases)

Induction of a Decision Tree: Example

Sex of the patient

Division w.r.t. male/female.

Assignment of drug

male:	50% correct	(in 3 of 6 cases)
female:	50% correct	(in 3 of 6 cases)
total:	50% correct	(in 6 of 12 cases)

No	Sex	Drug
1	male	A
6	male	A
12	male	A
4	male	B
8	male	B
9	male	B
3	female	A
5	female	A
10	female	A
2	female	B
7	female	B
11	female	B

Induction of a Decision Tree: Example

Age of the patient

Sort according to age.

Find best age split.
here: ca. 40 years

Assignment of drug

≤ 40:	A.	67% correct	(in 4 of 6 cases)
> 40:	B.	67% correct	(in 4 of 6 cases)
total:		67% correct	(in 8 of 12 cases)

No	Age	Drug
1	20	A
11	26	B
6	29	A
10	30	A
4	33	B
3	37	A
8	42	B
5	48	A
7	52	B
12	54	A
9	61	B
2	73	B

Induction of a Decision Tree: Example

Blood pressure of the patient

Division w.r.t. high/normal/low.

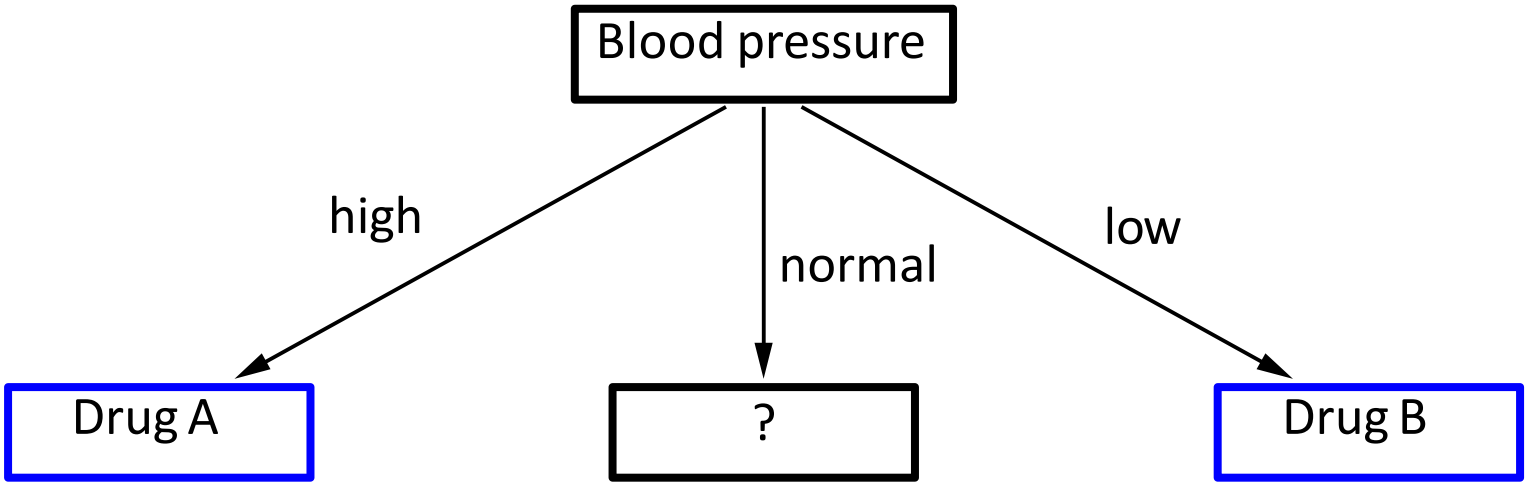
Assignment of drug

high:	A	100% correct	(in 3 of 3 cases)
normal:		50% correct	(in 3 of 6 cases)
low:	B	100% correct	(in 3 of 3 cases)
total:		75% correct	(in 9 of 12 cases)

No	Blood pr.	Drug
3	high	A
5	high	A
12	high	A
1	normal	A
6	normal	A
10	normal	A
2	normal	B
7	normal	B
9	normal	B
4	low	B
8	low	B
11	low	B

Induction of a Decision Tree: Example

Current Decision Tree:



Induction of a Decision Tree: Example

Blood pressure and sex

Only patients with normal blood pressure.

Division w.r.t. male/female.

Assignment of drug

male:	A	67% correct	(2 of 3)
female:	B	67% correct	(2 of 3)
total:		67% correct	(4 of 6)

No	Blood pr.	Sex	Drug
3	high		A
5	high		A
12	high		A
1	normal	male	A
6	normal	male	A
9	normal	male	B
2	normal	female	B
7	normal	female	B
10	normal	female	A
4	low		B
8	low		B
11	low		B

Induction of a Decision Tree: Example

Blood pressure and age

Only patients with normal blood pressure.

Sort according to age.

