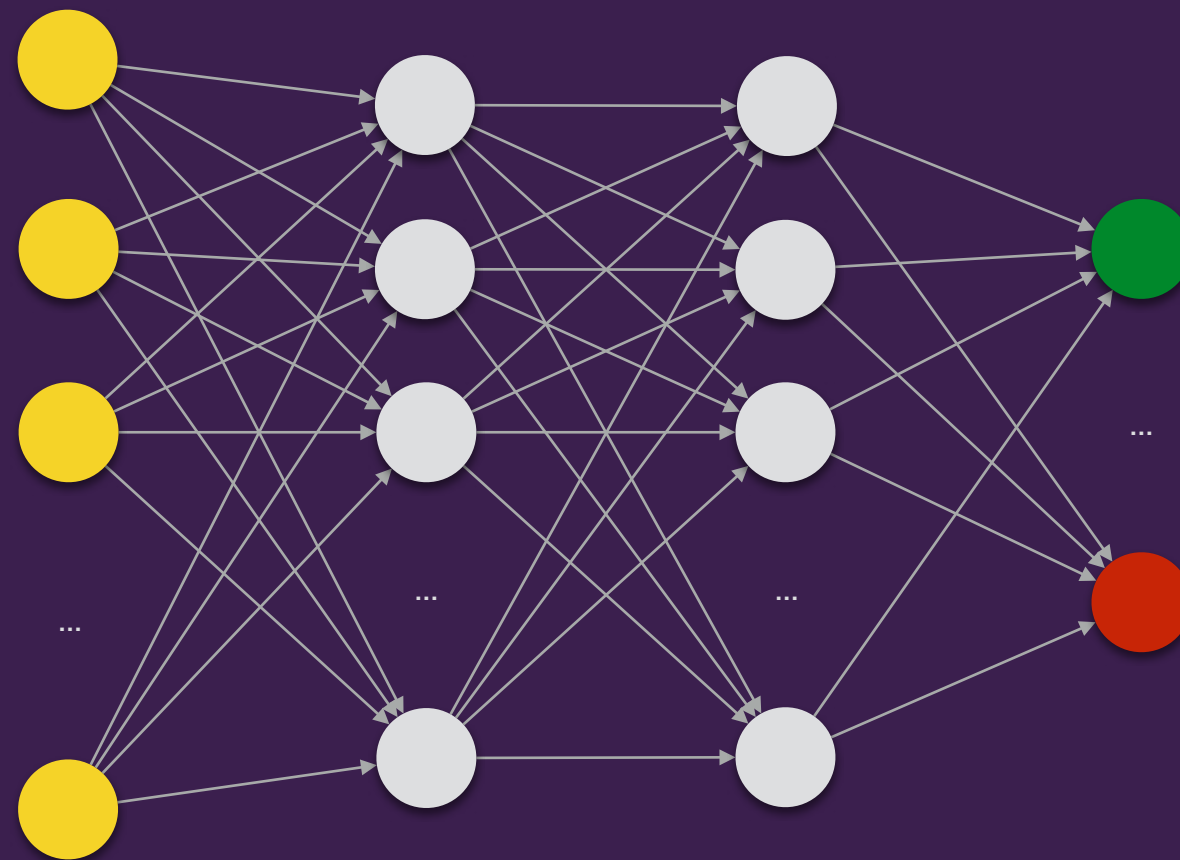


Perfectly Parallel Fairness Certification of Neural Networks



Caterina Urban

ANTIQUE Research Team

INRIA & École Normale Supérieure | Université PSL





WIRED

BUSINESS

MORE SIGN IN

SUBSCRIBE

ERIC NIILER

BUSINESS

03.25.2019 07:00 AM

Can AI Be a Fair Judge in Court? Estonia Thinks So

Estonia plans to use an artificial intelligence program to decide some small-claims cases, part of a push to make government services smarter.

WIRED

In 2019, predictive algorithms will start to make banking fair for all



AUTOMATED BACKGROUND CHECKS ARE DECIDING WHO'S FIT FOR A HOME

By [Colin Lecher](#) | [@colinlecher](#) | Feb 1, 2019, 8:00am EST

The AI doctor will see you now

Medicine is at the point computer-driven financial trading was in the early 2000s

BROOKE MASTERS

The Telegraph

AI used for first time in job interviews in UK to find best applicants

By Charles Hymas

27 SEPTEMBER 2019 • 10:00 PM





WIRED

BUSINESS

MORE ▾ SIGN IN

SUBSCRIBE

ERIC NIILER

BUSINESS

03.25.2019 07:00 AM

Can AI Be a Fair Judge in Court? Estonia Thinks So

Machine Bias

There's software used across the country to predict future criminals. And it's biased against blacks.

by Julia Angwin, Jeff Larson, Surya Mattu and Lauren Kirchner, ProPublica

May 23, 2016



The AI doctor will see you now

Medicine is at the point computer-driven financial trading was in the early

Millions of black people affected by racial bias in health-care algorithms

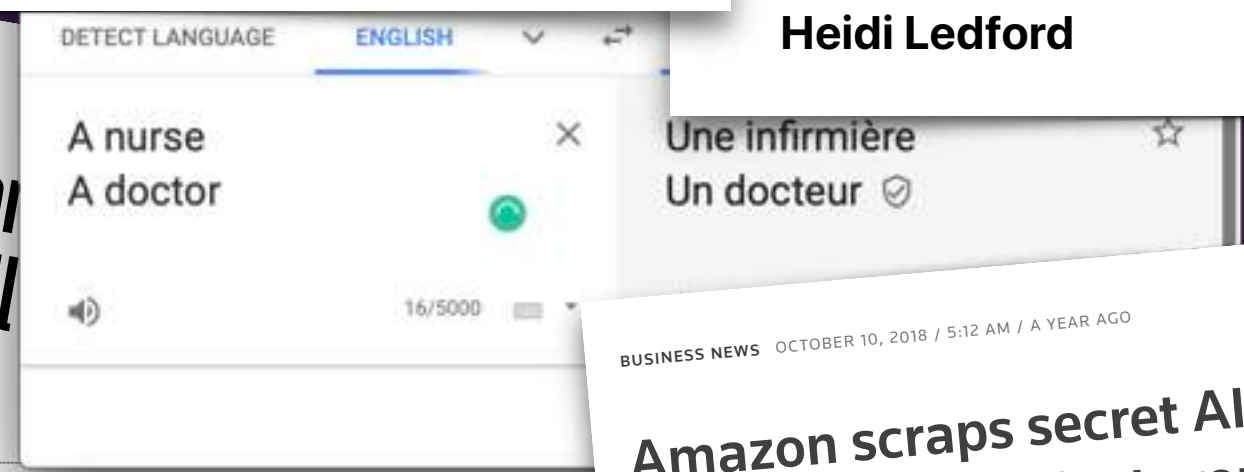
Study reveals rampant racism in decision-making software used by US hospitals — and highlights ways to correct it.

24 October 2019

NEWS

Heidi Ledford

In 2019, predictive algo
make banking fair for all



AUTOMATED BACKGROUND CHECKS ARE DECIDING WHO'S FIT FOR A HOME

Amazon scraps secret AI recruiting tool that showed bias against women

Jeffrey Dastin

China 'social credit': Beijing sets up huge system

By Celia Hatton
BBC News, Beijing

26 October 2015

AI used for first time in job interviews in UK to find best applicants

By Charles Hymas

27 SEPTEMBER 2019 • 10:00 PM



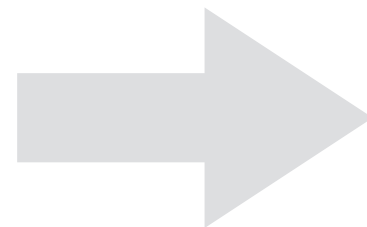
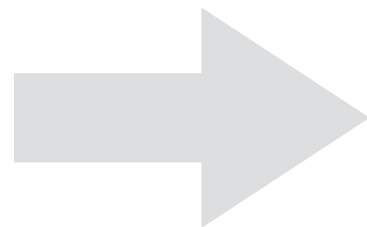
Fairness Certification of Machine Learning Systems is Now Critical!



Feed Forward Neural Networks

Classification of Tabular Data

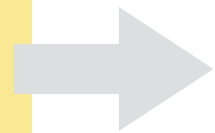
**Fairness Certification
of Machine Learning Systems
is Now Critical!**



Data Science Pipeline



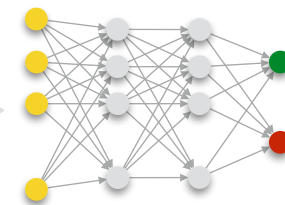
Data is Dirty



pre-processing



training



data analysis



inconsistent data

incorrect data

incomplete data

inaccurate data

Pre-Processing is Fragile



pre-processing



mislabeled data

accidentally duplicated data

wrongly converted data

accidentally (un)used data

TECHNOLOGY

The New York Times

For Big-Data Scientists, 'Janitor Work' Is Key Hurdle to Insights

By Steve Lohr

Aug. 17, 2014

Technology revolutions come in measured, sometimes foot-dragging steps. The lab science and marketing enthusiasm tend to underestimate the bottlenecks to progress that must be overcome with hard work and practical engineering.

The field known as “big data” offers a contemporary case study. The catchphrase stands for the modern abundance of digital data from many sources — the web, sensors, smartphones and corporate databases — that can be mined with clever software for discoveries and insights. Its promise is smarter, data-driven decision-making in every field. That is why data scientist is the economy’s hot new job.

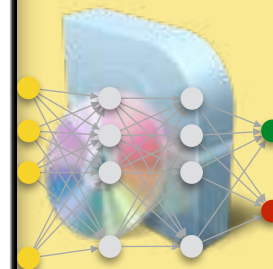
Yet far too much handcrafted work — what data scientists call “data wrangling,” “data munging” and “data janitor work” — is still required. Data scientists, according to interviews and expert estimates, spend from 50 percent to 80 percent of their time mired in this more mundane labor of collecting and preparing unruly digital data, before it can be explored for useful nuggets.

“Data wrangling is a huge — and surprisingly so — part of the job,” said Monica Rogati, vice president for data science at Jawbone, whose sensor-filled wristband and software track activity, sleep and food consumption, and suggest dietary and health tips based on the numbers. “It’s something that is not appreciated by data civilians. At times, it feels like everything we do.”

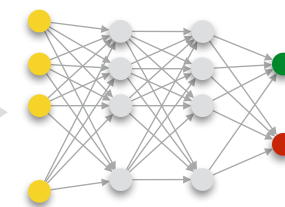
Several start-ups are trying to break through these big data bottlenecks by developing software to automate the gathering, cleaning and organizing of disparate data, which is plentiful but messy. The modern Wild West of data needs to be tamed somewhat so it can be recognized and exploited by a computer program.

“It’s an absolute myth that you can send an algorithm over raw data and have insights pop up,” said Jeffrey Heer, a professor of computer science at the University of Washington and a co-founder of Trifacta, a start-up based in San Francisco.

Accuracy is Meaningless



training



data analysis



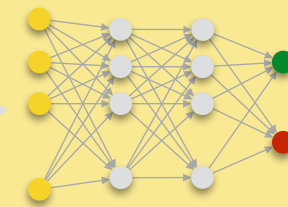
Inscrutability



pre-processing



training



data analysis



MIT Technology Review



Artificial Intelligence / Machine Learning

The Dark Secret at the Heart of AI

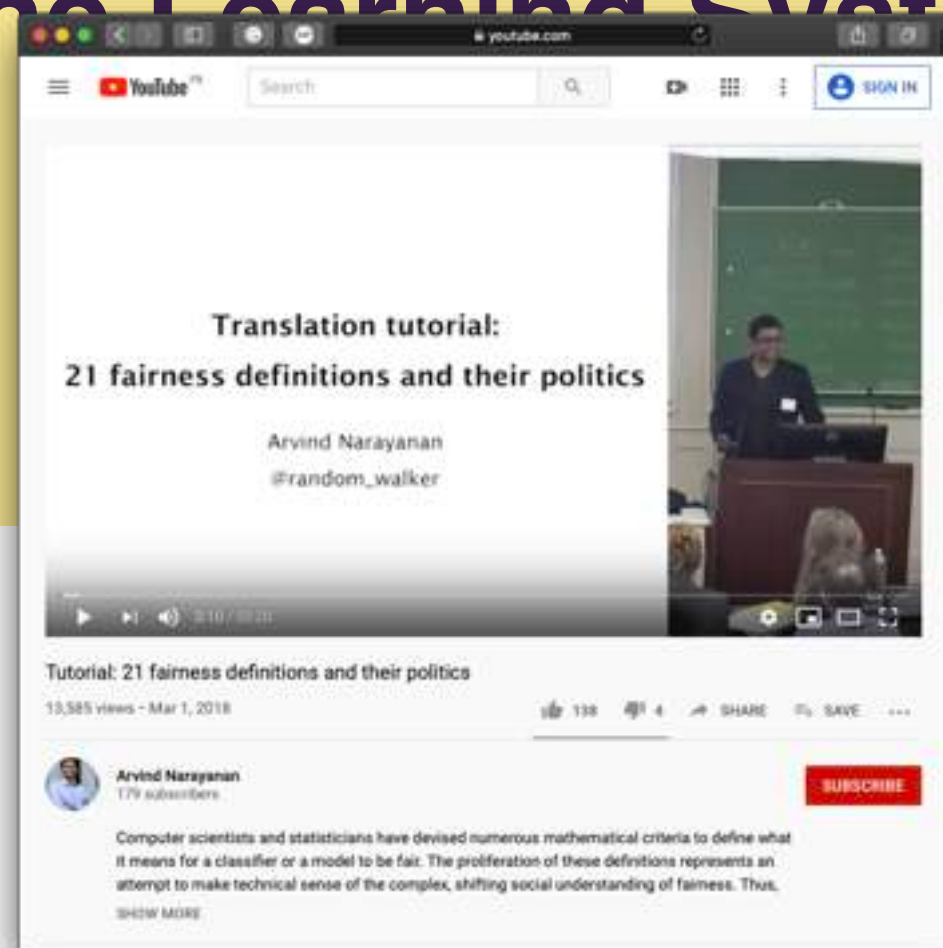
No one really knows how the most advanced algorithms do what they do. That could be a problem.

by Will Knight

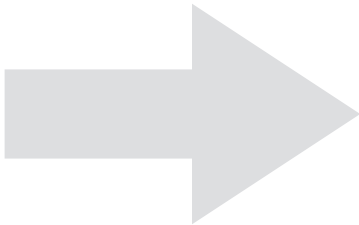
Apr 11, 2017



Fairness Certification of Machine Learning Systems is



Causal Fairness



Static Analysis by Abstract Interpretation



AIRBUS



AREVA

HELBACO



Fairness Certification

of Machine Learning Systems

is not a Fair Game!

