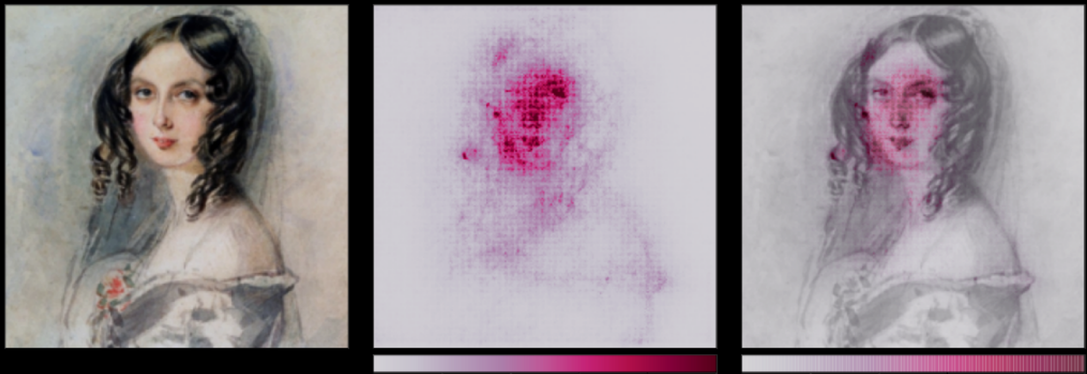
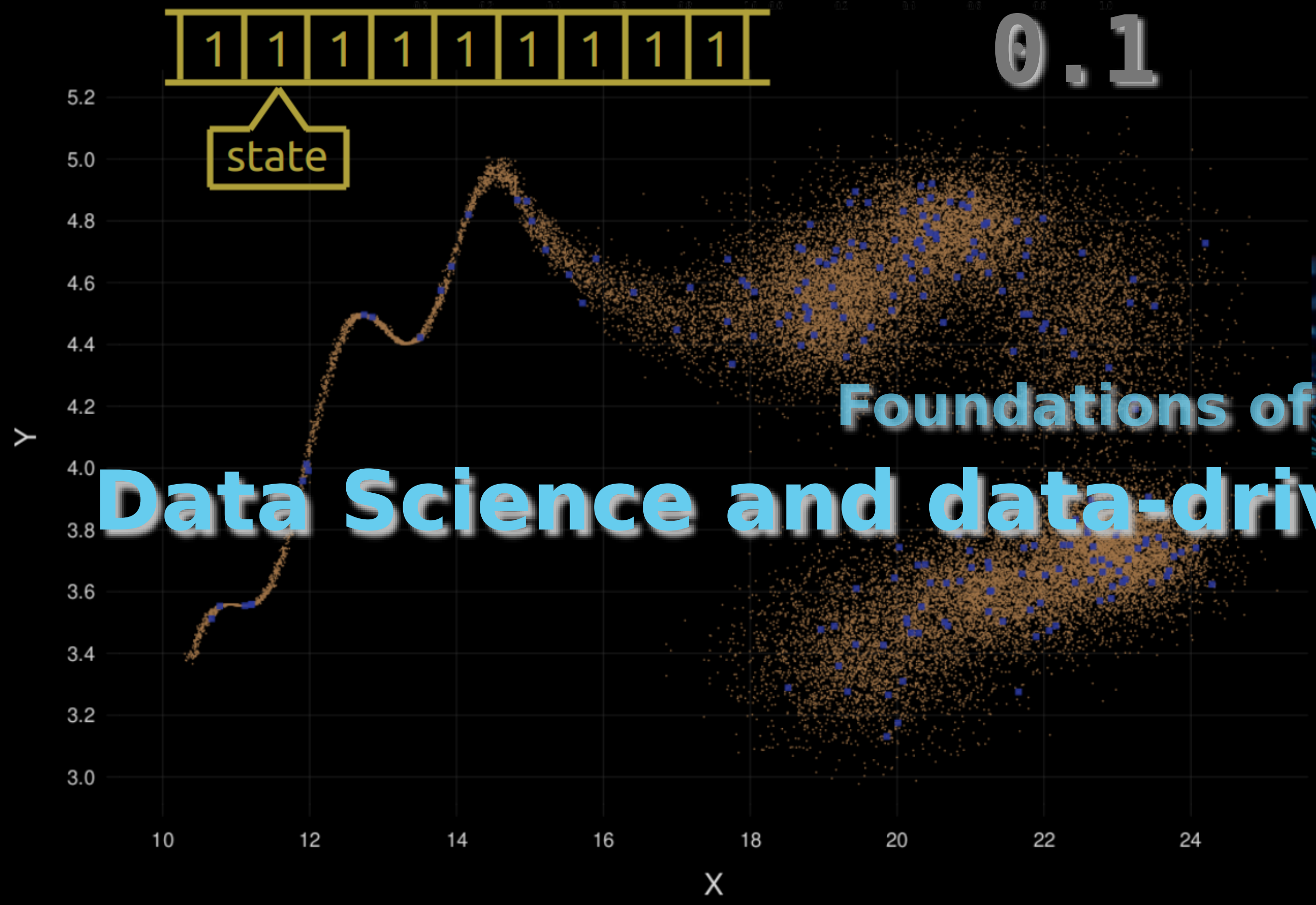
HVL Artificial Intelligence Engineering