Find best age split.
here: ca. 40 years

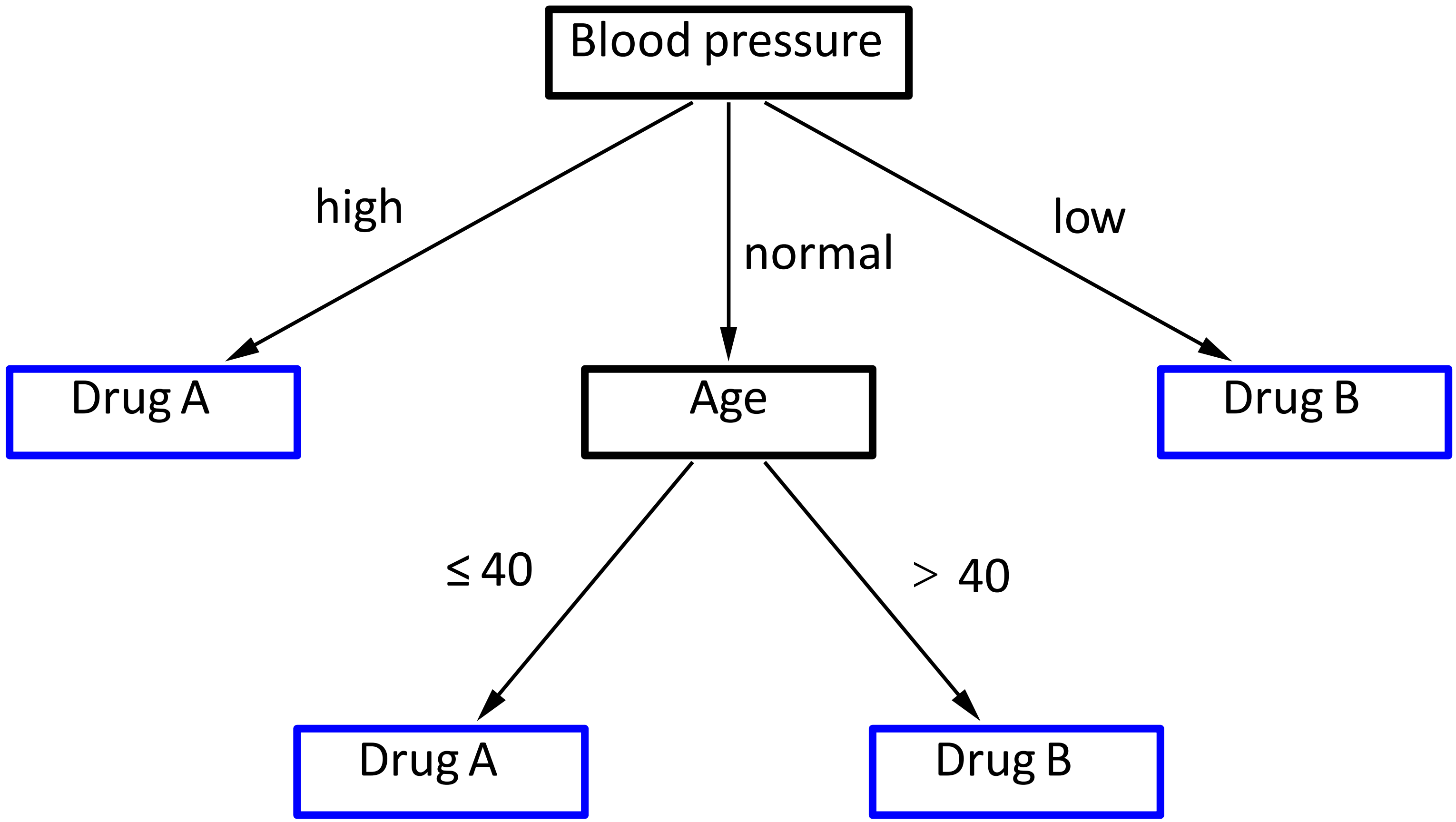
Assignment of drug

≤ 40:	A.	100% correct	(3 of 3)
> 40:	B.	100% correct	(3 of 3)
total:		100% correct	(6 of 6)

No	Blood pr.	Age	Drug
3	high		A
5	high		A
12	high		A
1	normal	20	A
6	normal	29	A
10	normal	30	A
7	normal	52	B
9	normal	61	B
2	normal	73	B
11	low		B
4	low		B
8	low		B

Result of Decision Tree Induction

Assignment of a drug to a patient:



Evaluation Measures

Evaluation measure used in the above example:
Rate of correctly classified example cases.

- Advantage: simple to compute, easy tounderstand.
- Disadvantage: works well only for twoclasses.

If there are more than two classes, the rate of misclassified example cases **neglects a lot of the available information**.

- Only the majority class—that is, the class occurring most often in (a subset of) the example cases—is really considered.
- The distribution of the other classes has no influence. However, a good choice here can be important for deeper levels of the decision tree.

Therefore : Several other evaluation measures are studied, e.g.

Information gain and its various normalizations.

An Information-theoretic Evaluation Measure

Information Gain (Kullback and Leibler 1951, Quinlan 1986)

Based on Shannon Entropy $H = -\sum_{i=1}^{n} p_i \log_2 p_i$ (Shannon 1948)

$$I_{\text{gain}}(C, A) = \overbrace{H(C)} - \overbrace{H(C|A)}$$

$$= -\sum_{i=1}^{n_C} p_{i.} \log_2 p_{i.} - \sum_{j=1}^{n_A} p_{.j} \left(-\sum_{i=1}^{n_C} p_{i|j} \log_2 p_{i|j} \right)$$

$H(C)$	Entropy of the class distribution (C: class attribute)
$H(C\|A)$	*Expected entropy* of the class distribution if the value of the attribute A becomes known
$H(C) - H(C\|A)$	Expected entropy reduction or *information gain*

Inducing the Decision Tree by Information Gain

Information gain for drug and sex:

$$H(\text{Drug}) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$$

$$H(\text{Drug} \mid \text{Sex}) = \frac{1}{2}\underbrace{\left(-\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}\right)}_{H(\text{Drug}|\text{Sex=male})} + \frac{1}{2}\underbrace{\left(-\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}\right)}_{H(\text{Drug}|\text{Sex=female})} = 1$$

$$I_{\text{gain}}(\text{Drug}, \text{Sex}) = 1 - 1 = 0$$

No gain at all since the initial the uniform distribution of drug is splitted into two (still) uniform distributions.

Inducing the Decision Tree by Information Gain

Information gain for drug and age:

$$H(\text{Drug}) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$$

$$H(\text{Drug} \mid \text{Age}) = \frac{1}{2}\underbrace{\left(-\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}\right)}_{H(\text{Drug}|\text{Age}\leq 40)} + \frac{1}{2}\underbrace{\left(-\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}\right)}_{H(\text{Drug}|\text{Age}>40)} \approx 0.9183$$

$$I_{\text{gain}}(\text{Drug}, \text{Age}) = 1 - 0.9183 = 0.0817$$

Splitting w. r. t. age can reduce the overall entropy.

Inducing the Decision Tree by Information Gain

Information gain for drug and blood pressure:

$$H(\text{Drug}) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$$

$$H(\text{Drug} \mid \text{Blood\_pr}) = \frac{1}{4}\cdot 0 + \frac{1}{2}\Bigg(\underbrace{-\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}}_{H(\text{Drug}|\text{Blood\_pr=normal})}\Bigg) + \frac{1}{4}\cdot 0 = 0.5$$

$$I_{\text{gain}}(\text{Drug}, \text{Blood\_pr}) = 1 - 0.5 = 0.5$$

Largest information gain, so we first split w. r. t. blood pressure (as in the example with misclassification rate).

Inducing the Decision Tree by Information Gain

Next level: Subtree blood pressure is normal.

Information gain for drug and sex:

$$H(\text{Drug}) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$$

$$H(\text{Drug} \mid \text{Sex}) = \frac{1}{2}\underbrace{\left(-\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3}\right)}_{H(\text{Drug}|\text{Sex=male})} + \frac{1}{2}\underbrace{\left(-\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}\right)}_{H(\text{Drug}|\text{Sex=female})} = 0.9183$$

$$I_{\text{gain}}(\text{Drug}, \text{Sex}) = 0.0817$$

Entropy can be decreased.

Inducing the Decision Tree by Information Gain

Next level: Subtree blood pressure is normal.