ABSTRACT INTERPRETATION : A UNIFIED LATTICE MODEL FOR STATIC ANALYSIS
OF PROGRAMS BY CONSTRUCTION OR APPROXIMATION OF FIXPOINTS

Patrick Cousot* and Radhia Cousot**

Laboratoire d'informatique, U.S.M.G., BP. 53
38041 Grenoble cedex, France

1. Introduction

A program denotes computations in some universe of objects. Abstract interpretation of programs consists in using that denotation to describe computations in another universe of abstract objects, so that the results of abstract execution give some informations on the actual computations. An intuitive example (which we borrow from Sintzoff [72]) is the rule of signs. The text $-1515 * 17$ may be understood to denote computations on the abstract universe $\{(+), (-), (\pm)\}$ where the semantics of arithmetic operators is defined by the rule of signs. The abstract execution $-1515 * 17 \Rightarrow -(+) * (+) \Rightarrow (-) * (+) \Rightarrow (-)$, proves that $-1515 * 17$ is a negative number. Abstract interpretation is concerned by a particular underlying structure of the usual universe of computations (the sign, in our example). It gives a summary of some facets of the actual executions of a program. In general this summary is simple to obtain but inaccurate (e.g. $-1515 * 17 \Rightarrow -(+) * (+) \Rightarrow$

Abstract program properties are modeled by a complete semilattice, Birkhoff[61]. Elementary program constructs are locally interpreted by order preserving functions which are used to associate a system of recursive equations with a program. The program global properties are then defined as one of the extreme fixpoints of that system, Tarski[55]. The abstraction process is defined in section 6. It is shown that the program properties obtained by an abstract interpretation of a program are consistent with those obtained by a more refined interpretation of that program. In particular, an abstract interpretation may be shown to be consistent with the formal semantics of the language. Levels of abstraction are formalized by showing that consistent abstract interpretations form a lattice (section 7). Section 8 gives a constructive definition of abstract properties of programs based on constructive definitions of fixpoints. It shows that various classical algorithms such as Kildall [73], Wegbreit[75] compute program properties as limits of finite Kleene[52]'s sequences. Section



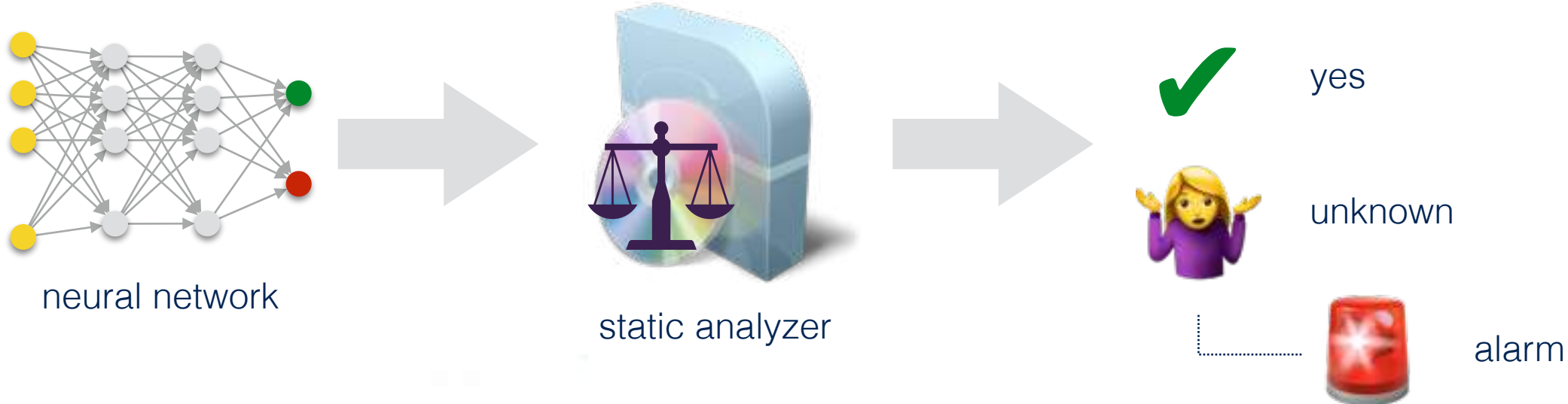
Radhia Cousot



Patrick Cousot

Relaxing the Problem

Over-Approximation



A shopping cart filled with various food items, each with a price tag. The items and their prices are:

- Raspberries: ~~€ 5.35~~ **€ 6**
- Cookie: ~~€ 2.75~~ **€ 3**
- Butter: ~~€ 2.95~~ **€ 3**
- Ice cream: ~~€ 3.65~~ **€ 4**

A collection of Euro banknotes (20, 10, and 5) and a yellow warning triangle with an exclamation mark. A red alarm icon is also present. The text **false alarm** is displayed next to the warning triangle.