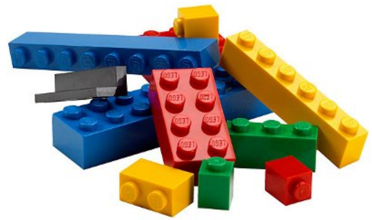


Foundations of Data Science and data-driven engineering

$$P(Y \mid X)$$

Today’s goal

Find the building blocks for a decision-making AI



~~Build an optimal decision-making AI~~

not yet!

⚠ The goal of the next exercise is **NOT** to get the “correct” answer!

🧠🌱 The goal is for you to do some **introspection**:

- *How am I reasoning to solve this problem?*
- *Is any of my reasoning, on second thought, misleading or illogical?*
- *Could my reasoning be implemented as an algorithm?*
- *Which information am I using to find the solution?*
- *Is there irrelevant information?*
- *. . . ?*

Accept or discard?

A particular kind of electronic component is produced on an assembly line. At the end of the line, there is an automated inspection device that works as follows with every newly produced component coming out of the line:

The inspection device first makes some tests on the new component. The tests give an uncertain forecast of whether that component will either *fail within its first year of use*, or *after*.

Then the device decides whether the component is accepted and packaged for sale, or discarded and thrown away.

When a new electronic component is sold, the manufacturer has a net *gain* of **1$**. If the component fails within a year of use, however, the manufacturer incurs a net *loss* of **11$** (12$ loss, minus the 1$ gained at first), owing to warranty refunds and damage costs to be paid to the buyer. When a new electronic component is discarded, the manufacturer has **0$** net gain.

For a new electronic component just come out of the assembly line, the tests of the automated inspection device indicate that there is a **10%** probability that the component will fail *within its first year* of use, and **90%** that it will fail *after*.

Should the inspection device ***accept*** *the new component? or* ***discard*** *it?*

("not choosing" is the same as accepting)

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail

+1$

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail not-fail

+1$ +1$

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail
+1$	+1$	+1$	+1$	+1$	+1$	–11$

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$	= **-2$**

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$	= **-2$**

“gain” per component = **-0.2$**

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$	= **-2$**

“gain” per component = **-0.2$**

What if, in **10** scenarios exactly like this, we would **always discard**?

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1\$	+1\$	+1\$	+1\$	+1\$	+1\$	–11\$	+1\$	+1\$	+1\$	= **-2\$**

“gain” per component = **-0.2\$**

What if, in **10** scenarios exactly like this, we would **always discard**?

not-fail
0\$

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$	= **-2$**

“gain” per component = **-0.2$**

What if, in **10** scenarios exactly like this, we would **always discard**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail
0$	0$	0$	0$	0$	0$	0$

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$	= **-2$**

“gain” per component = **-0.2$**

What if, in **10** scenarios exactly like this, we would **always discard**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail
0$	0$	0$	0$	0$	0$	0$	0$	0$	0$

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$	= **-2$**

“gain” per component = **-0.2$**

What if, in **10** scenarios exactly like this, we would **always discard**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
0$	0$	0$	0$	0$	0$	0$	0$	0$	0$	= **0$**

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$	= **-2$**

“gain” per component = **-0.2$**

What if, in **10** scenarios exactly like this, we would **always discard**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
0$	0$	0$	0$	0$	0$	0$	0$	0$	0$	= **0$**

“gain” per component = **0$**

Intuitive approach

What if, in **10** scenarios exactly like this, we would **always accept**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
+1$	+1$	+1$	+1$	+1$	+1$	–11$	+1$	+1$	+1$	= **-2$**

“gain” per component = **-0.2$**

best decision

in *these* circumstances!

What if, in **10** scenarios exactly like this, we would **always discard**?

not-fail	not-fail	not-fail	not-fail	not-fail	not-fail	fail	not-fail	not-fail	not-fail	
0$	0$	0$	0$	0$	0$	0$	0$	0$	0$	= **0$**