Information gain for drug and age:

$$H(\text{Drug}) = -\left(\frac{1}{2}\log_2\frac{1}{2} + \frac{1}{2}\log_2\frac{1}{2}\right) = 1$$

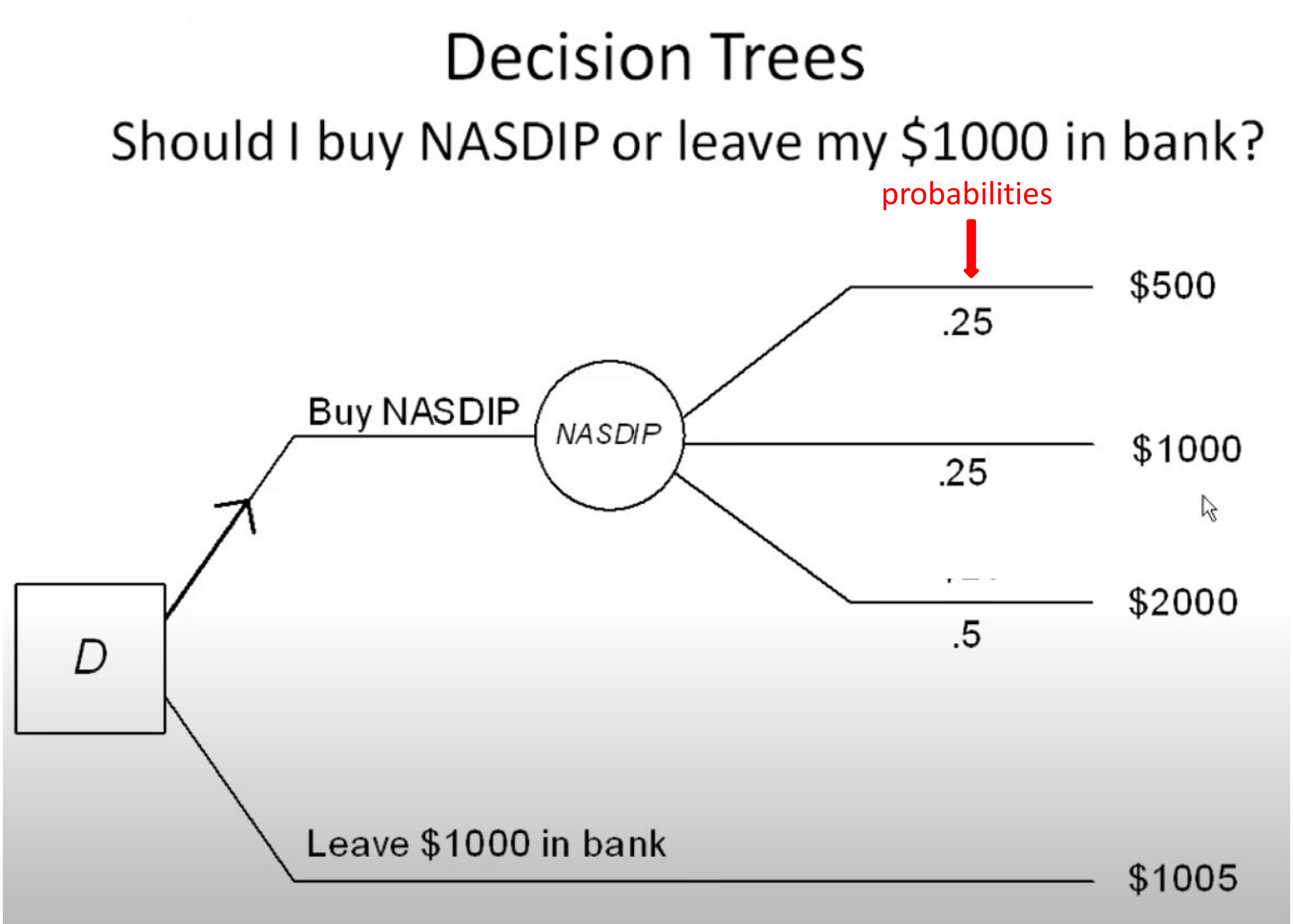
$$H(\text{Drug} \mid \text{Age}) = \frac{1}{2}\cdot 0 + \frac{1}{2}\cdot 0 = 0$$

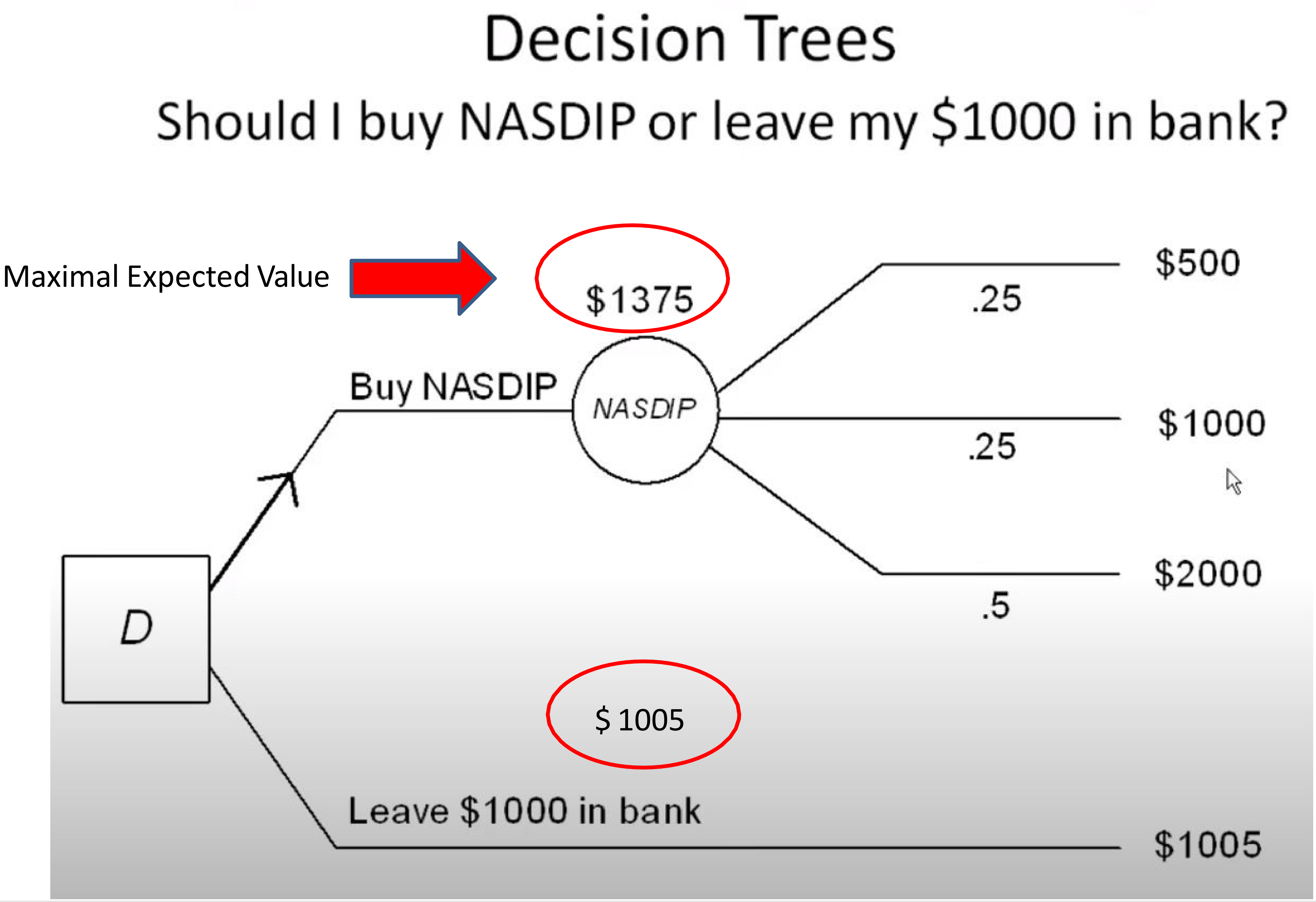
$$I_{\text{gain}}(\text{Drug}, \text{Age}) = 1$$