Static Analysis Recipe

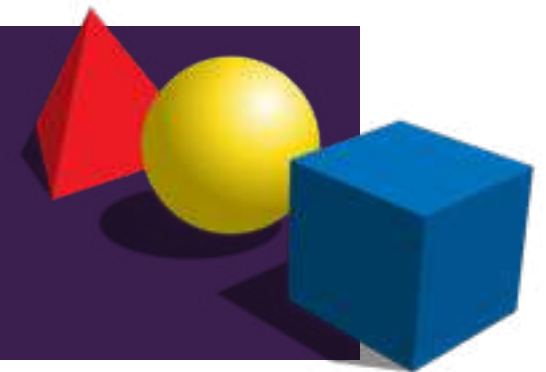
practical tools

targeting specific programs



algorithmic approaches

to decide program properties



mathematical models

of the program behavior



Static Analysis Recipe

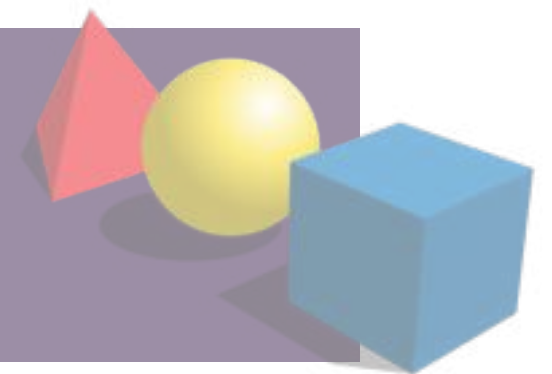
practical tools

targeting specific programs



algorithmic approaches

to decide program properties



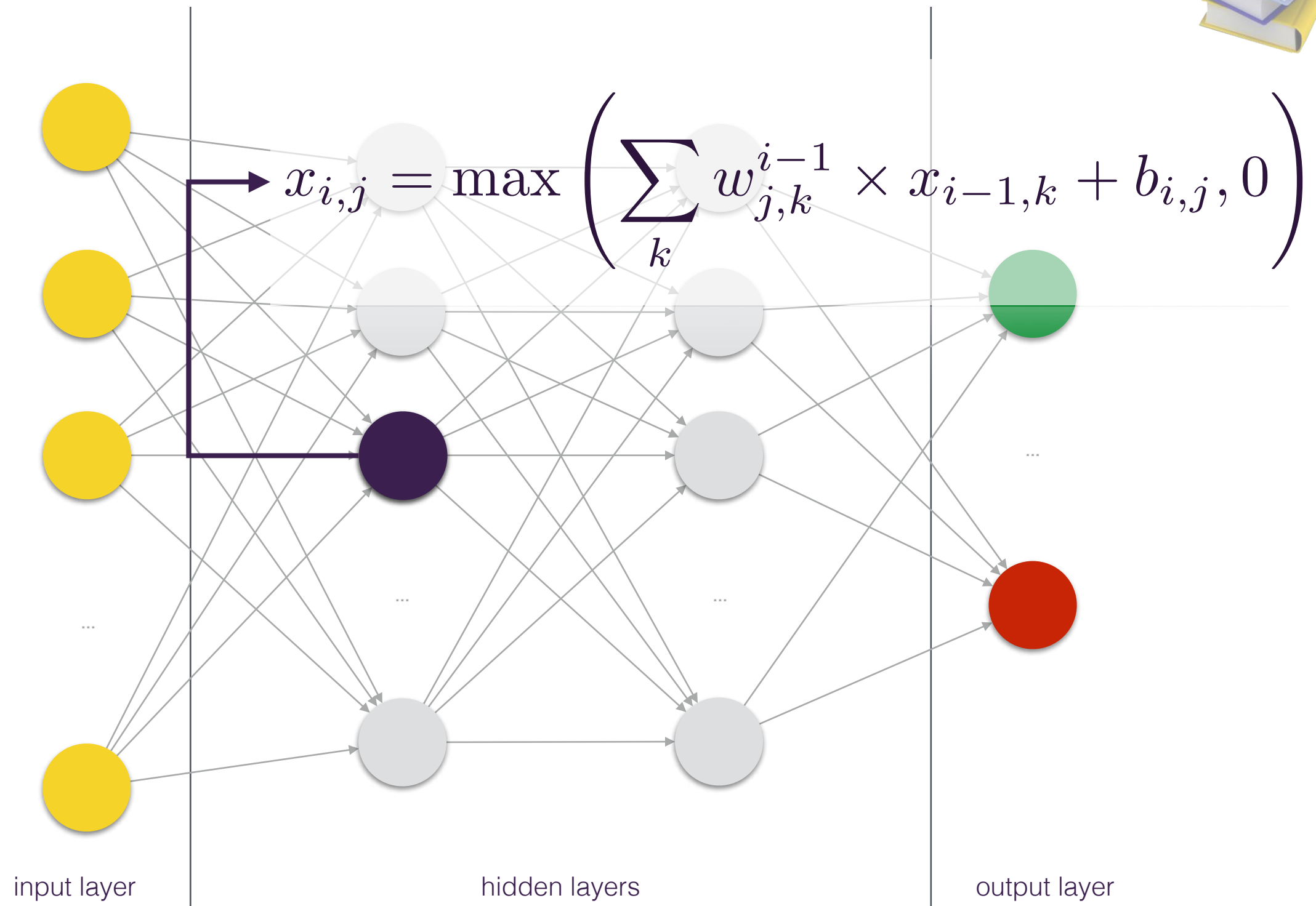
mathematical models

of the program behavior

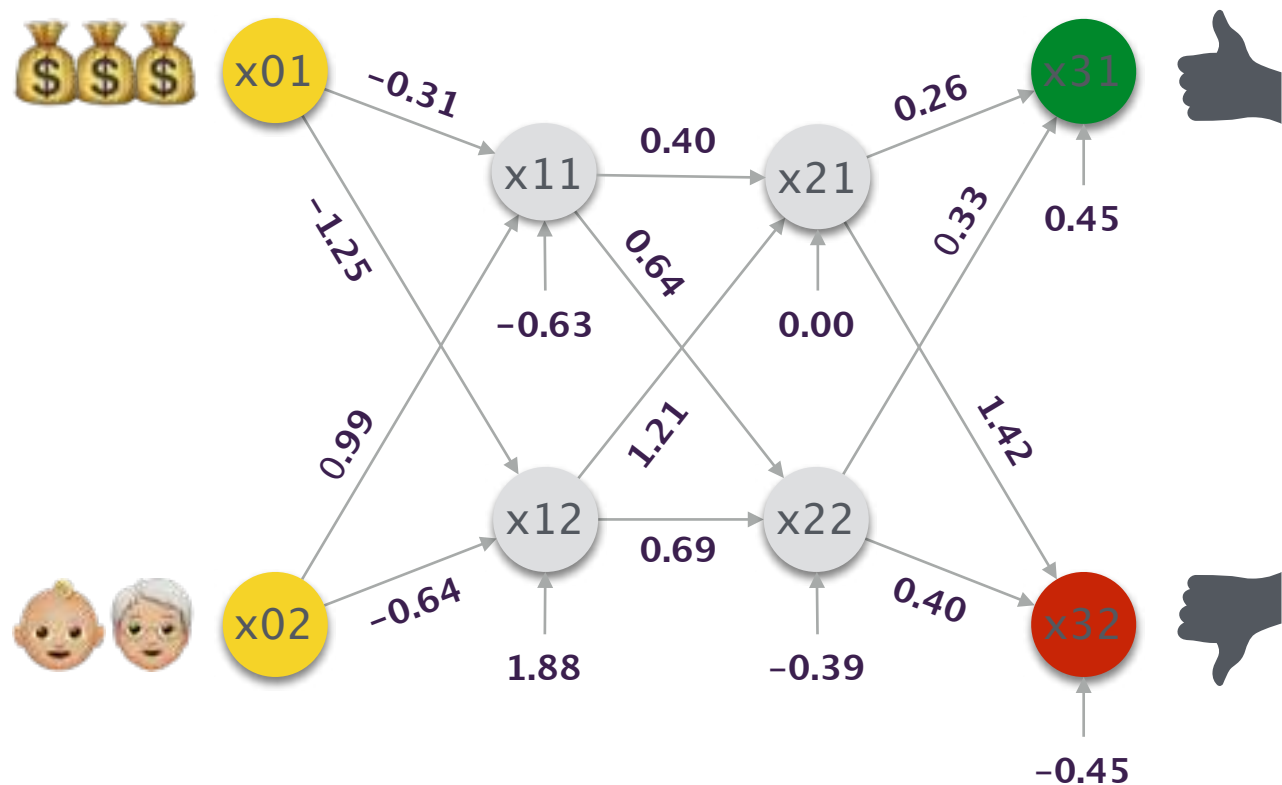


Feed-Forward Neural Networks

with ReLU Activations



Toy Example



```
x01 = input()
x02 = input()
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

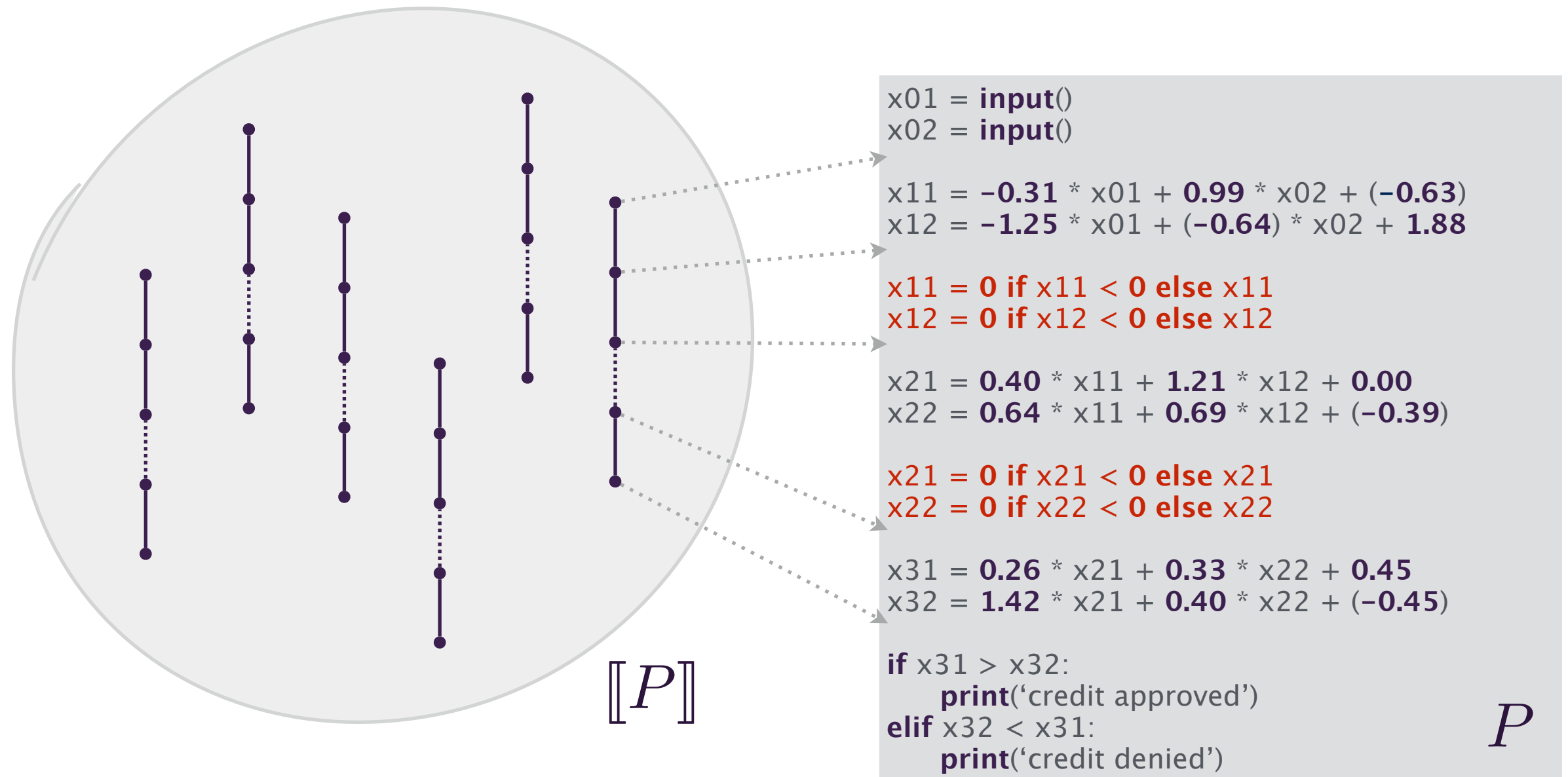
```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Trace Semantics

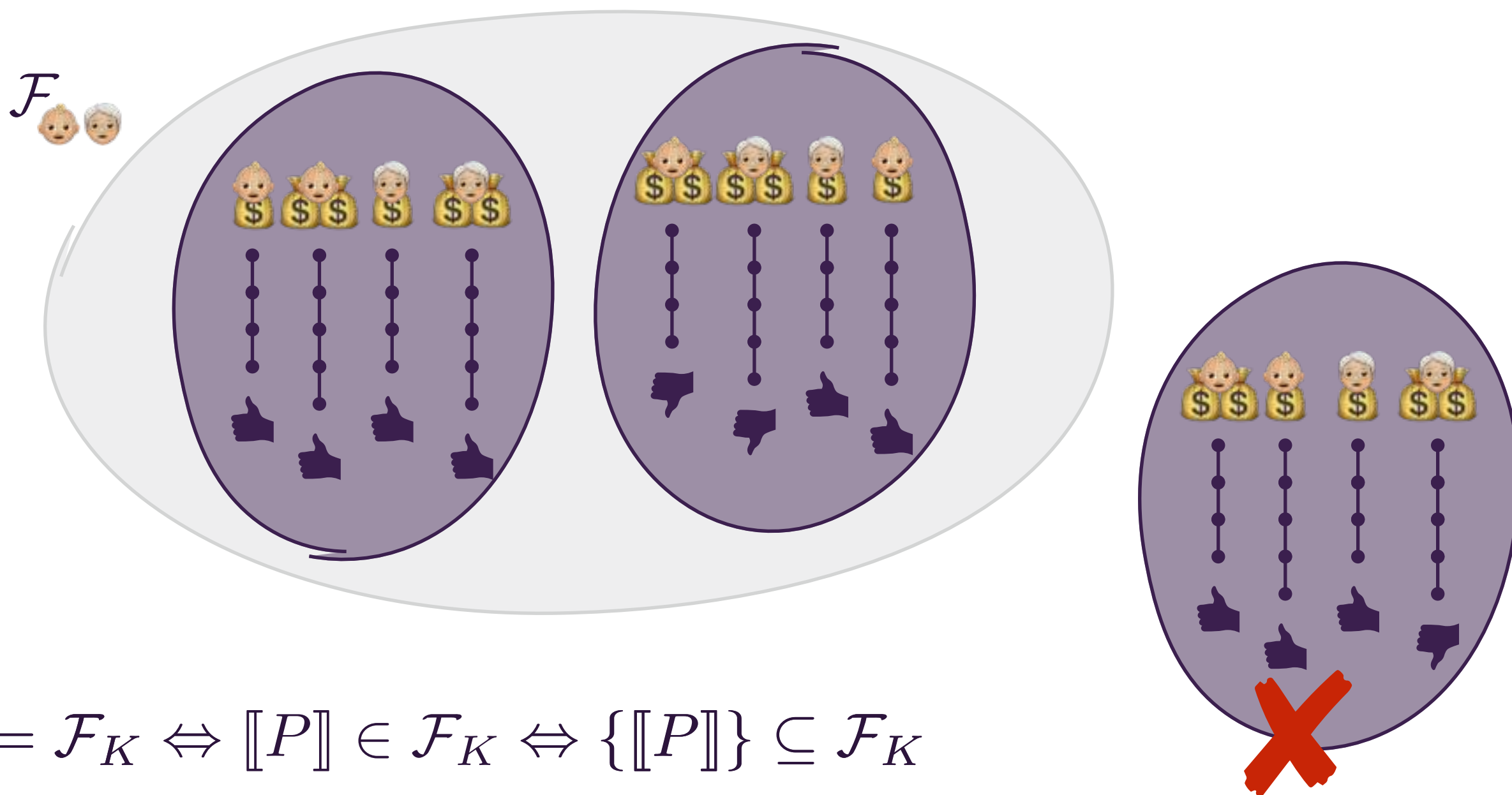




Causal Fairness

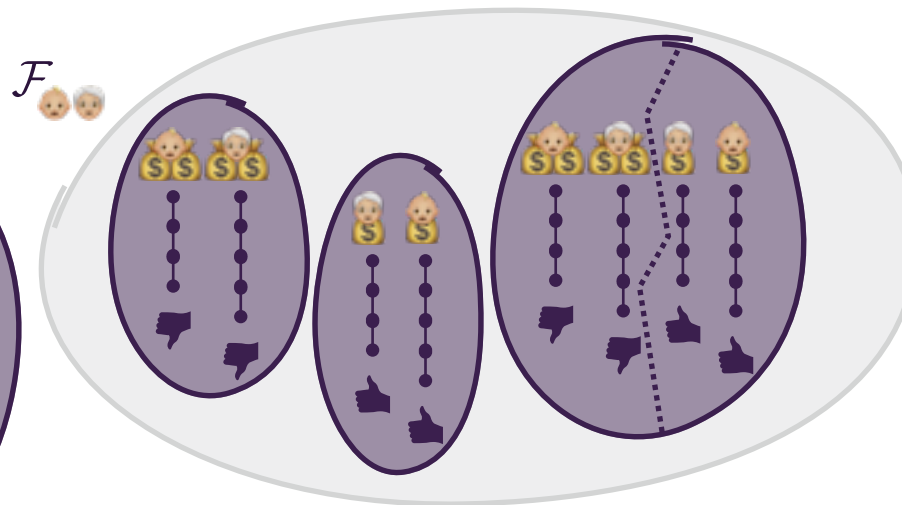
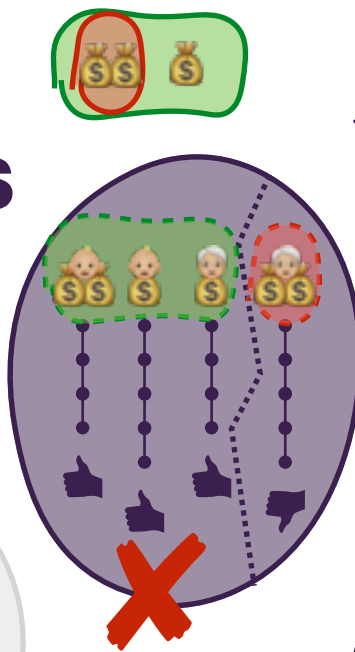
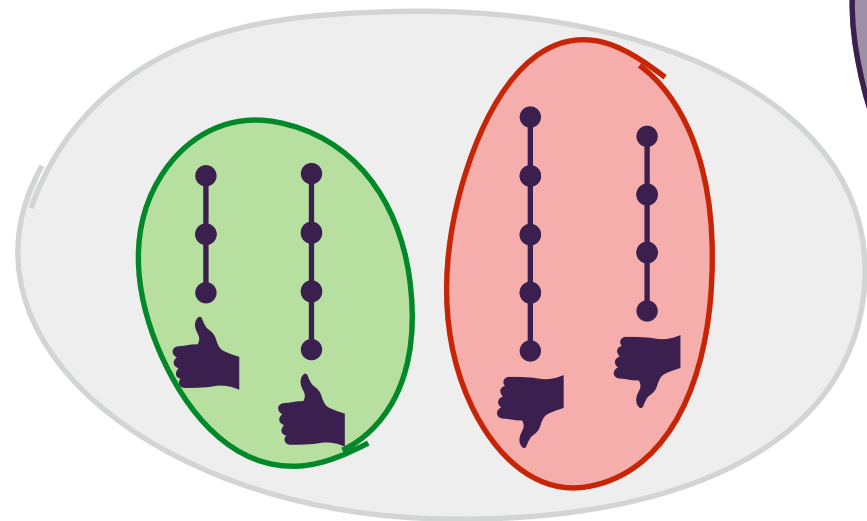
the classification is independent from the sensitive inputs

$$\mathcal{F}_K \stackrel{\text{def}}{=} \{ \llbracket P \rrbracket \mid \text{UNUSED}_K(\llbracket P \rrbracket) \}$$



$$P \models \mathcal{F}_K \Leftrightarrow \llbracket P \rrbracket \in \mathcal{F}_K \Leftrightarrow \{ \llbracket P \rrbracket \} \subseteq \mathcal{F}_K$$

Outcome Semantics



$$P \models \mathcal{F}_K \Leftrightarrow \llbracket P \rrbracket_{\bullet} \subseteq \mathcal{F}_K$$

$$\Leftrightarrow \forall S_1, S_2 \in \llbracket P \rrbracket_{\bullet}: S_1[0]|_K \cap S_2[0]|_K = \emptyset$$

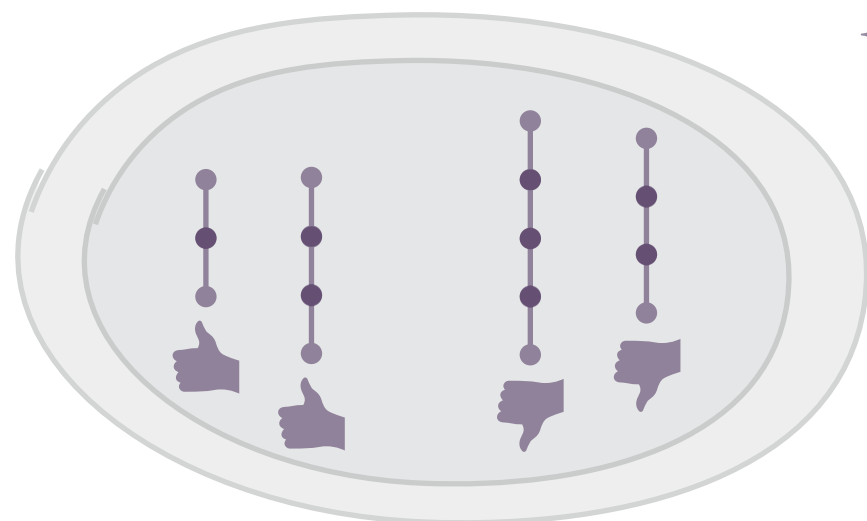
$\llbracket P \rrbracket_{\bullet}$

α_{\bullet}

$\{\llbracket P \rrbracket\}$



partition executions based on their outcome



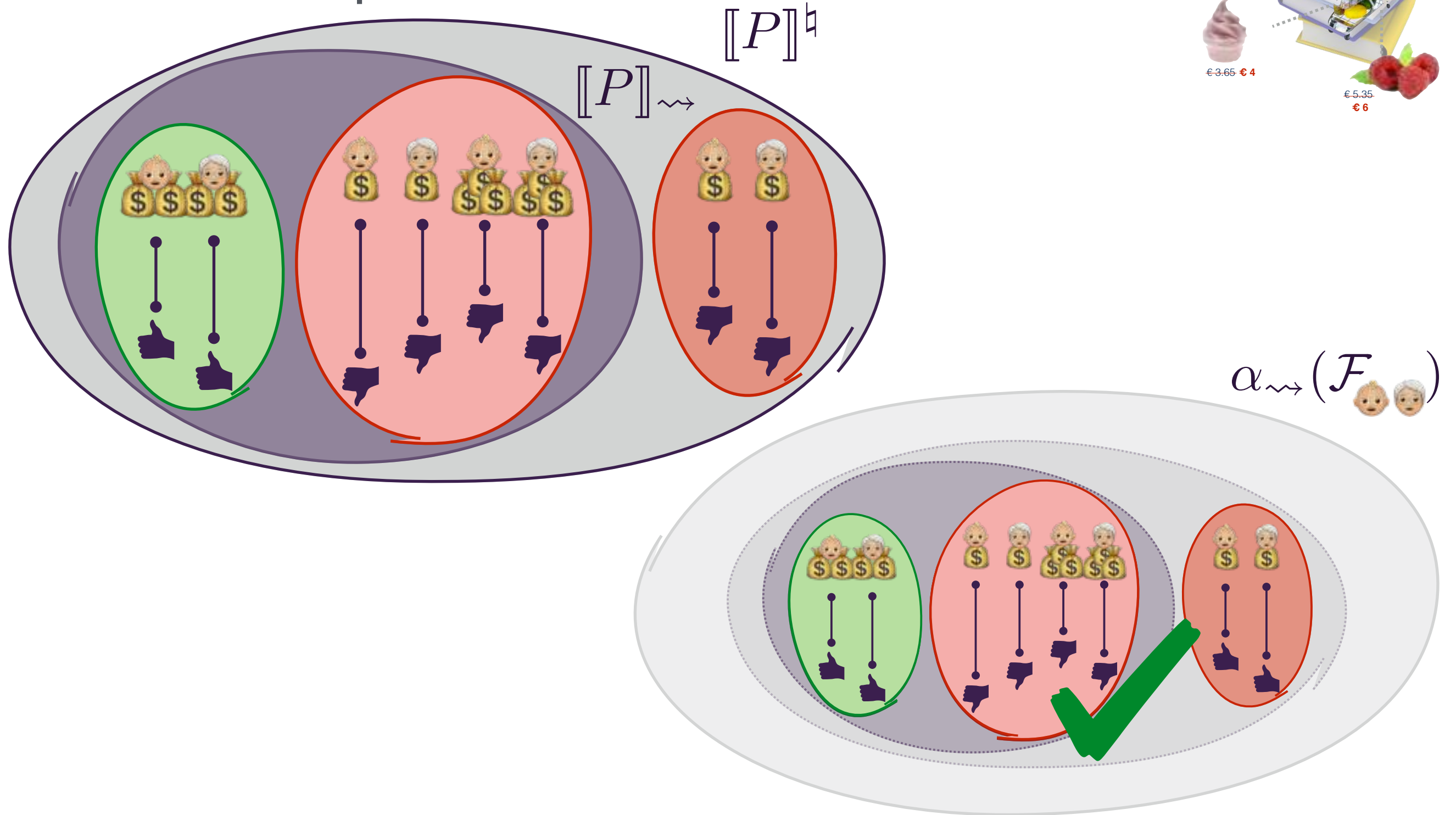
$$P \models \mathcal{F}_K \Leftrightarrow \{\llbracket P \rrbracket\} \subseteq \mathcal{F}_K$$