“gain” per component = **0$**

Scenario 2

- ACCEPT

& fail	**-5$**
& not-fail	**+1$**

- DISCARD

& fail	**0$**
& not-fail	**0$**

• fail < 1yr	**10%**
• not fail < 1yr	**90%**

Scenario 3

- ACCEPT

  & fail **-11$**

  & not-fail **+1$**

- DISCARD

  & fail **0$**

  & not-fail **0$**

- fail < 1yr **5%**
- not fail < 1yr **95%**

Scenario 4

- ACCEPT

& fail	**-11\$**
& not-fail	**+2\$**

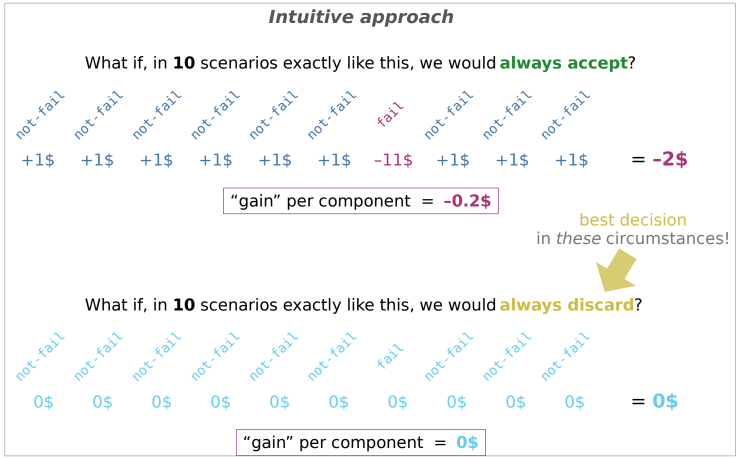
- DISCARD

& fail	**0\$**
& not-fail	**0\$**

- fail < 1yr **10%**
- not fail < 1yr **90%**

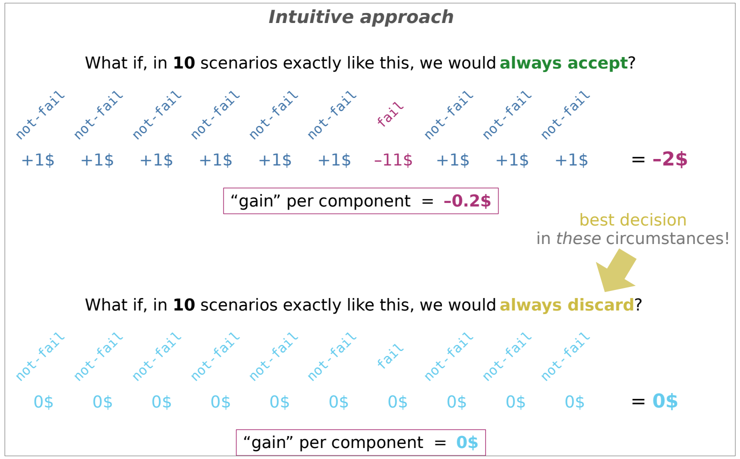
Is our “intuitive” approach fully acceptable?

Problems & limitations?



Is our “intuitive” approach fully acceptable?

Problems & limitations?



What if the situation is somewhat unique and *cannot* be replicated?

(important examples: medical decisions)

Decision-making under uncertainty

We’d like to build a framework that can be universally applied

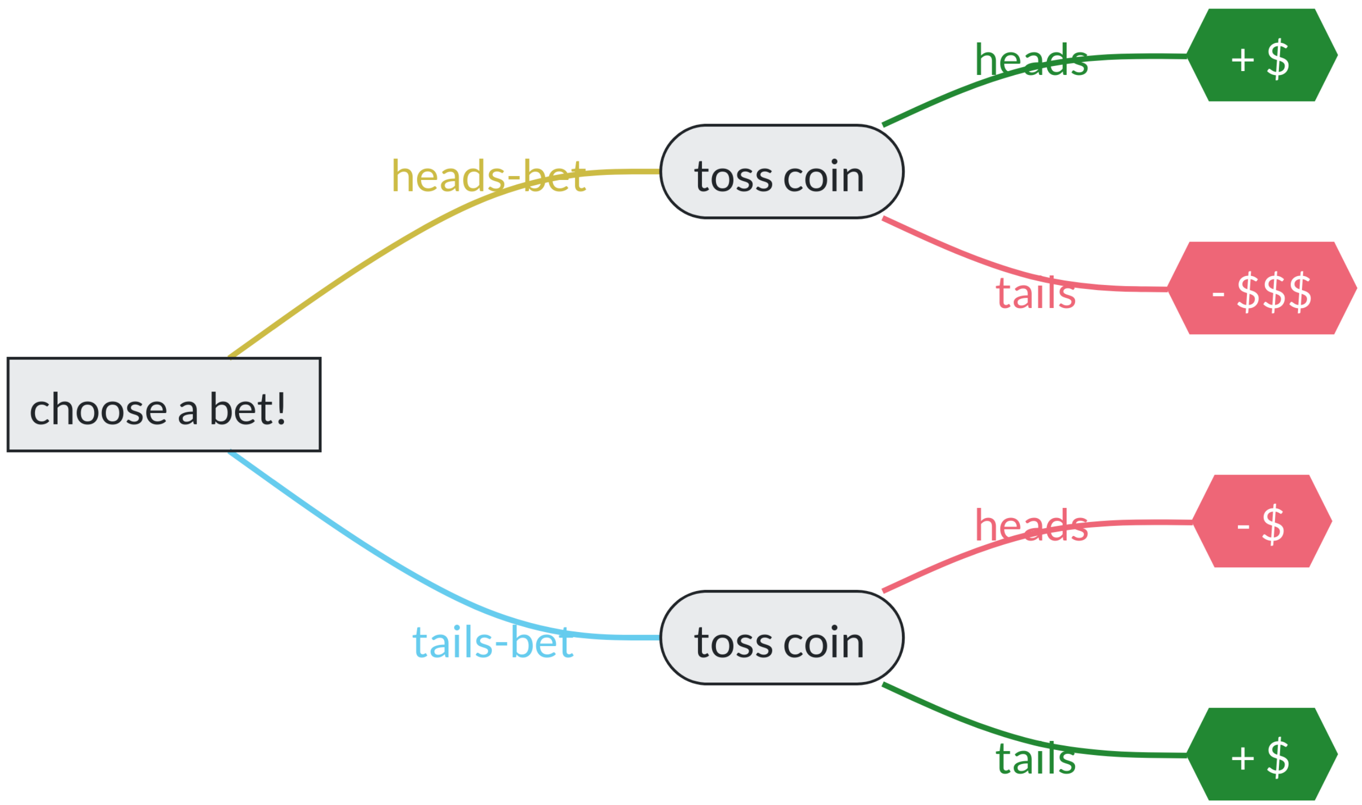
What should this framework achieve?