Maximal information gain, that is we result in a perfect classification (again, as in the case of using misclassification rate).

Decision Trees

in Economics


Decision Trees
Should I buy NASDIP or leave my $1000 in bank?
probabilities
$500
.25
Buy NASDIP
NASDIP
$1000
.25
$2000
.5
D
Leave $1000 in bank
$1005


Decision Trees
Should I buy NASDIP or leave my $1000 in bank?
Maximal Expected Value
$1375
Buy NASDIP
NASDIP
$500
.25
$1000
.25
$2000
.5
D
$ 1005
Leave $1000 in bank
$1005

Expected Utility

Where do utilities come from?

- underlying foundations of utility theory tightly couple utility with action/choice
- a utility function can be determined by asking someone about their preferences for actions in specific scenarios (or “lotteries” over outcomes)

Utility functions needn’t be unique

- if I multiply U by a positive constant, all decisions have same relative utility
- if I add a constant to U, same thing
- *U is unique up to positive affine transformation*

Decision Problems: Uncertainty

A *decision problem under uncertainty* is:

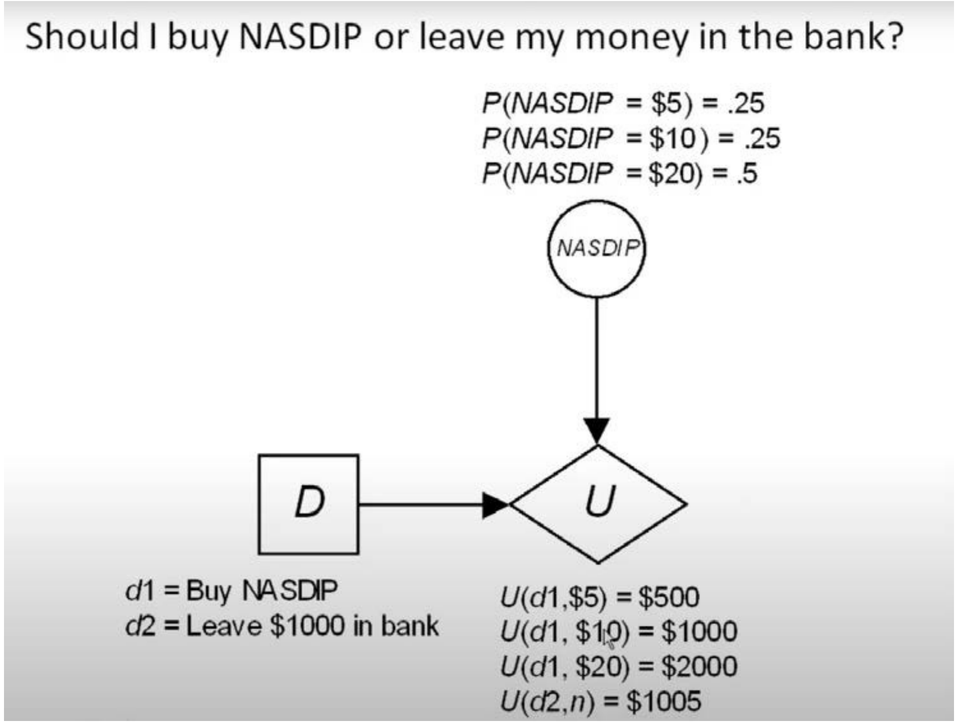
- a set of *decisions* D
- a set of *outcomes* or states S
- an *outcome function* $P : D \rightarrow \Delta(S)$
  $\Delta(S)$ is the set of distributions over S (e.g., P_d)
- ◦ a *utility function* U over S

A solution to a decision problem under uncertainty is any $d^* \in D$ such that $EU(d) \leq EU(d^*)$ for all $d \in D$

Influence Diagrams

Outcome space is large

- like all of our problems, states spaces can be huge
- don't want to spell out distributions like P_d explicitly
- Solution: Extend Bayes Networks with decision nodes and utility nodes
- Use **Influence Diagrams**



Methods for solving Influence Diagrams

Decision space is large

- usually our decisions are not one-shot actions
- rather they involve sequential choices (likeplans)
- if we treat each plan as a distinct decision, decision space is too large to handle directly

Solution: use dynamic programming methods to *construct* optimal plans (actually generalizations of plans, called policies . . . like in gametrees)

Simple Example: Influence Diagrams

Sam has a non-small-cell carcinoma of the lung.

The preferred treatment in this situation is a thoracotomy.

The alternative treatment is radiation.

We don't know if mediastinal metastasis is present.

If mediastinal metastasis is present, a thoracotomy would be contraindicated because it subjects the patient to a risk of death with no health benefit.

If mediastinal metastases are absent, a thoracotomy offers a substantial survival advantage as long as the primary tumor has not metastasized to distant organs.

We have two tests available for assessing the involvement of the mediastinum.

They are CT scan and mediastinoscopy.