Sounds and Complete Causal Fairness Certification



Abstract Semantics

Sounds and Complete Causal Fairness Certification



$$P \models \mathcal{F}_K \Leftrightarrow \llbracket P \rrbracket_{\rightsquigarrow} \subseteq \alpha_{\rightsquigarrow}(\mathcal{F}_K) \Leftarrow \llbracket P \rrbracket^\sharp \subseteq \alpha_{\rightsquigarrow}(\mathcal{F}_K) \\ \Leftarrow \forall S_1, S_2 \in \llbracket P \rrbracket^\sharp: (S_1[\omega] \neq S_2[\omega] \Rightarrow S_1[0]|_K \cap S_2[0]|_K = \emptyset)$$

Causal Fairness Analysis

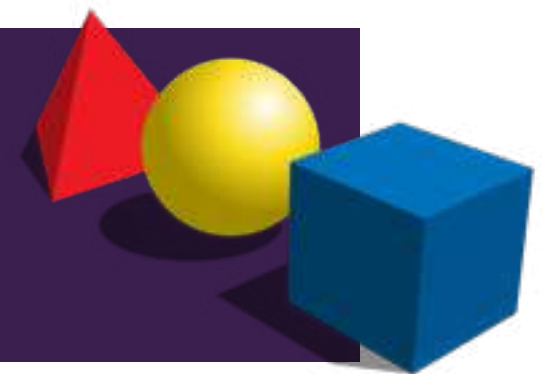
practical tools

targeting specific programs



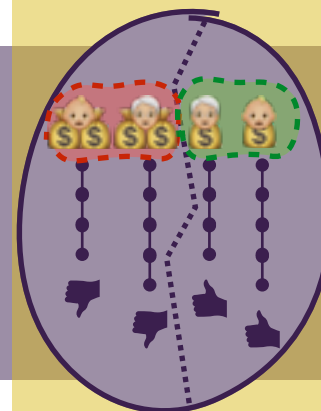
algorithmic approaches

to decide program properties

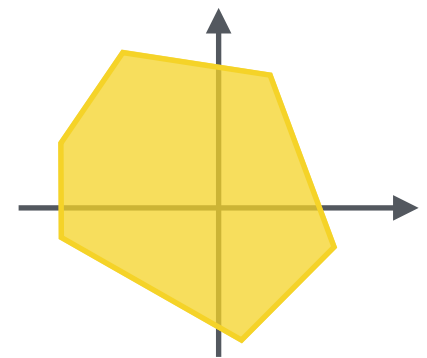
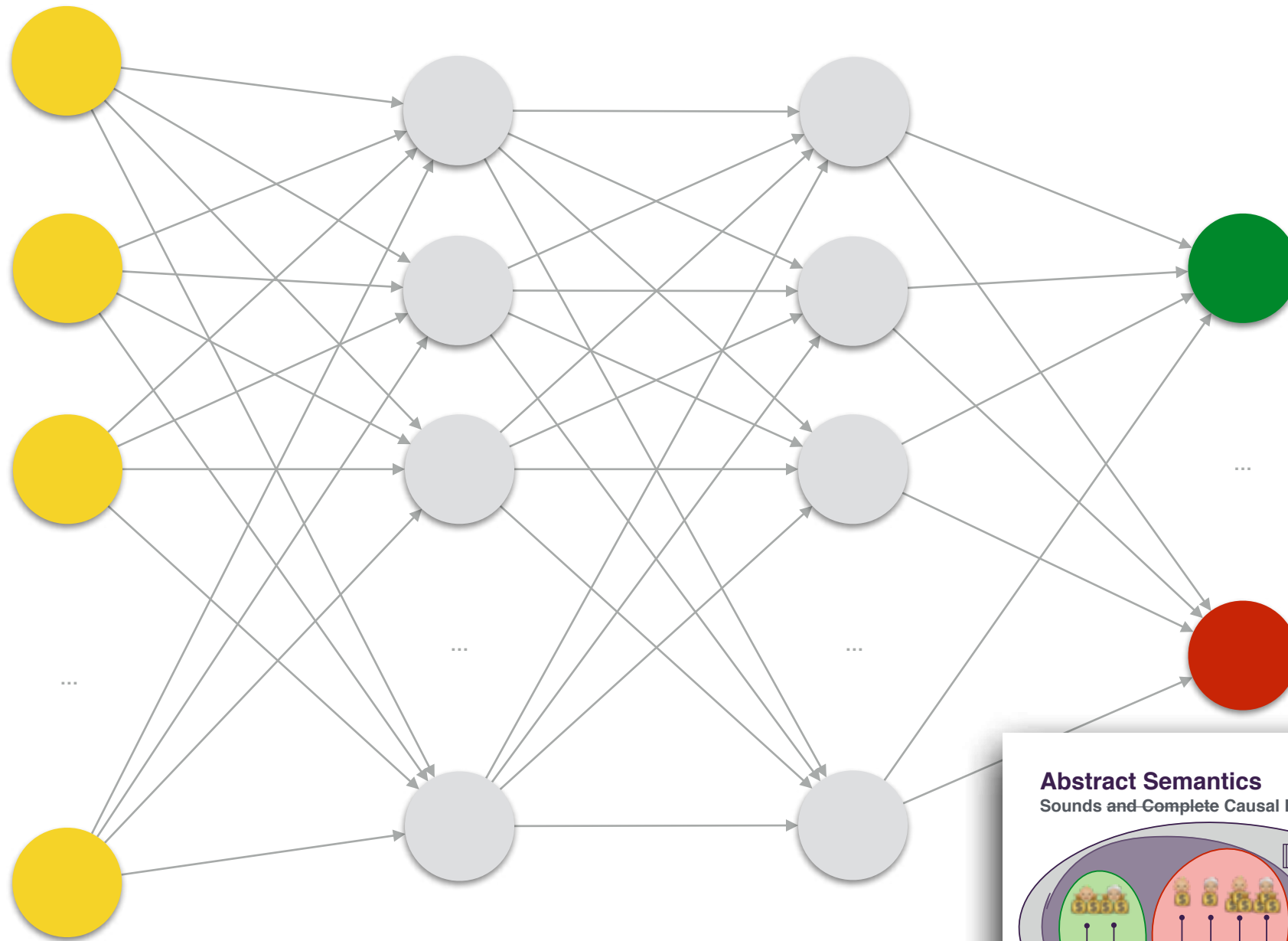
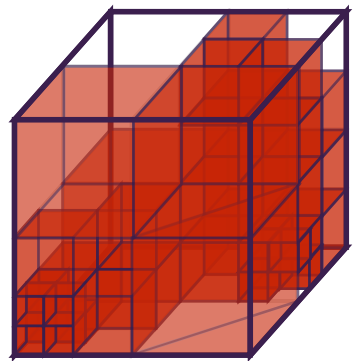
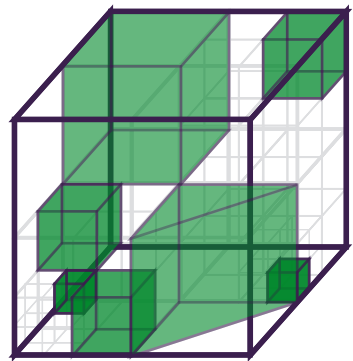
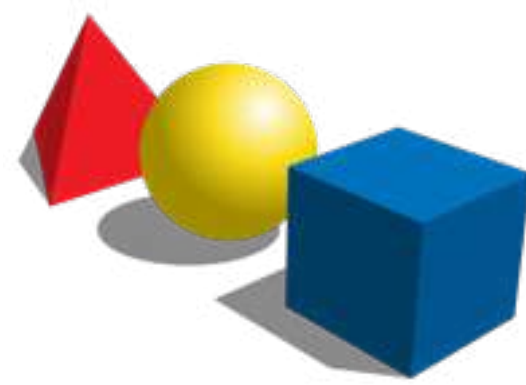


mathematical models

of the program behavior

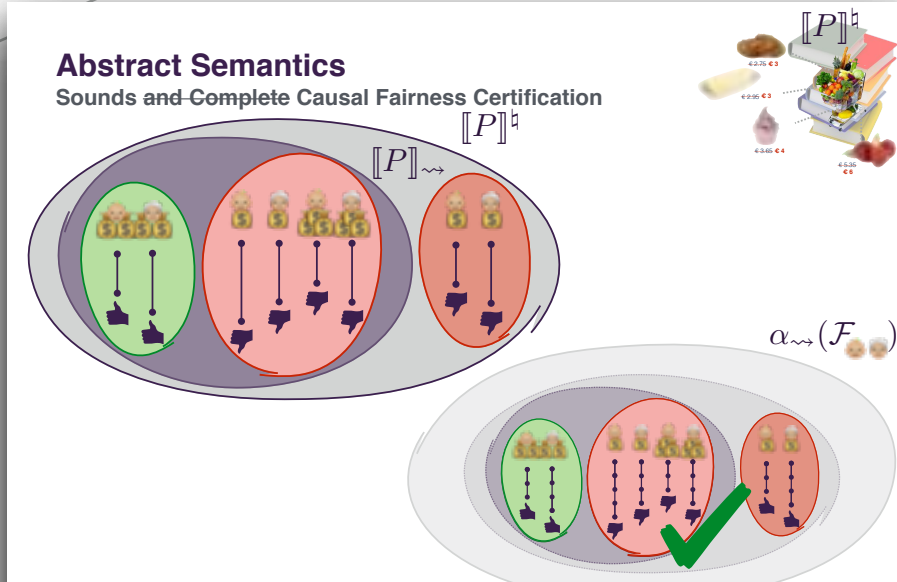


Naïve Backward Analysis



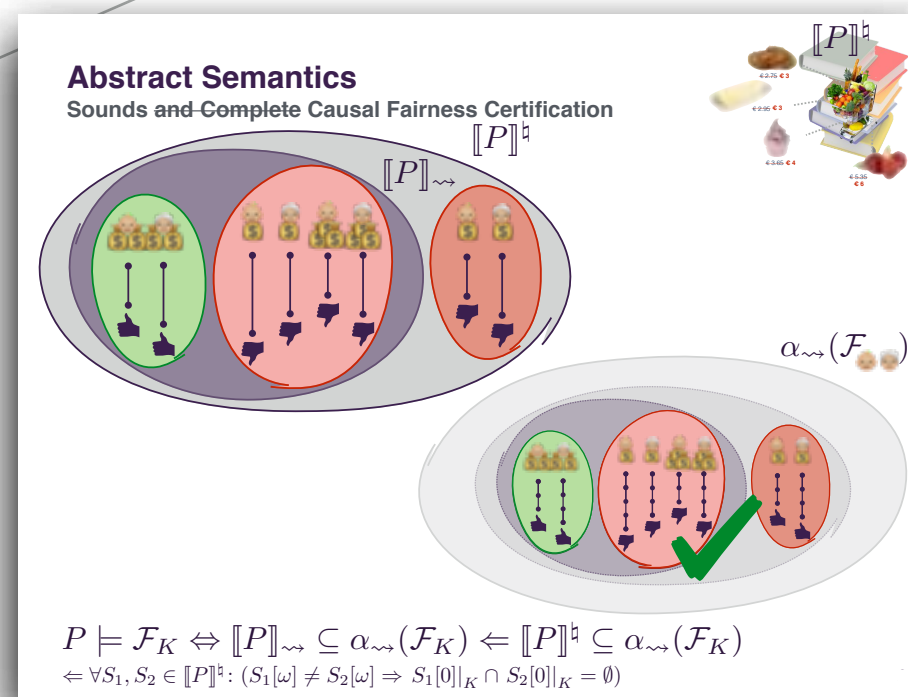
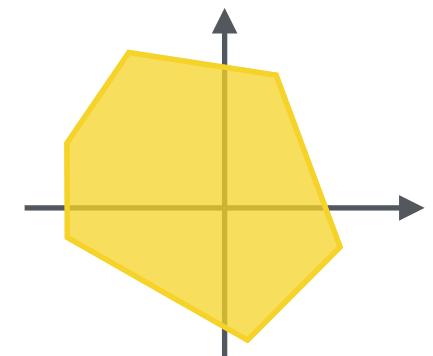
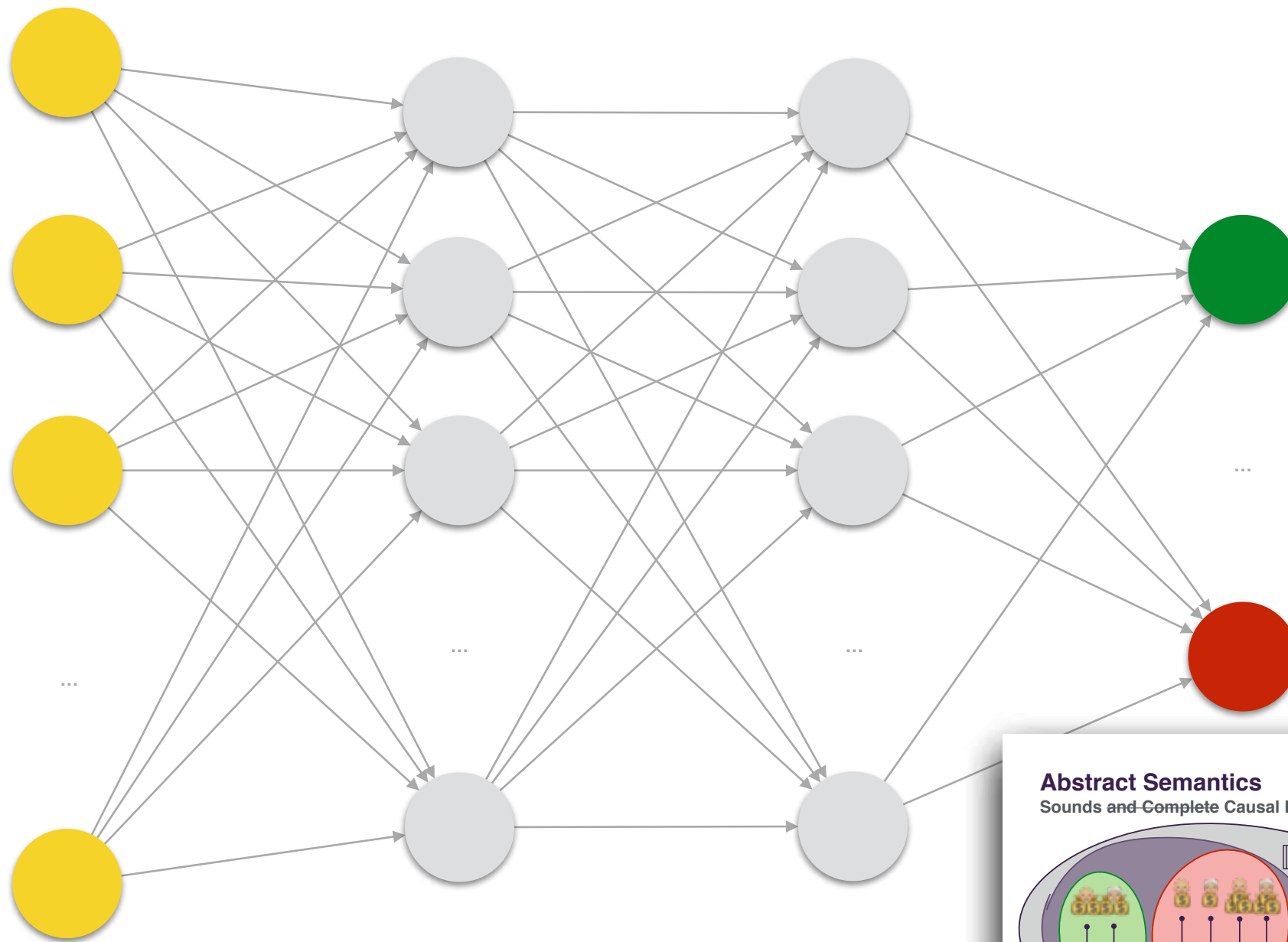
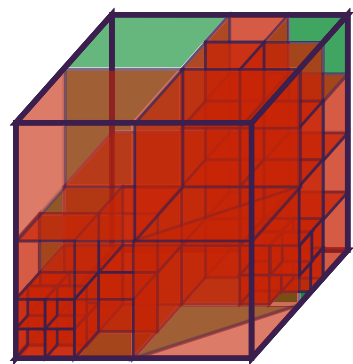
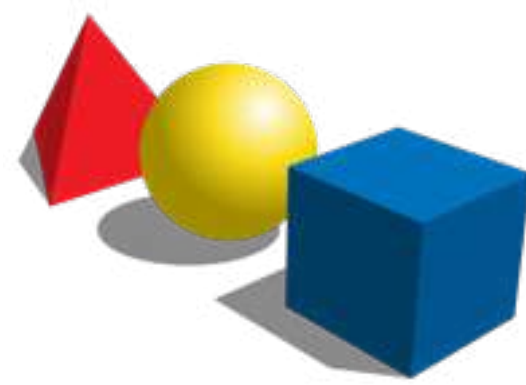
Abstract Semantics