It should be successful – tell what’s the winning decision!

Choose your bet!

- **“heads-bet”**
  - if the coin lands heads → you win a small amount of money
  - if the coin lands tails → you lose a large amount of money

- **“tails-bet”**
  - if the coin lands tails → you win a small amount of money
  - if the coin lands heads → you lose a small amount of money



Decision-making under uncertainty

We’d like to build a framework that can be universally applied

Features

- Must handle uncertainty (no shit, Sherlock!)
- Must handle decisions (no shit, Sherlock!) and gains/losses
- Optimal
- Modular, recursive
- Algorithmic, can be automated
- Use all available information (learning)
- Set a standard

There actually is such a framework!

Decision Theory

There actually is such a framework!

Decision Theory

Utility Theory

There actually is such a framework!

Decision Theory

Utility Theory

Probability Theory

(Belief Theory)

There actually is such a framework!

Decision Theory

Utility Theory

Probability Theory
(Belief Theory)

Theorem:

Decision Theory is ***the*** *normative* theory of decision-making under uncertainty

There actually is such a framework!

Decision Theory

Utility Theory

Probability Theory
(Belief Theory)

Theorem:

Decision Theory is ***the*** *normative* theory of decision-making under uncertainty

Any other theory:

- ⮥ either it's equivalent to Decision Theory
  (that is, it's Decision Theory but presented with different math clothes and terminology)
- ⤵ or it leads to logically inconsistent or sub-optimal decisions

Basic decision problem

Building blocks

Basic decision problem

Building blocks

Agent

Basic decision problem

Building blocks

- **Agent**
- Background (prior) information

Basic decision problem

Building blocks

- **Agent**
- Background (prior) information
- List of uncertain **outcomes**

Basic decision problem

Building blocks

- **Agent**
- Background (prior) information
- List of uncertain **outcomes**
- List of **decisions** (courses of action)

Basic decision problem

Building blocks

- **Agent**
- Background (prior) information
- List of uncertain **outcomes**
- List of **decisions** (courses of action)
- **Probabilities** of outcomes

Basic decision problem

Building blocks

- **Agent**
- Background (prior) information
- List of uncertain **outcomes**
- List of **decisions** (courses of action)
- **Probabilities** of outcomes
- **Utilities** of decisions & outcomes