Simple Example: Influence Diagrams

This problem involves three decisions:

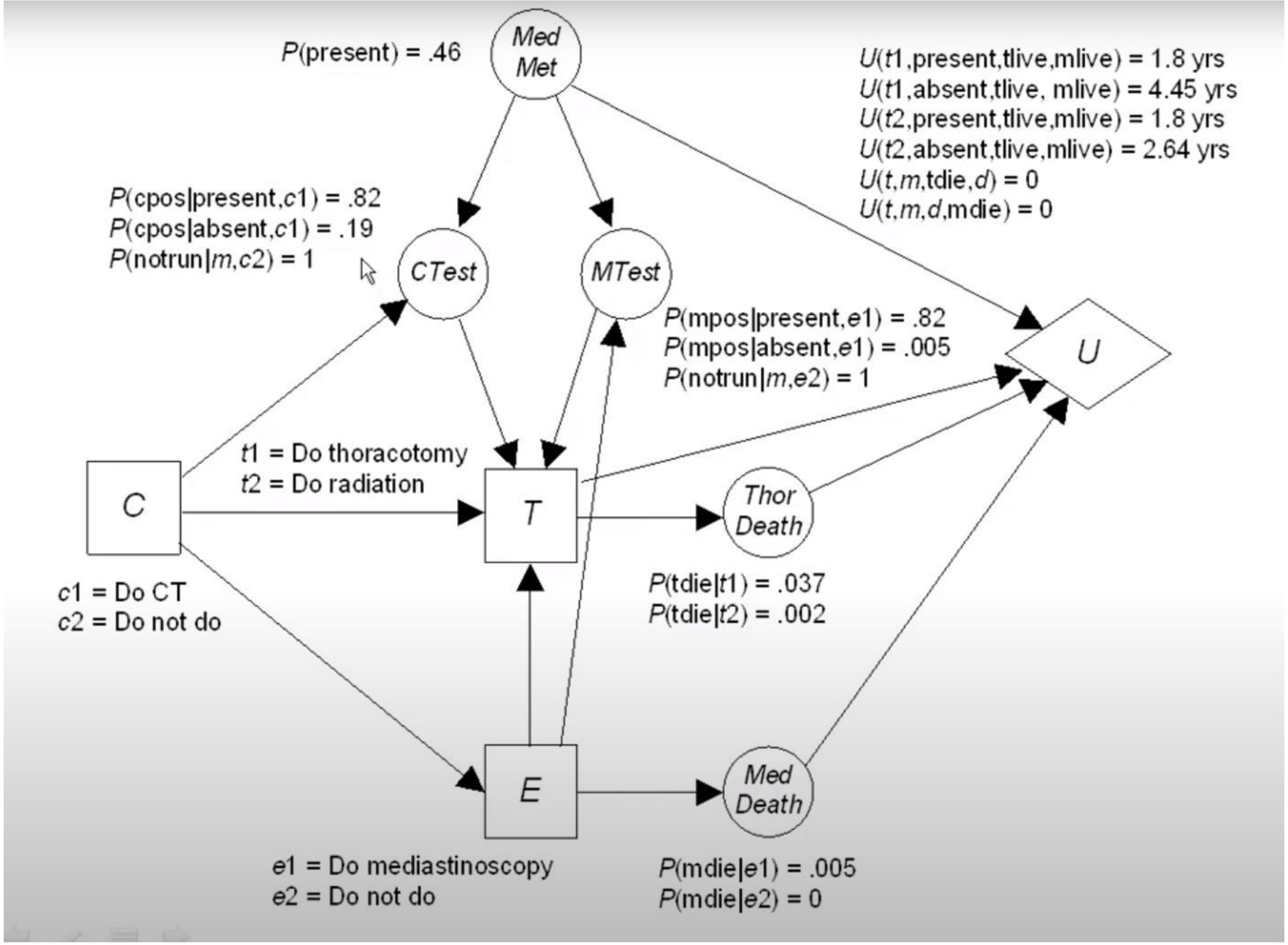
1. Should Sam undergo a CT scan?
2. Second, given this decision and any CT results, should he undergo mediastinoscopy?
3. Third, given these decisions and any test results, should the patient undergo a thoracotomy?

Influence Diagrams

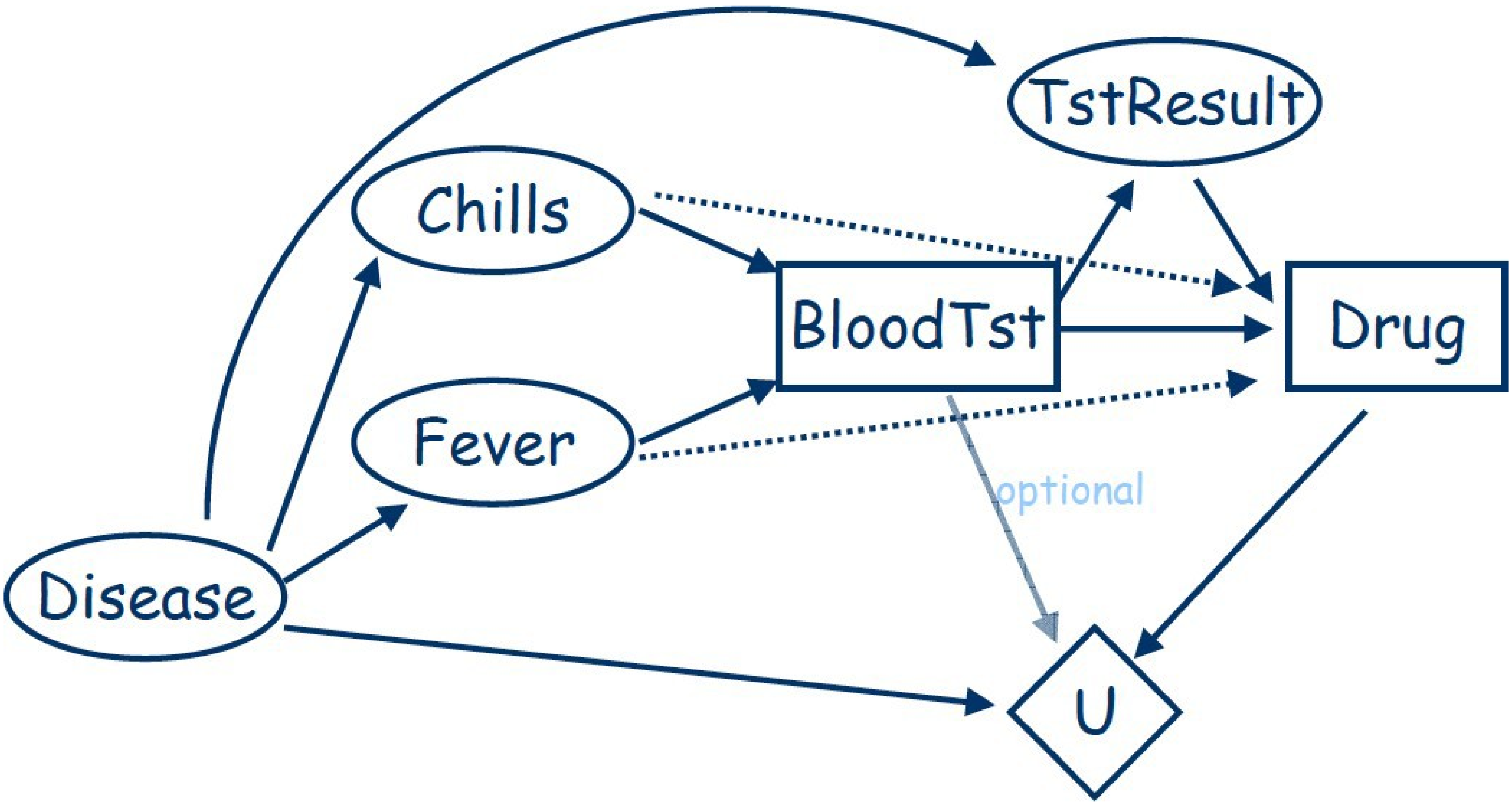
Influence diagrams provide a way of representing sequential decision problems