Sounds and Complete Causal Fairness Certification



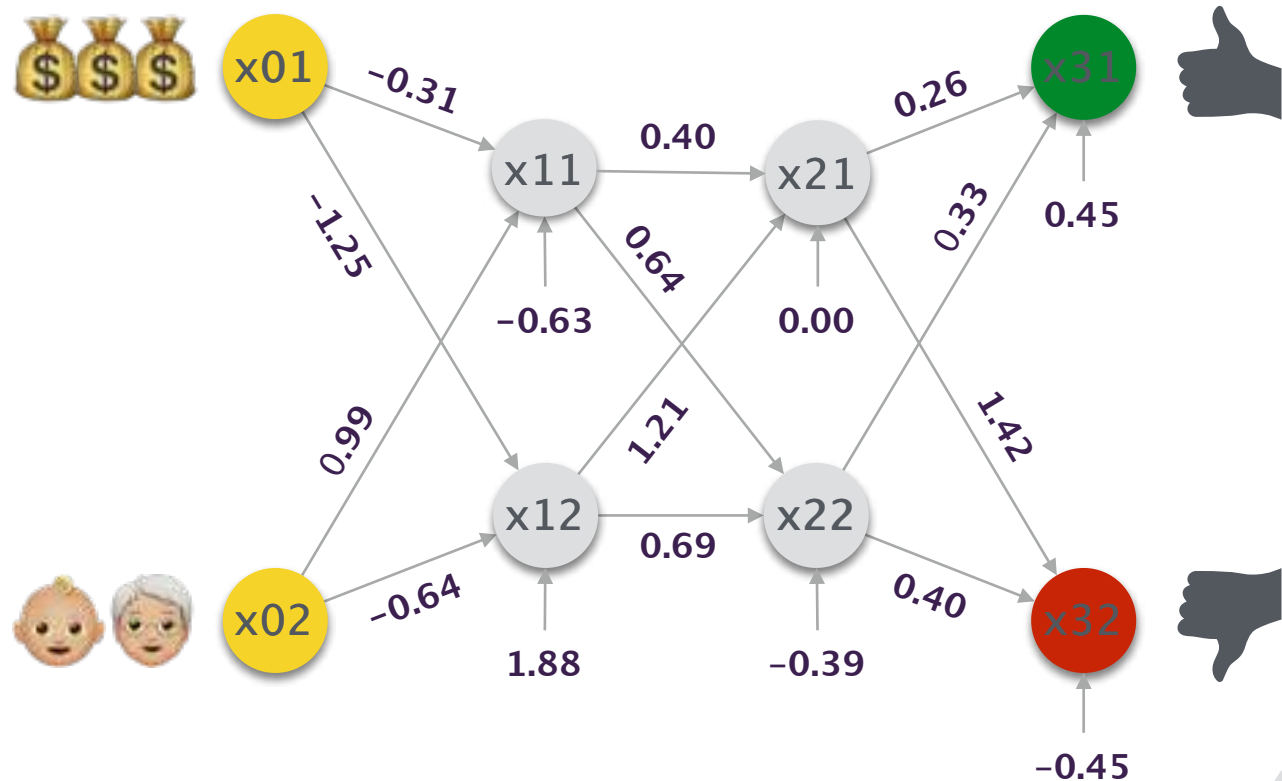
$$P \models \mathcal{F}_K \Leftrightarrow \llbracket P \rrbracket_{\sim} \subseteq \alpha_{\sim}(\mathcal{F}_K) \Leftarrow \llbracket P \rrbracket^{\sharp} \subseteq \alpha_{\sim}(\mathcal{F}_K) \\ \Leftarrow \forall S_1, S_2 \in \llbracket P \rrbracket^{\sharp}: (S_1[\omega] \neq S_2[\omega] \Rightarrow S_1[0]_K \cap S_2[0]_K = \emptyset)$$