Basic decision problem

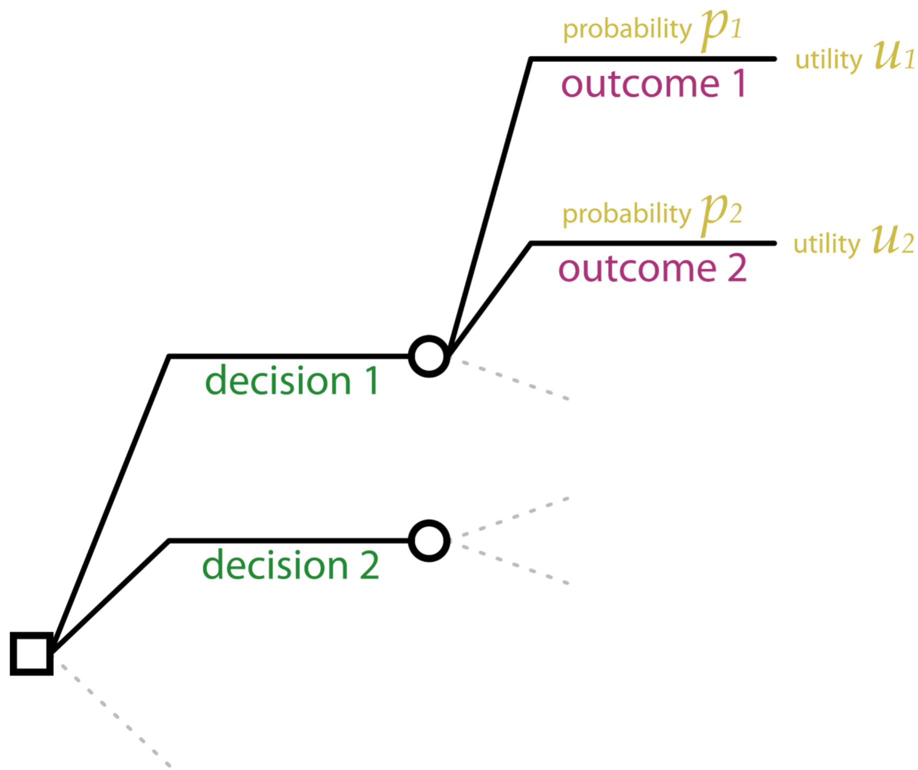
Building blocks

- **Agent**
- Background (prior) information
- List of uncertain **outcomes**
- List of **decisions** (courses of action)
- **Probabilities** of outcomes
- **Utilities** of decisions & outcomes
- Information & data

Basic decision problem

Building blocks

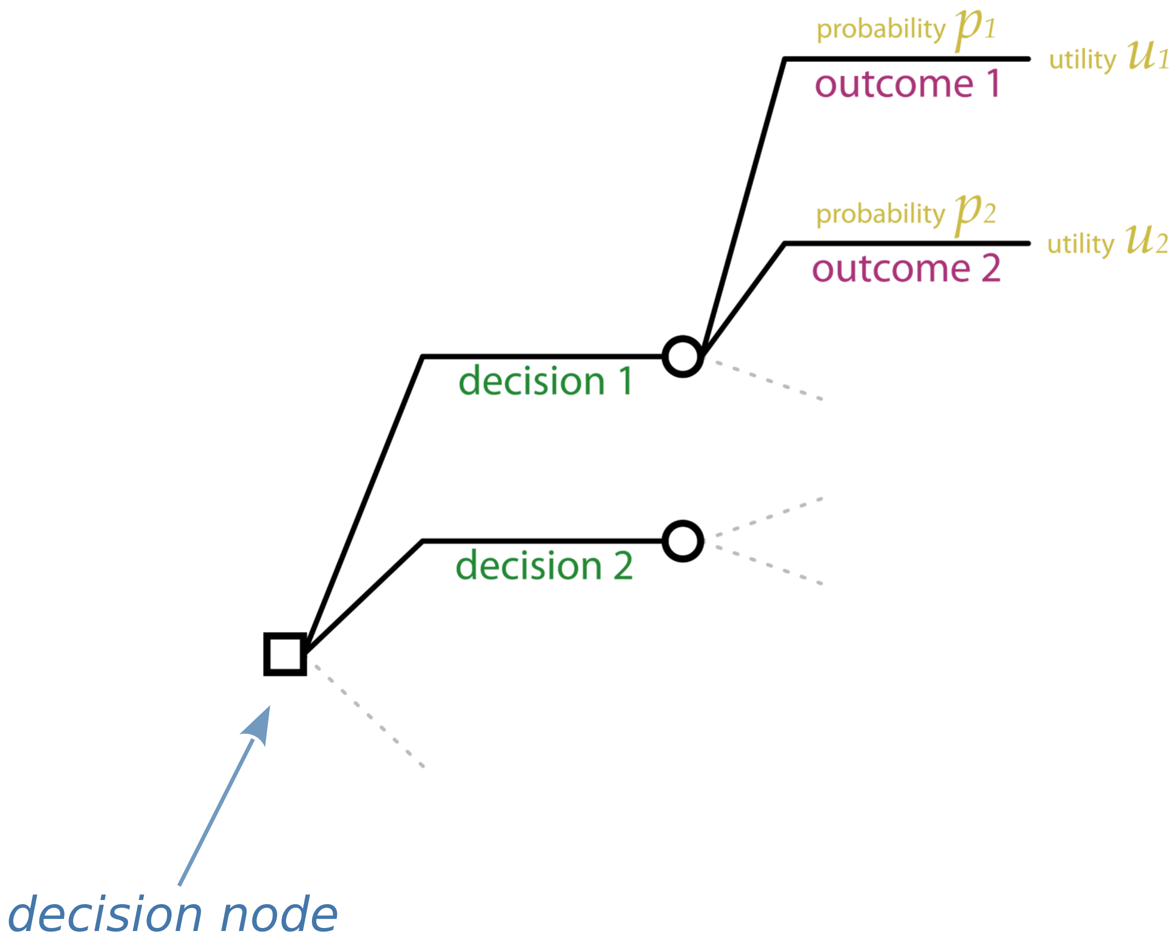
- **Agent**
- Background (prior) information
- List of uncertain **outcomes**
- List of **decisions** (courses of action)
- **Probabilities** of outcomes
- **Utilities** of decisions & outcomes
- Information & data