- basic idea: represent the variables in the problem as you would in a BN
- add decision variables – variables that you "control"
- add utility variables – how good different states are

Simple Example: Influence Diagrams

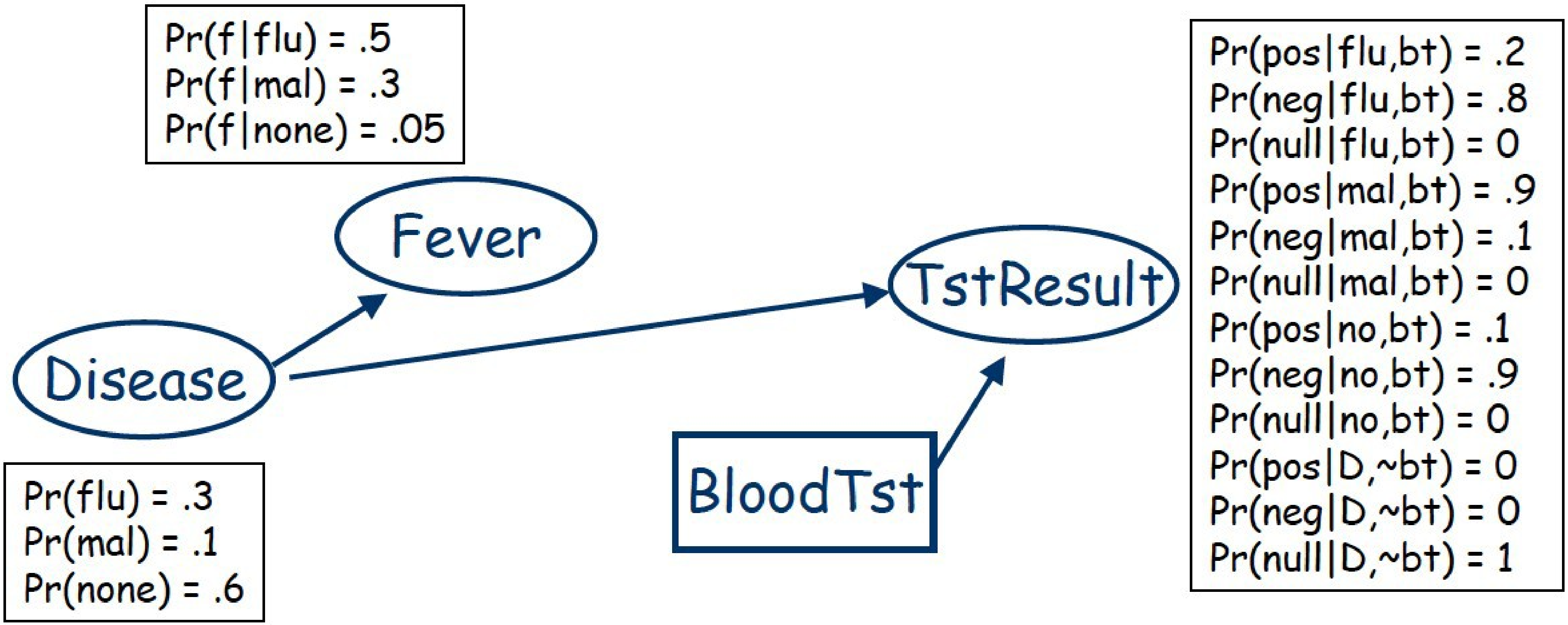


Example: Influence diagram



Chance nodes

- random variables, denoted by circles
- as in a BN, probabilistic dependence on parents

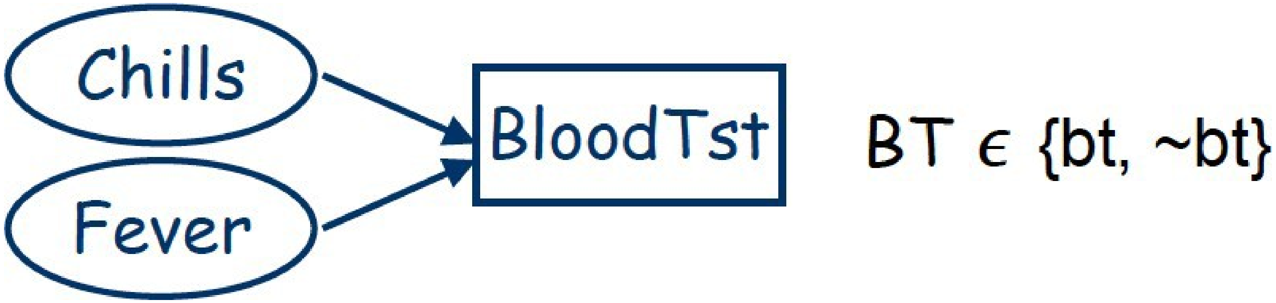


Decision nodes

- variables decision maker sets, denoted by squares
- parents reflect *information available* at time decision is to be made

In example decision node: the actual values of Chills and Fever will be observed before the decision to take test must be made

- agent can make different decisions for each instantiation of parents (i.e., policies)