Naïve Backward Analysis



Toy Example

Naïve Backward Analysis



too many disjunctions!



```
x01 = float(input())
x02 = float(input())
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
1.16 * x21 + 0.07 * x22 < 0.90
1.16 * x21 + 0.07 * x22 > 0.90
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
x31 > x32
x32 > x31
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

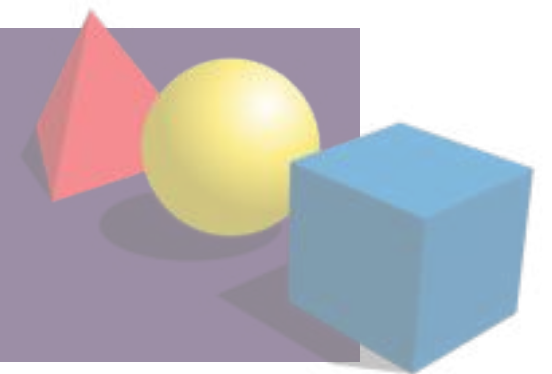
practical tools

targeting specific programs



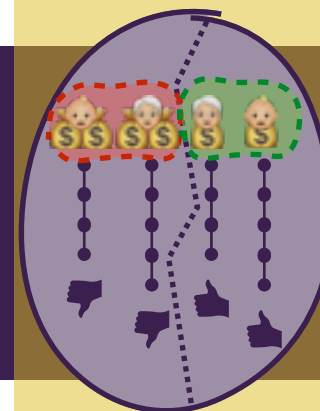
algorithmic approaches

to decide program properties

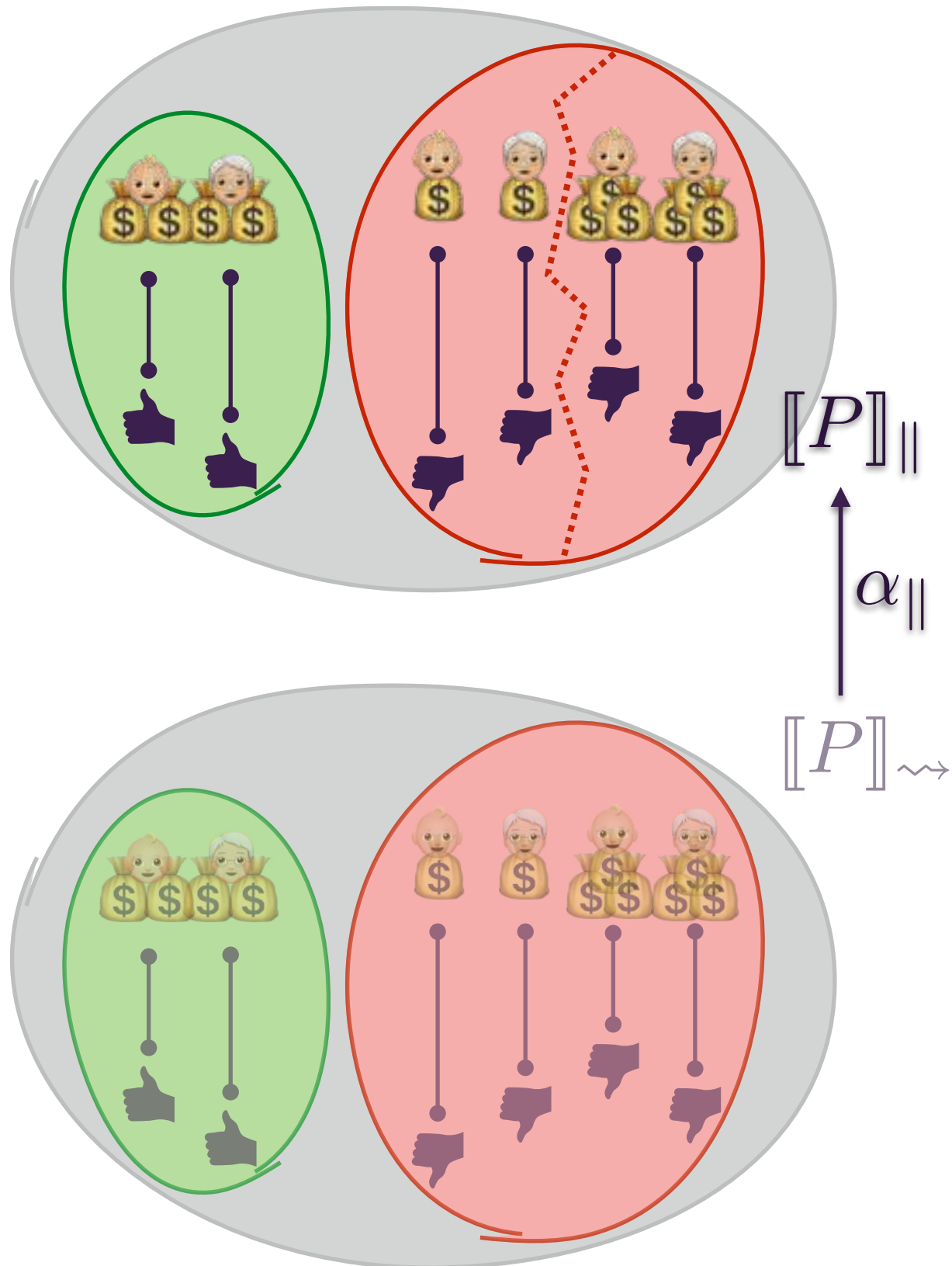


mathematical models

of the program behavior



Parallel Semantics



$$P \models \mathcal{F}_K \Leftrightarrow \llbracket P \rrbracket_{\parallel} \subseteq \alpha_{\rightsquigarrow}(\mathcal{F}_K)$$

$$\Leftrightarrow \forall S_1, S_2 \in \llbracket P \rrbracket_{\parallel} : (S_1[\omega] \neq S_2[\omega] \Rightarrow S_1[0]|_K \cap S_2[0]|_K = \emptyset)$$