Basic decision problem

Building blocks

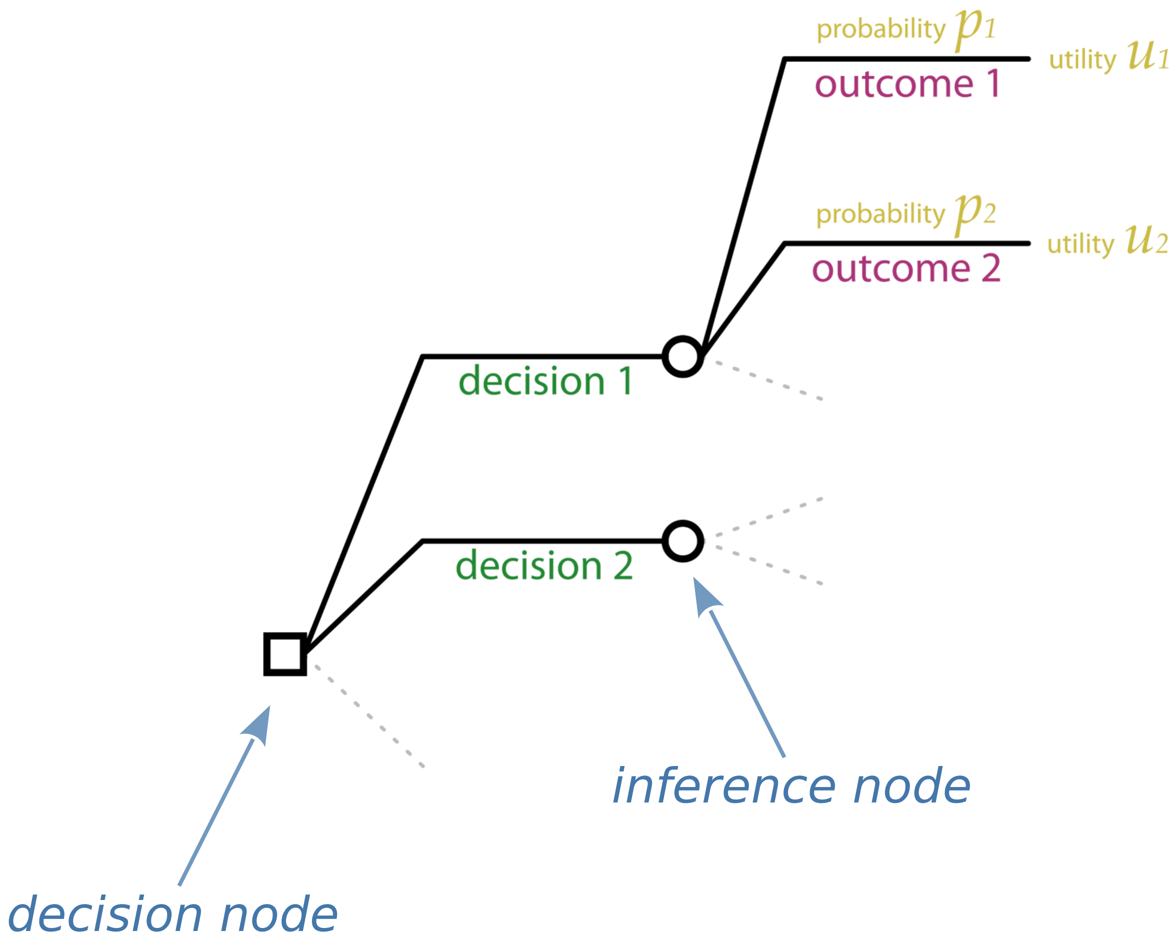
- **Agent**
- Background (prior) information
- List of uncertain **outcomes**
- List of **decisions** (courses of action)
- **Probabilities** of outcomes
- **Utilities** of decisions & outcomes
- Information & data