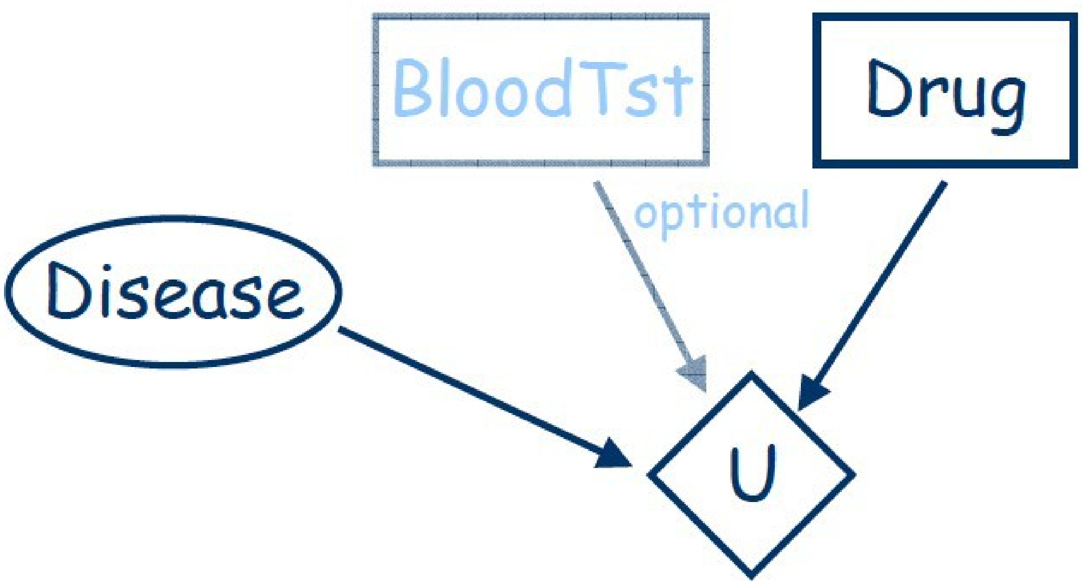


Utility Nodes

Utility node

- specifies utility of a state, denoted by a diamond
- utility depends *only on state of parents* of value node
- generally: only one value node in a decision network

Utility depends only on disease and drug



U(fludrug, flu) = 20
U(fludrug, mal) = -300
U(fludrug, none) = -5
U(maldrug, flu) = -30
U(maldrug, mal) = 10
U(maldrug, none) = -20
U(no drug, flu) = -10
U(no drug, mal) = -285
U(no drug, none) = 30

Influence Diagrams: Assumptions

In Standard Influence Diagrams two assumptions are made:

Decision nodes are totally ordered

- decision variables $D_1, D_2, \ldots, D_n$
- decisions are made in sequence

No-forgetting property

- any information available when decision D_i is made is available when decision D_j is made (for $i < j$)

Thus all parents of D_i are parents of D_j

Example: BloodTest is done before Drug Assignment, and at the time of the Drug Assignment the decision maker is aware of of the result of the Blood Test.

In Non-Standard Influence Diagrams other assumptions may hold

Example: The solution of a **Limited Memory Influence Diagram** (LIMID, used in HUGIN) is a strategy consisting of one policy for each decision. The policy is a function from the known variables to the states of the decision. It is not a function of all past observations as the decision maker is assumed only to know the most recent observation. This is different from the traditional influence diagram where the policy would be a function from all past observations and decisions as the decision maker is assumed to be non-forgetting. **There need not be a total order on the decisions**.

Policies

Let $Par(D_i)$ be the parents of decision node D_i

- $Dom(Par(D_i))$ is the set of assignments to parents

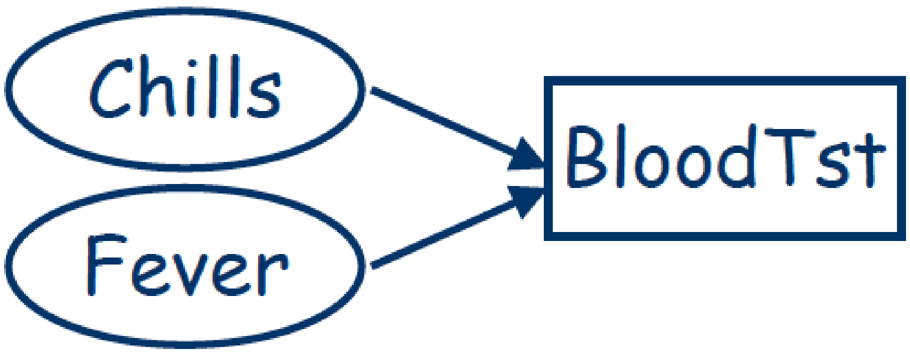
A policy δ is a set of mappings δ_i, one for each decision node D_i

- $\delta_i : Dom(Par(D_i)) \rightarrow (D_i)$

- δ_i associates a decision with each parent assignment for D_i

For example, a policy for BT might be:

$$\delta_{BT}(c, f) = bt$$
$$\delta_{BT}(c, \sim f) = \sim bt$$
$$\delta_{BT}(\sim c, f) = bt$$
$$\delta_{BT}(\sim c, \sim f) = \sim bt$$



Policies

Value of a policy δ is the expected utility given that decision nodes are executed according to δ

Given associates $\boldsymbol{x}$ to the set $\boldsymbol{X}$ of all chance variables, let $\delta(\boldsymbol{x})$ denote the assignment to decision variables dictated by δ

- e.g., assigned to D_1 determined by it's parents' assignment in $\boldsymbol{x}$
- e.g., assigned to D_2 determined by it's parents' assignment in $\boldsymbol{x}$ along with whatever was assigned to D1
- etc.

Value of δ:

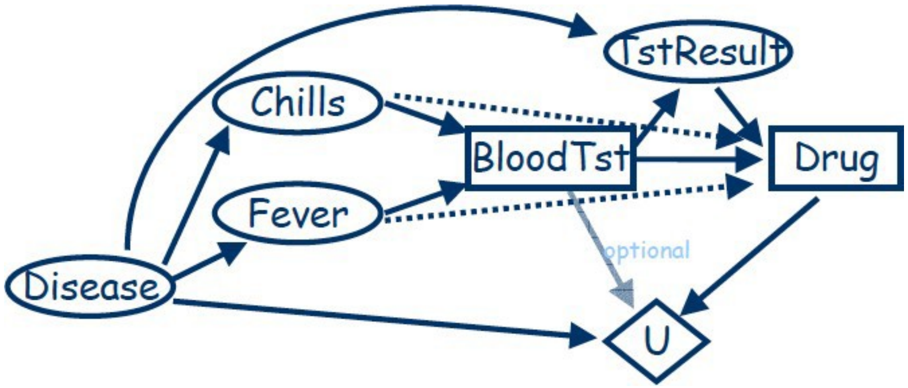
$$EU(\delta) = \sum_{\boldsymbol{X}} P(\boldsymbol{X}, \delta(\boldsymbol{X})U(\boldsymbol{X}, \delta(\boldsymbol{X}))$$

An *optimal policy* is a policy δ^* such that $EU(\delta^*) \geq EU(\delta)$ for all policies δ

We can work backwards as follows

First compute optimal policy for Drug (last decision)

- for each assignment to parents (C,F,BT,TR) and for each decision value (D = md,fd,none), *compute the expected value* of choosing that value of D
- set policy choice for each value of parents to be the value of D that has max value
- eg: $\delta_D(c, f, bt, pos) = md$



Computing the Best Policy

Next compute policy for BT given policy $\delta_D(C, F, BT, TR)$ just determined for Drug

- since $\delta_D(C, F, BT, TR)$ is fixed, we can treat Drug as a normal random variable with deterministic probabilities
- i.e., for any instantiation of parents, value of Drug is fixed by policy δ_D
- this means we can solve for optimal policy for BT just as before
- only uninstantiated variables are random variables (once we fix *its* parents)

Example

You want to buy a used car, but there's a good chance it is a "lemon" (i.e., prone to breakdown). Before deciding to buy it, you can take it to a mechanic for inspection. S/he will give you a report on the car, labelling it either "good" or "bad". A good report is positively correlated with the car being sound, while a bad report is positively correlated with the car being a lemon.