partition with respect to non-sensitive inputs

$$P \models \mathcal{F}_K \Leftrightarrow \llbracket P \rrbracket_{\rightsquigarrow} \subseteq \alpha_{\rightsquigarrow}(\mathcal{F}_K)$$

$$\Leftrightarrow \forall S_1, S_2 \in \llbracket P \rrbracket_{\rightsquigarrow} : S_1[0]|_K \cap S_2[0]|_K = \emptyset$$

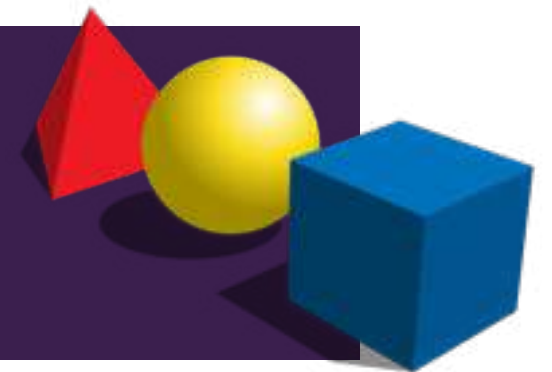
practical tools

targeting specific programs



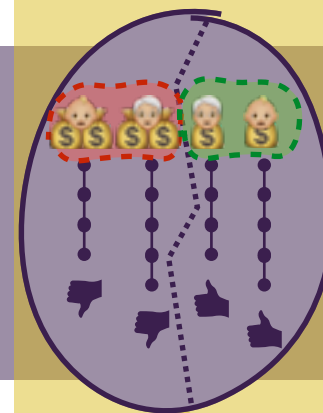
algorithmic approaches

to decide program properties

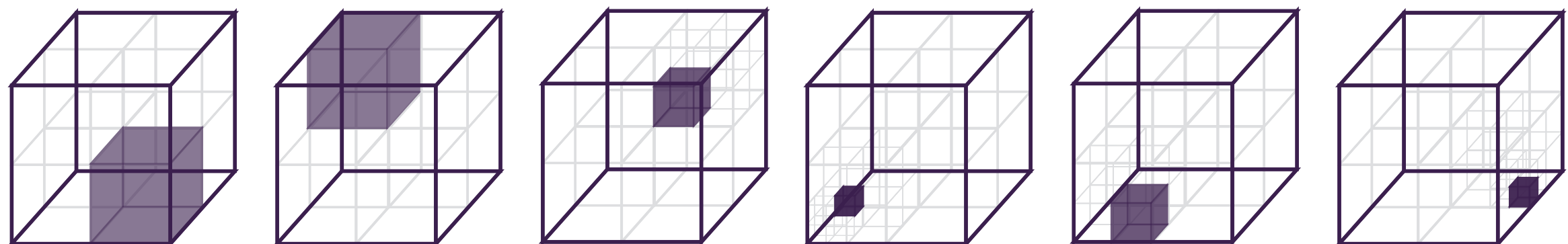
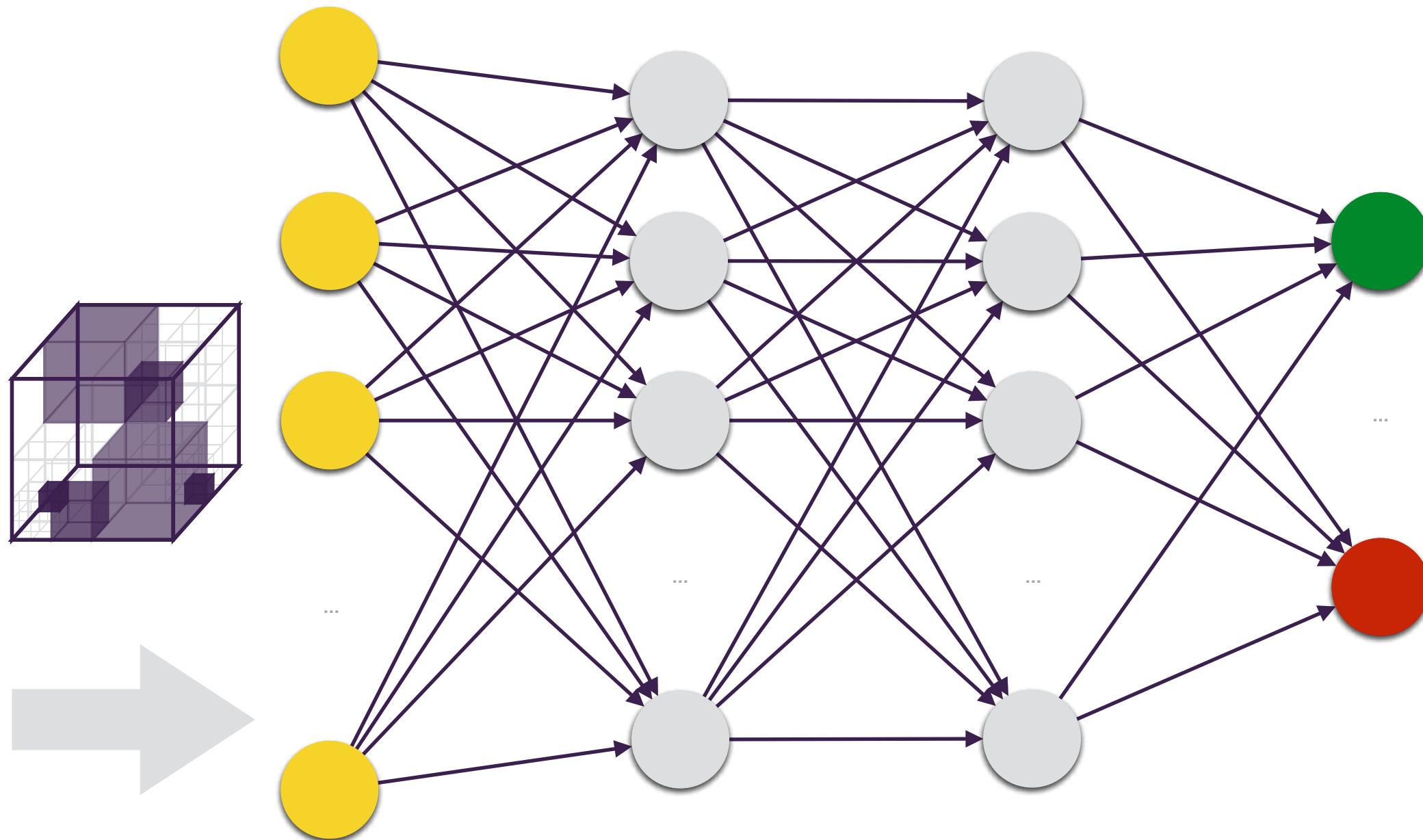
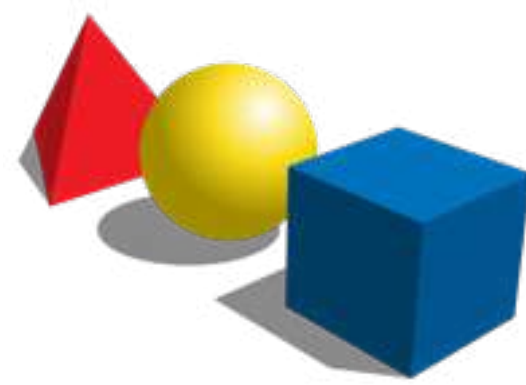


mathematical models

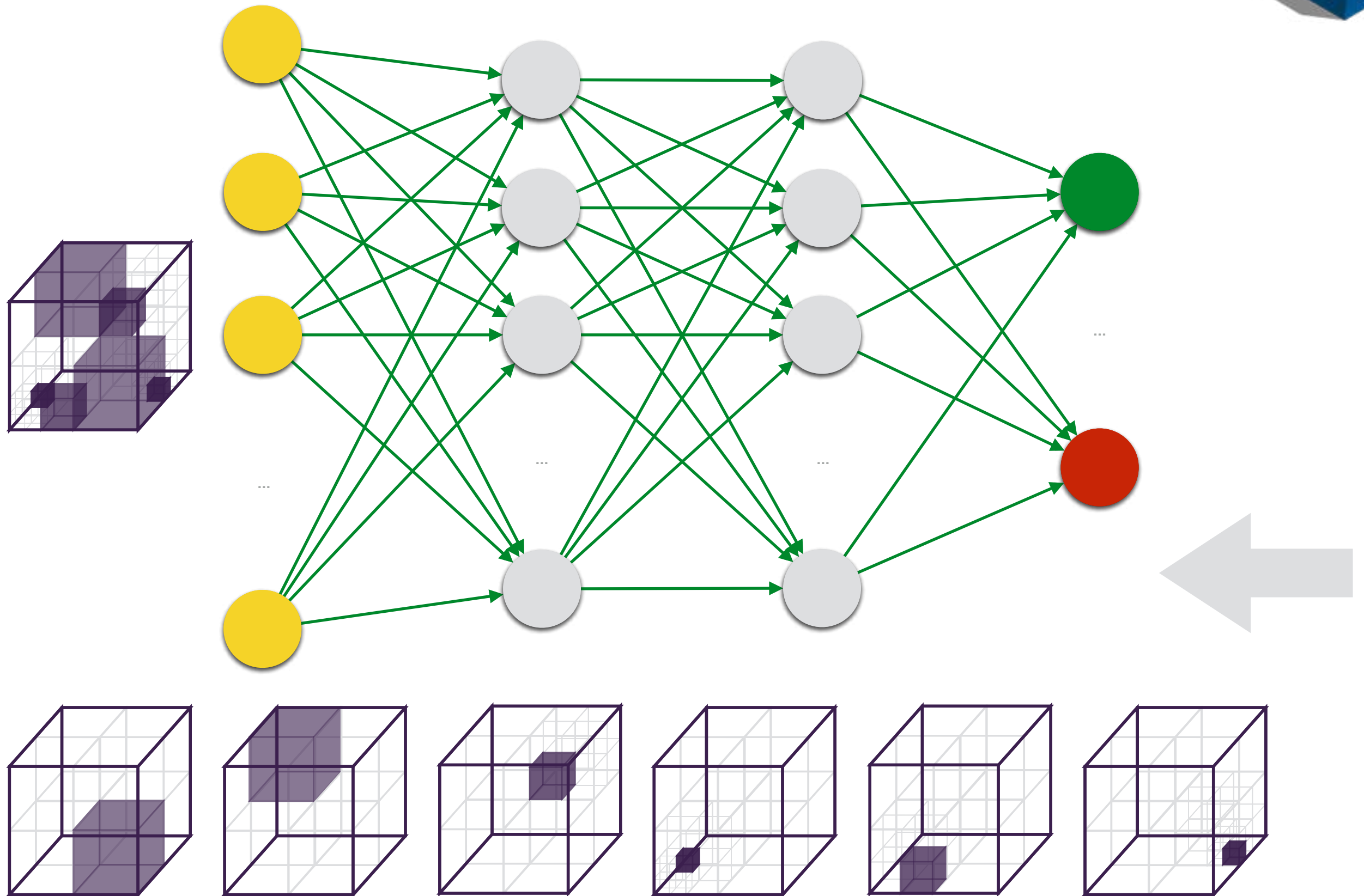
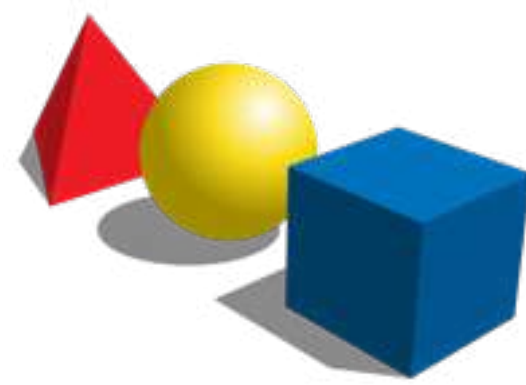
of the program behavior



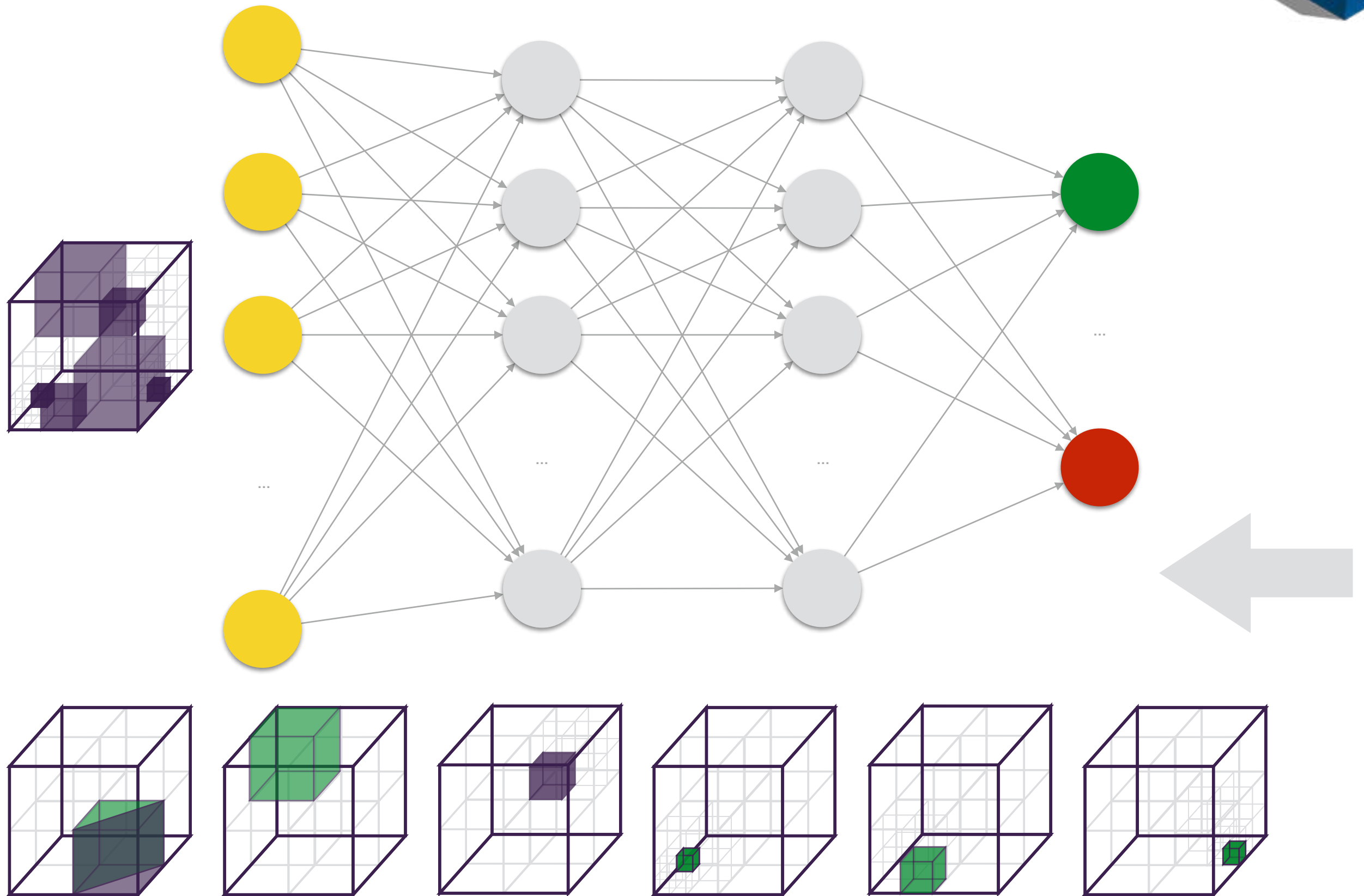
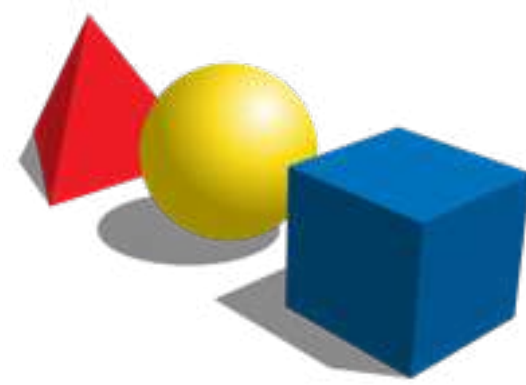
Forward and Backward Analysis



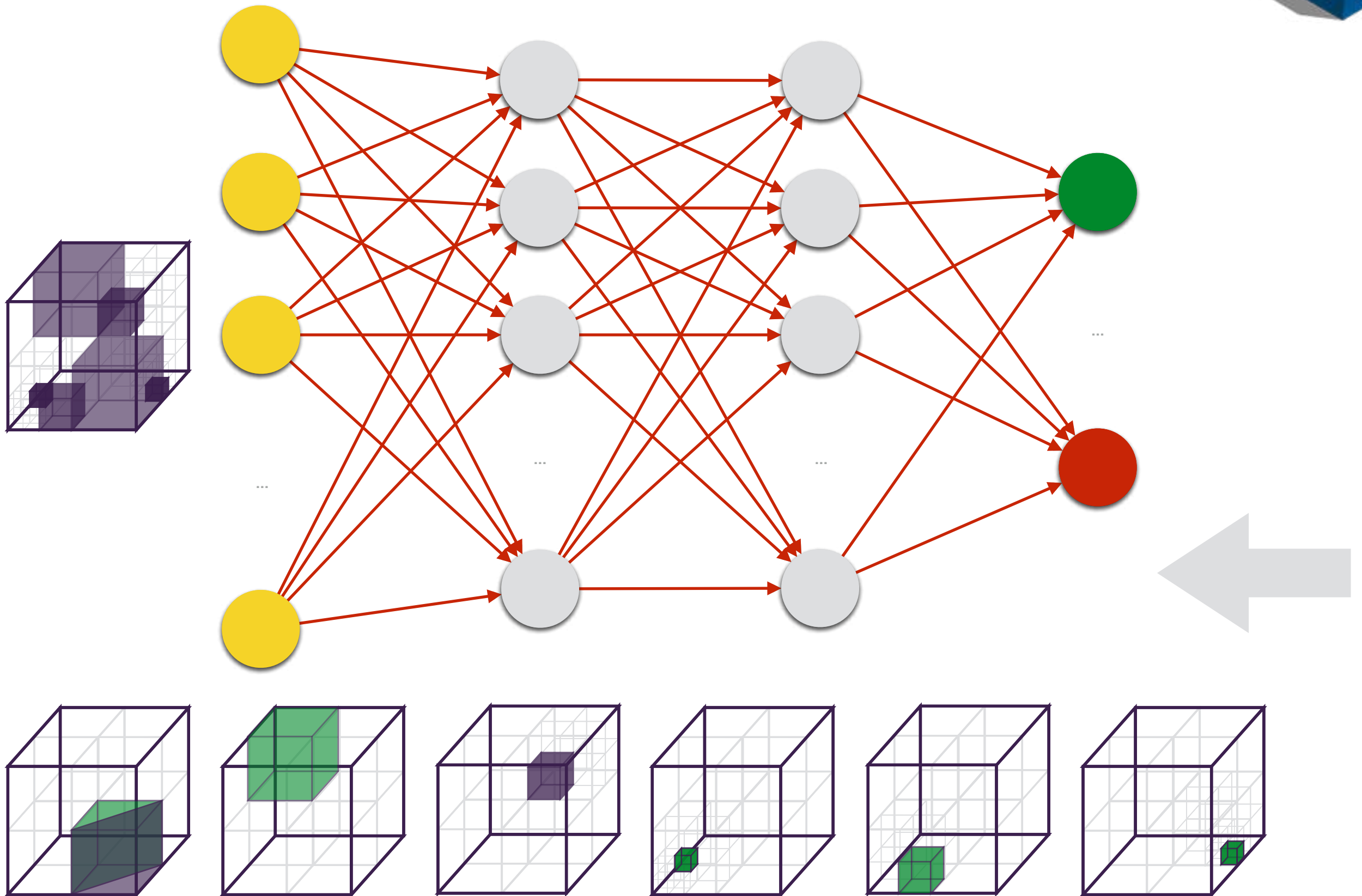
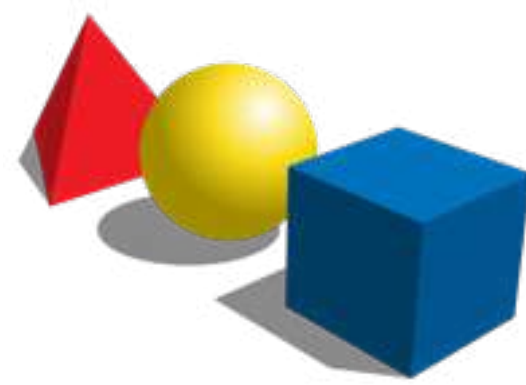