Basic decision problem

Building blocks

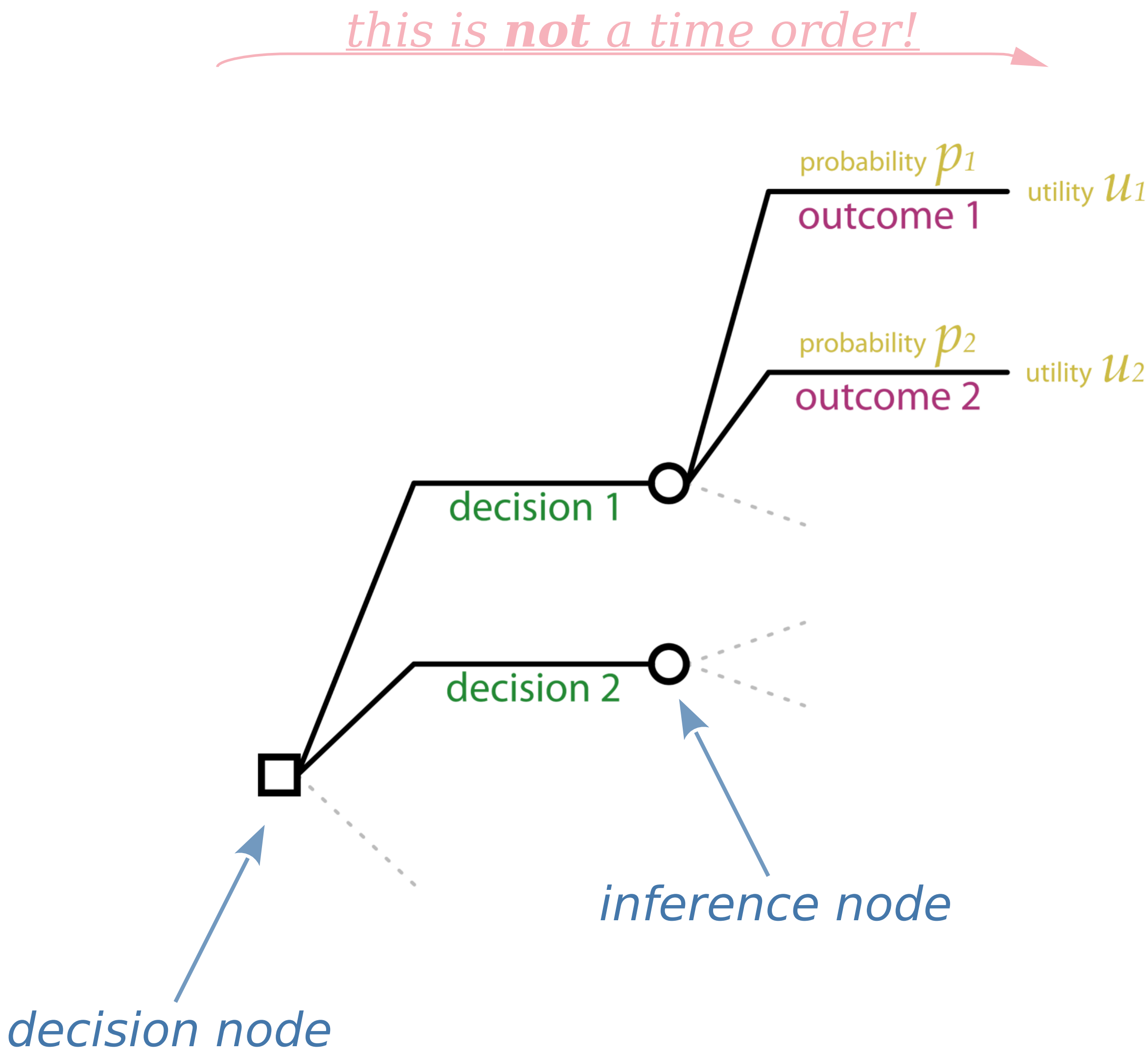
- **Agent**
- Background (prior) information
- List of uncertain **outcomes**
- List of **decisions** (courses of action)
- **Probabilities** of outcomes
- **Utilities** of decisions & outcomes
- Information & data