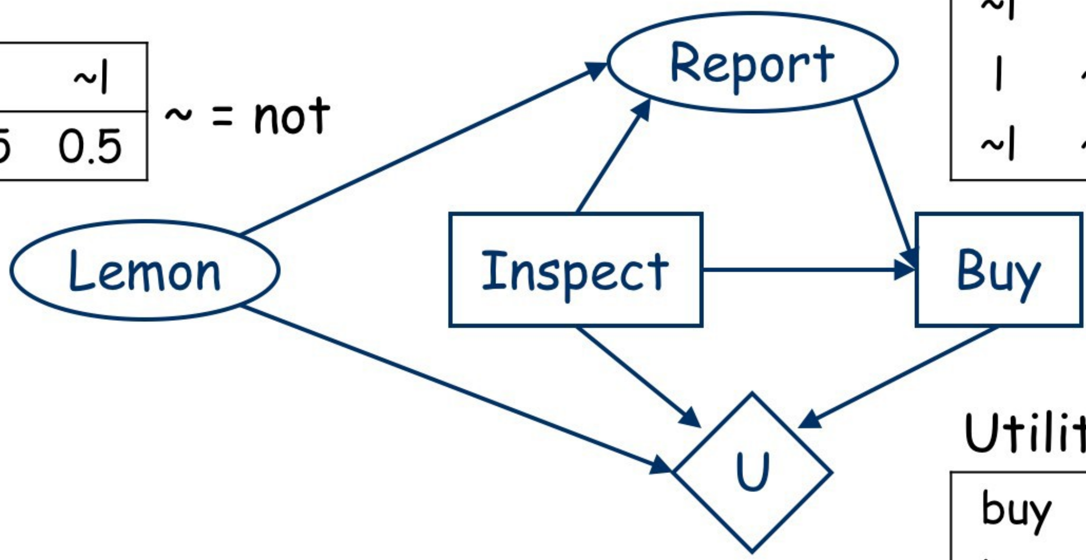
The report costs $50 however. So you could risk it, and buy the car without the report.

Owning a sound car is better than having no car, which is better than owning a lemon.

Car Buyer's Network

l	~l
0.5	0.5

~ = not



Report: good, bad, none

		g	b	n
l	i	0.2	0.8	0
~l	i	0.9	0.1	0
l	~i	0	0	1
~l	~i	0	0	1

Utility for Lemon and Buy

buy	l	-600
buy	~l	1000
~buy	l	-300
~buy	~l	-300

-50 if inspect

Evaluate Last Decision: Buy (1)

$EU(B|I,R) = \sum_L P(L|I,R,B)U(L,B)$

$I = i, R = g$:

$$\begin{aligned} EU(buy) &= P(l|i,g)U(l,buy) + P(\sim l|i,g)U(\sim l,buy) - 50 \\ &= 0.18 \cdot (-600) + 0.82 \cdot 1000 - 50 = 662 \\ EU(\sim buy) &= P(l|i,g)U(l,\sim buy) + P(\sim l|i,g)U(\sim l,\sim buy) - 50 \\ &= -300 - 50 = -350 \end{aligned}$$

So optimal $\delta_{Buy}(i,g) = buy$

$I = i, R = b$:

$$\begin{aligned} EU(buy) &= P(l|i,b)U(l,buy) + P(\sim l|i,b)U(\sim l,buy) - 50 \\ &= 0.89 \cdot (-600) + .11 \cdot 1000 - 50 = -474 \\ EU(\sim buy) &= P(l|i,b)U(l,\sim buy) + P(\sim l|i,b)U(\sim l,\sim buy) - 50 \\ &= -300 - 50 = -350 (-300 \text{ indep. of lemon}) \end{aligned}$$

So optimal $\delta_{Buy}(i,b) = \sim buy$

Evaluate Last Decision: Buy (2)

$I =\sim i,\ R = n$ (note: no inspection cost subtracted):

$$EU(buy) = P(l \mid \sim i, n)U(l, buy) + P(\sim l \mid \sim i, n)U(\sim l, buy)$$
$$= 0.5 \cdot (-600) + 0.5 \cdot 1000 = 200$$
$$EU(\sim buy) = P(l \mid \sim i, n)U(l, \sim buy) + P(\sim l \mid \sim i, n)U(\sim l, \sim buy) - 50$$
$$= -300 - 50 = -350$$

So optimal is: $\delta_{Buy}(\sim i, g) = buy$

So optimal policy for Buy is:

- $\delta_{Buy}(i, g) = buy$; $\delta_{Buy}(i, b) =\sim buy$; $\delta_{Buy}(\sim i, g) = buy$

Note: we don't bother computing policy for the other cases since these occur with probability 0

Evaluate First Decision: Inspect

$EU(I) = \sum_{L,R} P(L,R|I)U(L, \boldsymbol{\delta_{Buy}(I,R)})$,
where $P(R,L|I) = P(R|L,I)P(L|I)$

$$
\begin{aligned}
EU(i) &= 0.1 \cdot (-650) + 0.4 \cdot (-300) + 0.45 \cdot 1000 + 0.05 \cdot (-300) - 50 \\
&= 187.5 \\
EU(\sim i) &= P(l|\sim i, n)U(l, buy) + P(\sim l|\sim i, n)U(\sim l, buy) \\
&= .5 \cdot -600 + .5 \cdot 1000 = 200
\end{aligned}
$$

So optimal $\delta_{Inspect}(\sim i) = buy$

	$P(R,L\|I)$	$\boldsymbol{\delta_{Buy}}$	$U(L, \boldsymbol{\delta_{Buy}})$
g, l	0.1	buy	$-600 - 50 = -650$
$g, \sim l$	0.45	buy	$1000 - 50 = 950$
b, l	0.4	$\sim buy$	$-300 - 50 = -350$
$b, \sim l$	0.05	$\sim buy$	$-300 - 50 = -350$

So optimal policy is: don't inspect, buy the car

- EU = 200
- Notice that the EU of inspecting the car, then buying it iff you get a good report, is 237.5 less the cost of the inspection (50). So inspection not worth the improvement in EU.
- But suppose inspection cost \$25: then it would be worth it ($EU = 237.5-25 = 212.5 > EU(\sim i)$)
- The *expected value of information* associated with inspection is 37.5 (it improves expected utility by this amount ignoring cost of inspection). Gives opportunity to change decision ($\sim$ *buy* if bad).
- You should be willing to pay up to \$37.5 for the report