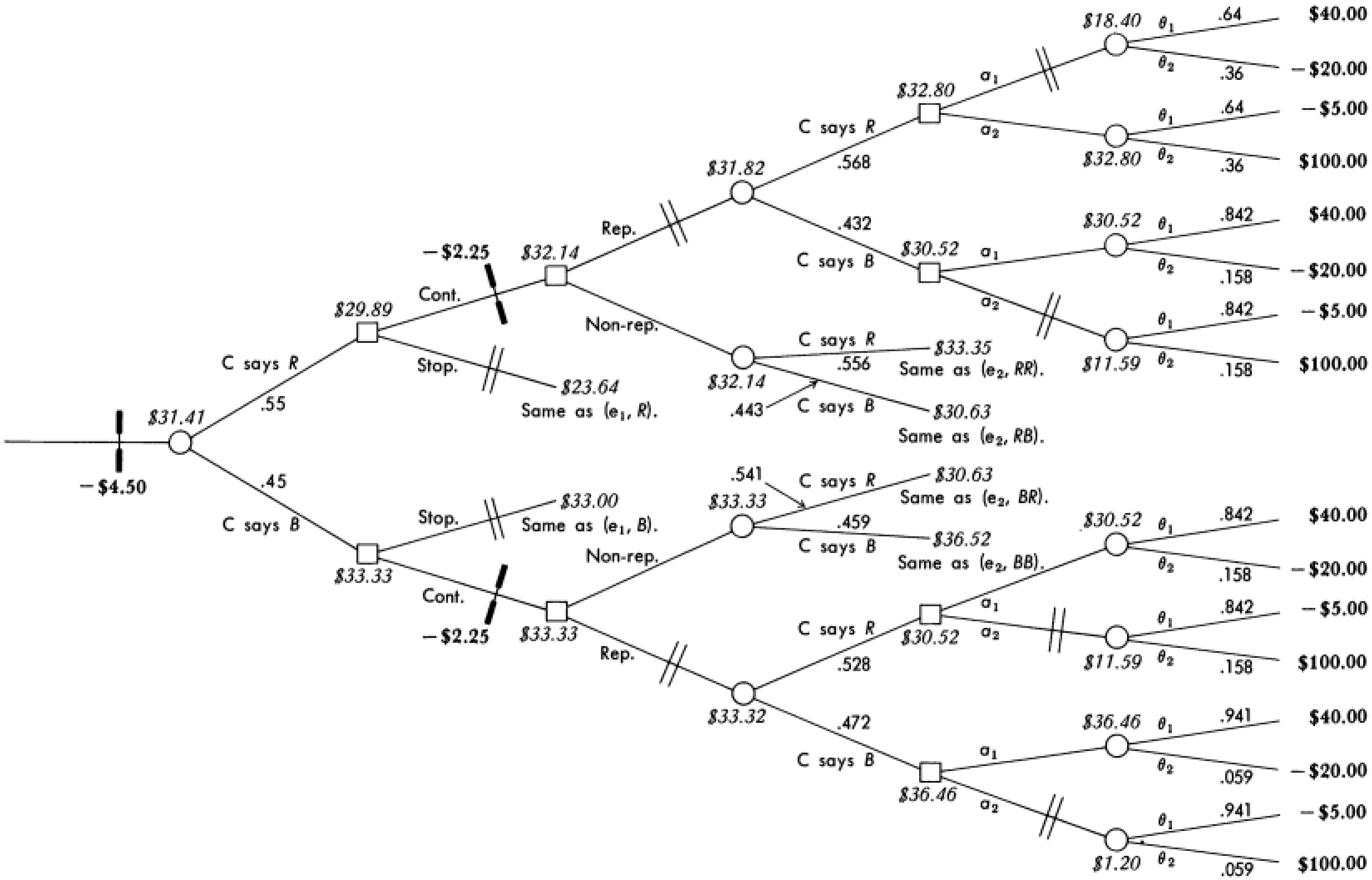


Basic decision problem

Building blocks

- **Agent**
- Background (prior) information
- List of uncertain **outcomes**
- List of **decisions** (courses of action)
- **Probabilities** of outcomes
- **Utilities** of decisions & outcomes
- Information & data





$40.00
−$20.00
−$5.00
$100.00
$40.00
−$20.00
−$5.00
$100.00
$40.00
−$20.00
−$5.00
$100.00
$40.00
−$20.00
−$5.00
$100.00
$18.40
$32.80
$32.80
$30.52
$30.52
$11.59
$31.82
$32.14
$29.89
$31.41
−$4.50
−$2.25
−$2.25
$23.64
Same as (e1, R).
$33.00
Same as (e1, B).
$33.35
Same as (e2, RR).
$30.63
Same as (e2, RB).
$30.63
Same as (e2, BR).
$36.52
Same as (e2, BB).
$32.14
$33.33
$33.33
$33.33
$33.32
$30.52
$30.52
$11.59
$36.46
$36.46
$1.20
C says R
C says B
Cont.
Stop.
Rep.
Non-rep.
.55
.45
.568
.432
.556
.443
.541
.459
.528
.472
.64
.36
.842
.158
.941
.059

probability × utility

↕

Optimal decision

↕

Optimal decision

probability × utility

↕

Optimal decision

In Data Science, whenever you stumble into a problem that seems to involve choice, ask:

In Data Science, whenever you stumble into a problem that seems to involve choice, ask:

🤖 Who has to make the decision? Who lacks certainty?

In Data Science, whenever you stumble into a problem that seems to involve choice, ask:

🤖 Who has to make the decision? Who lacks certainty?

What are the final possible decisions?

In Data Science, whenever you stumble into a problem that seems to involve choice, ask:

Who has to make the decision? Who lacks certainty?

What are the final possible decisions?

What is uncertain?

In Data Science, whenever you stumble into a problem that seems to involve choice, ask:

Who has to make the decision? Who lacks certainty?

What are the final possible decisions?

What is uncertain?

What are the gain/losses involved in making different decisions?

In Data Science, whenever you stumble into a problem that seems to involve choice, ask:

Who has to make the decision? Who lacks certainty?

What are the final possible decisions?

What is uncertain?

What are the gain/losses involved in making different decisions?

Utilities are still much underappreciated in machine learning. They are not examined, or examined only qualitatively