Forward and Backward Analysis



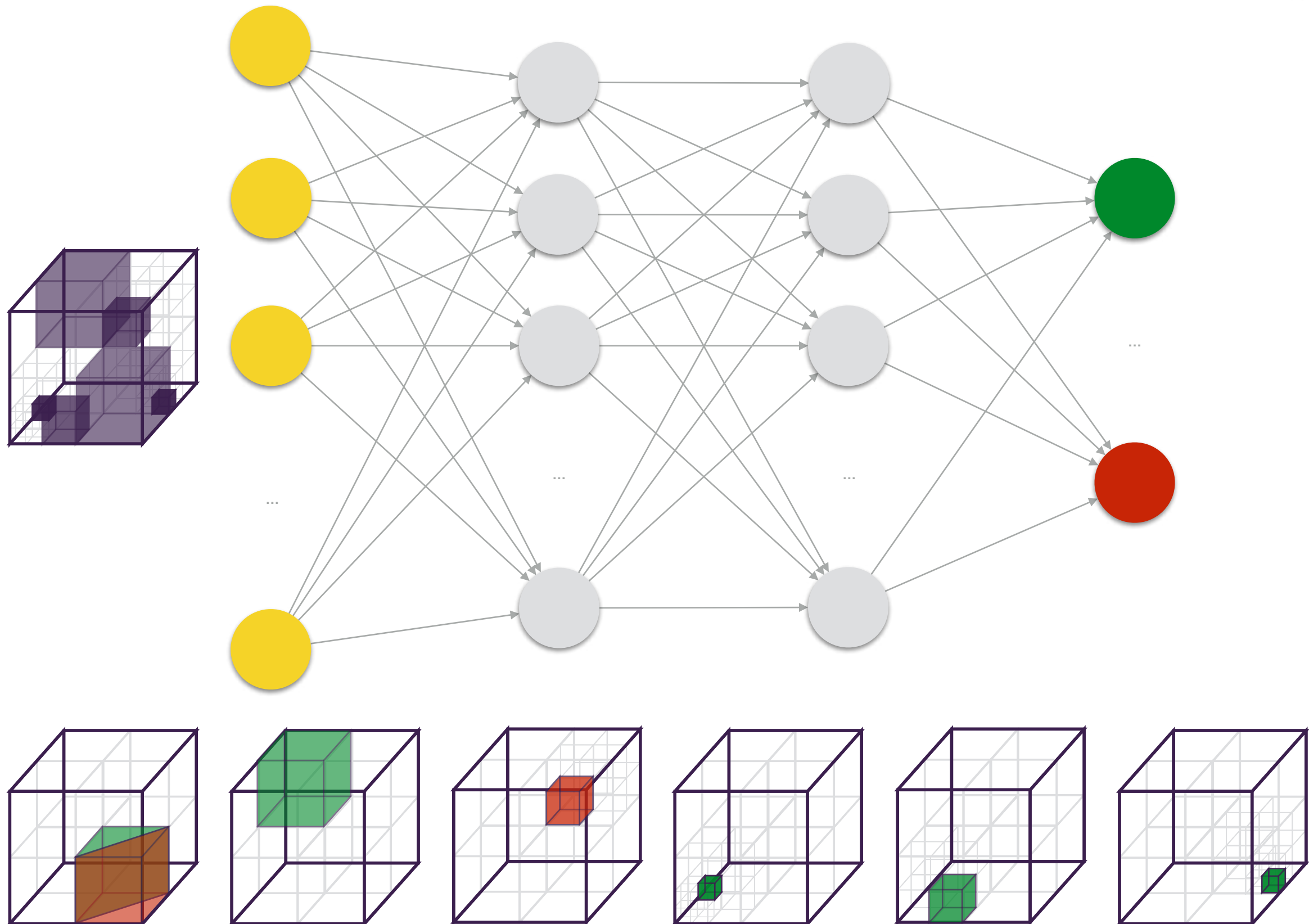
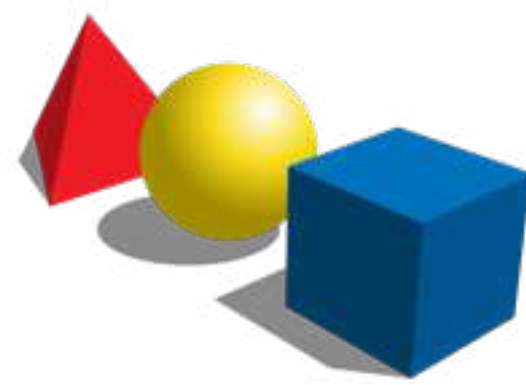
Forward and Backward Analysis



Forward and Backward Analysis

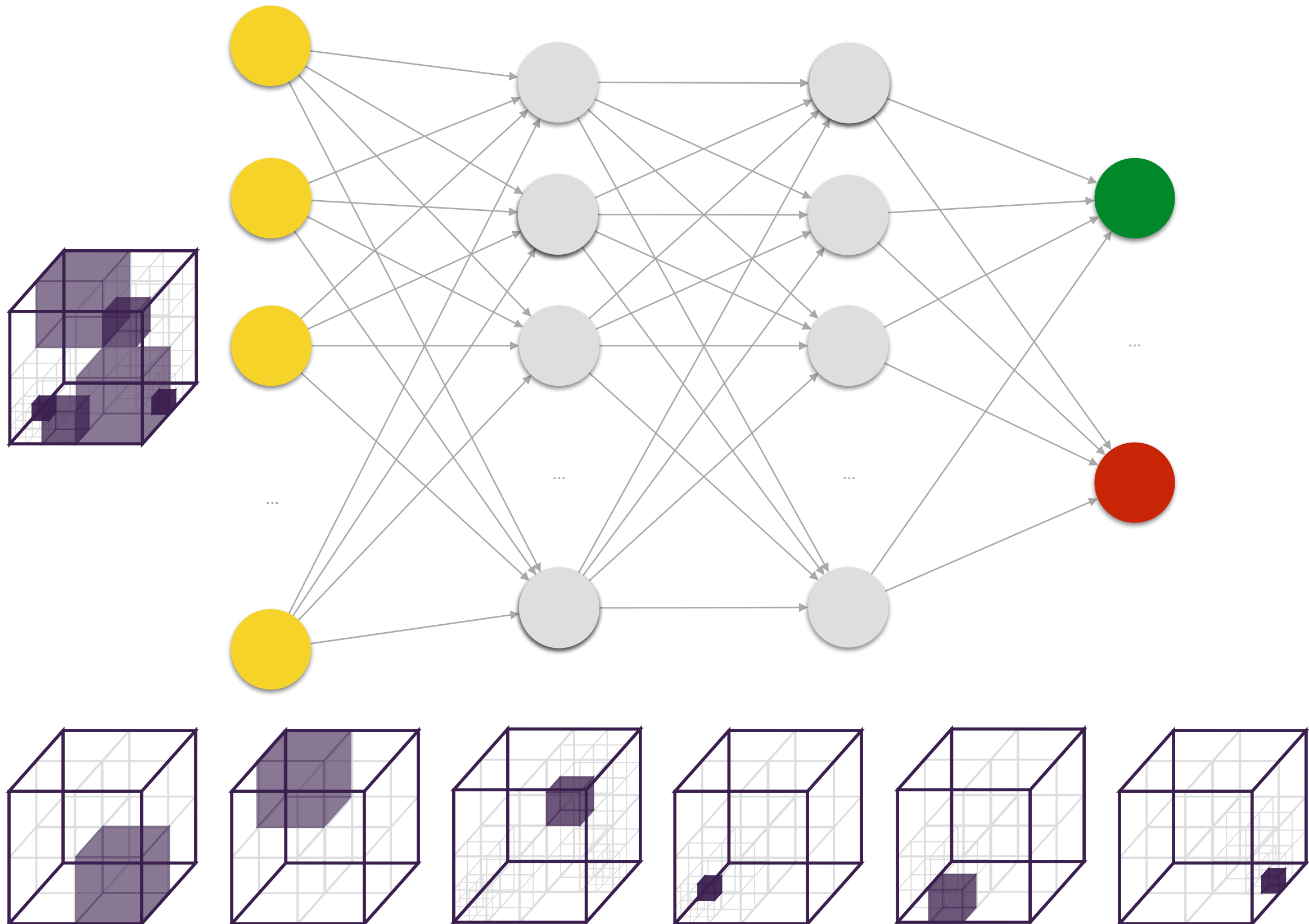
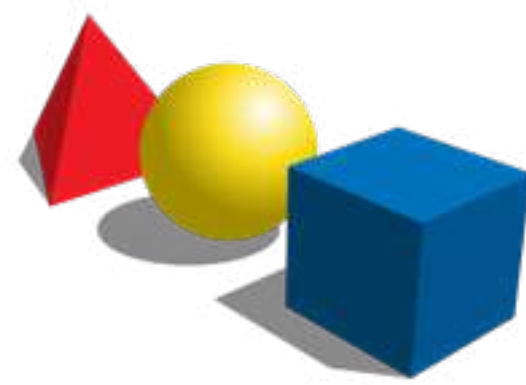


Forward and Backward Analysis



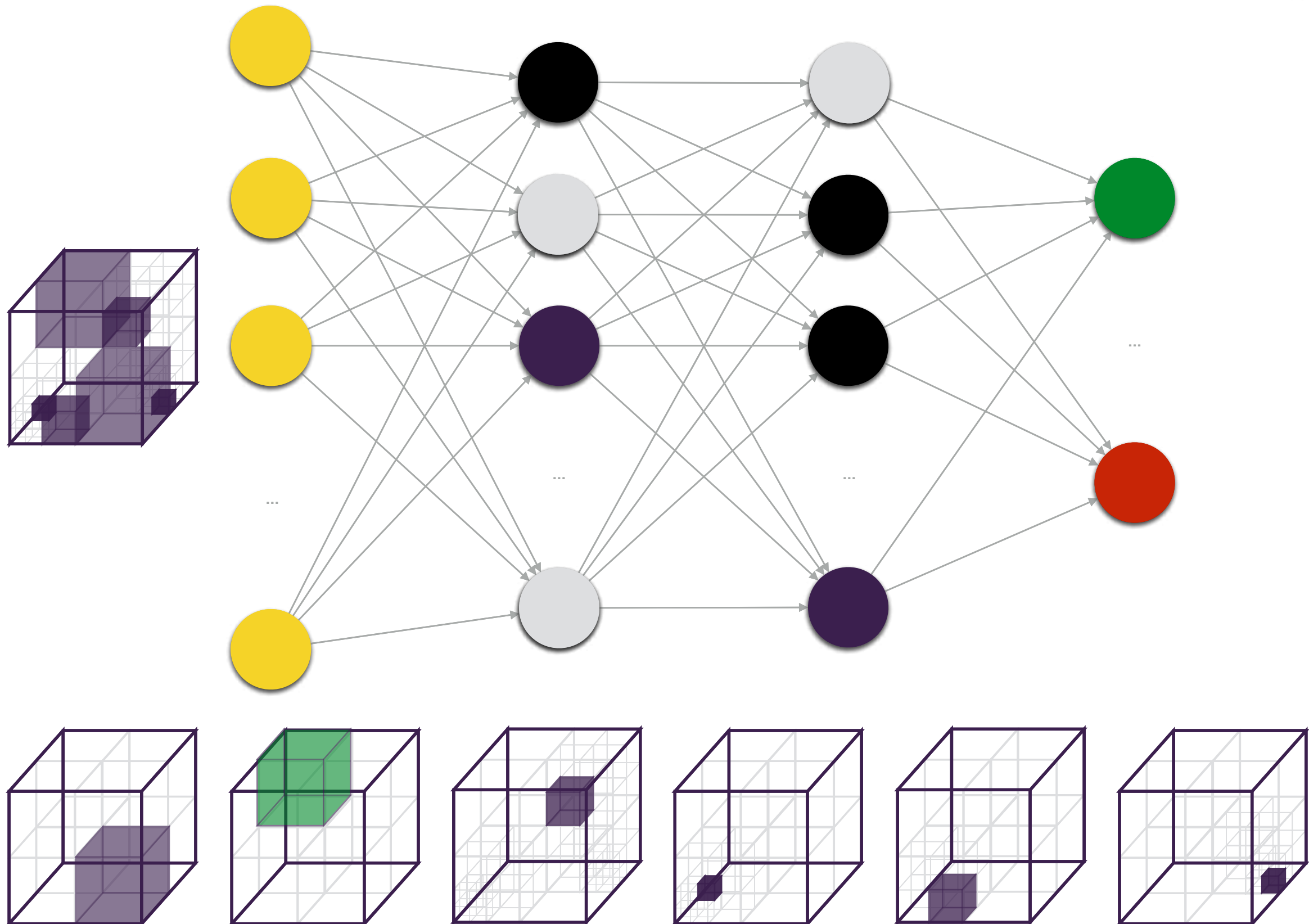
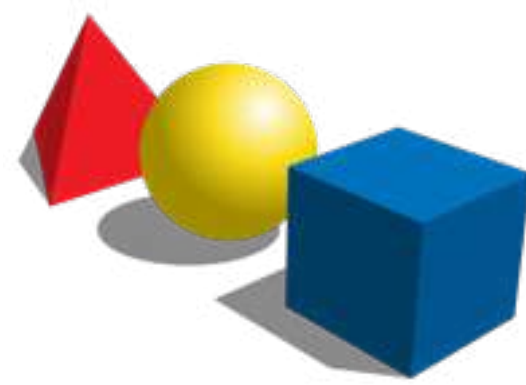
Forward and Backward Analysis

A Better Solution



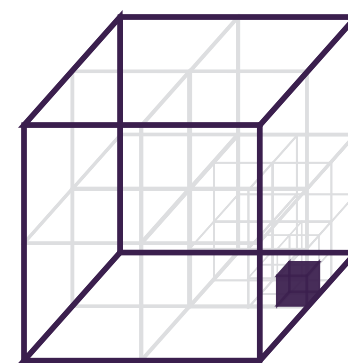
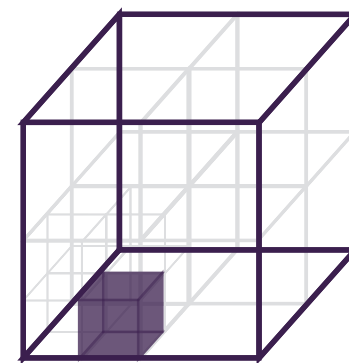
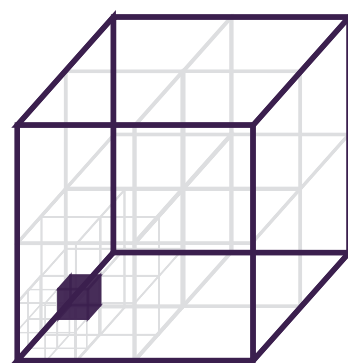
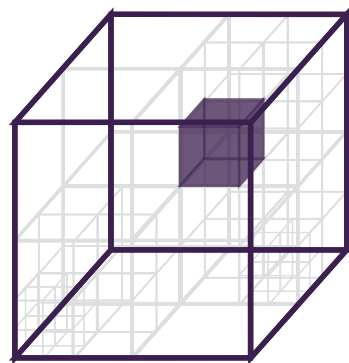
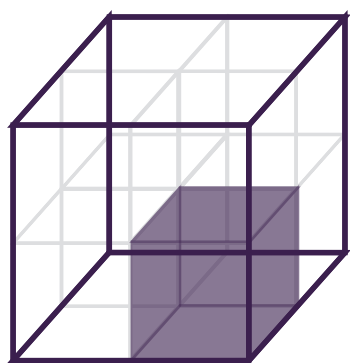
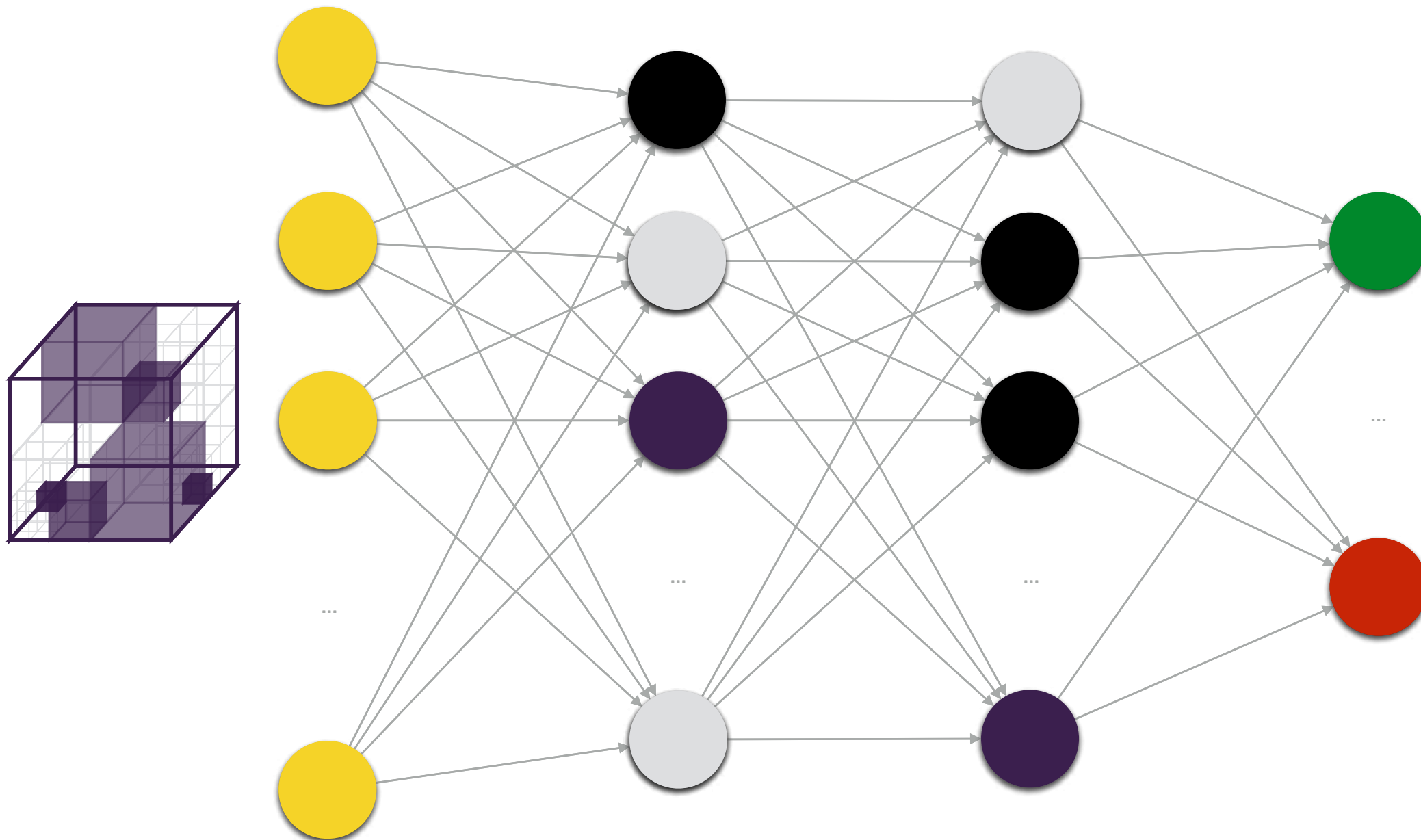
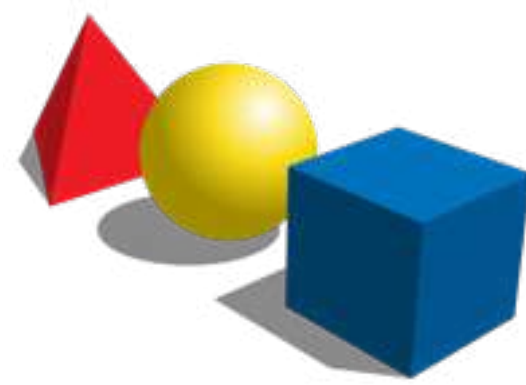
Forward and Backward Analysis

A Better Solution



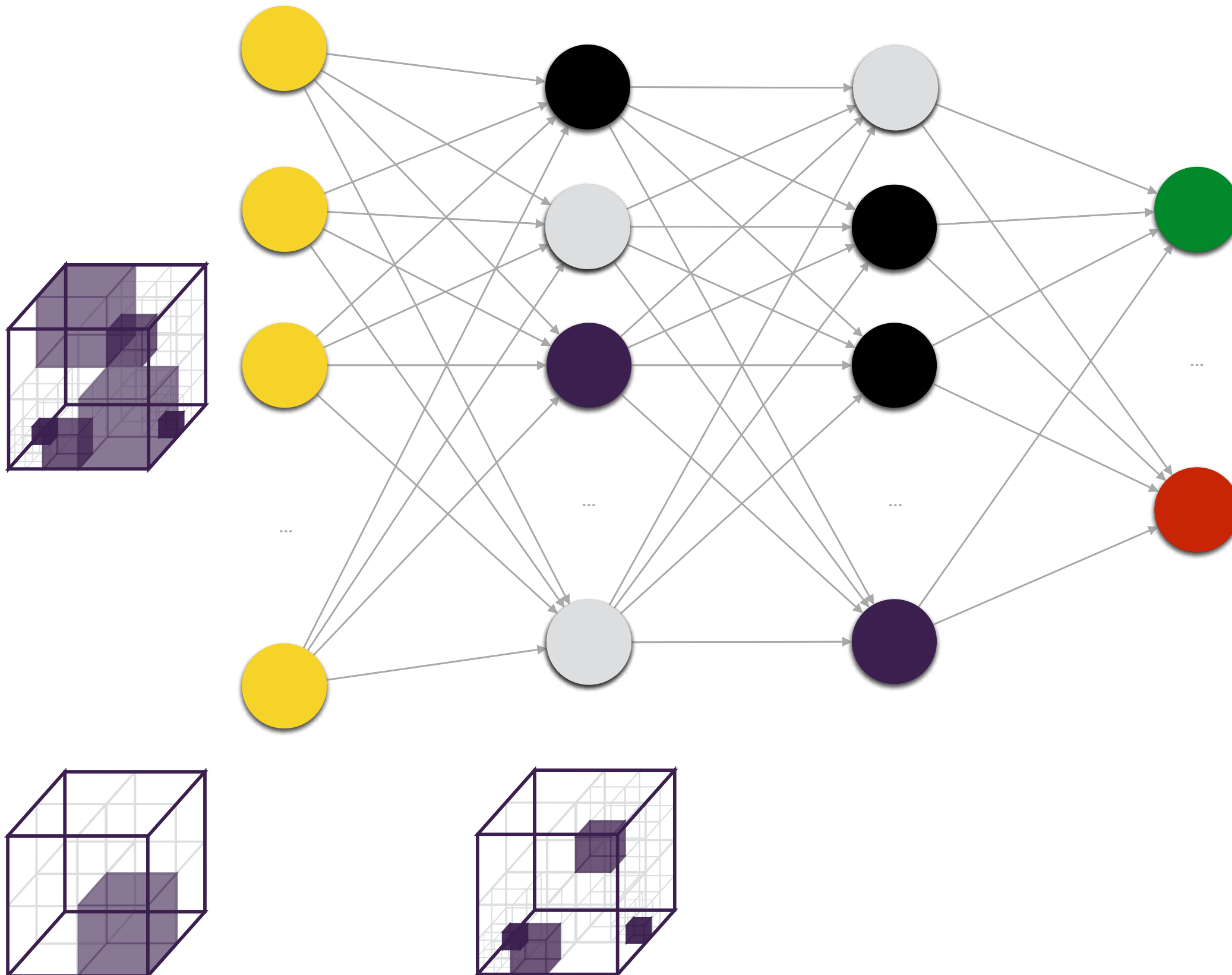
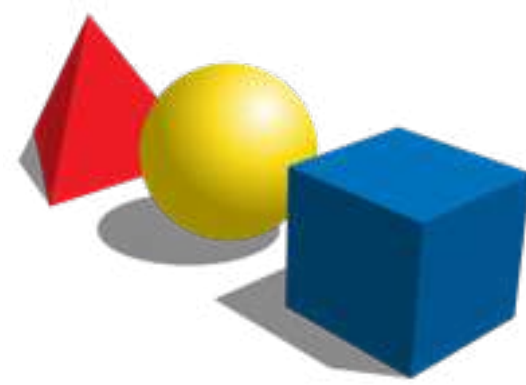
Forward and Backward Analysis

A Better Solution



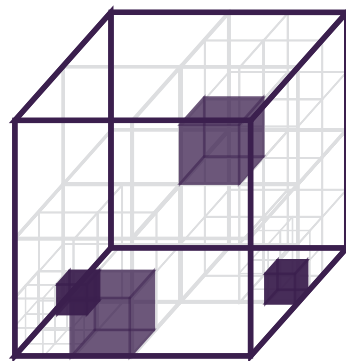
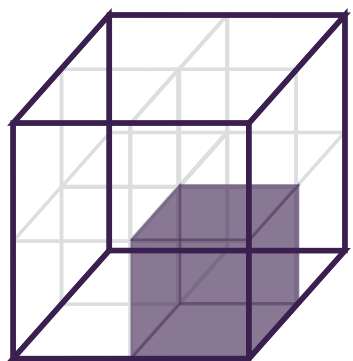
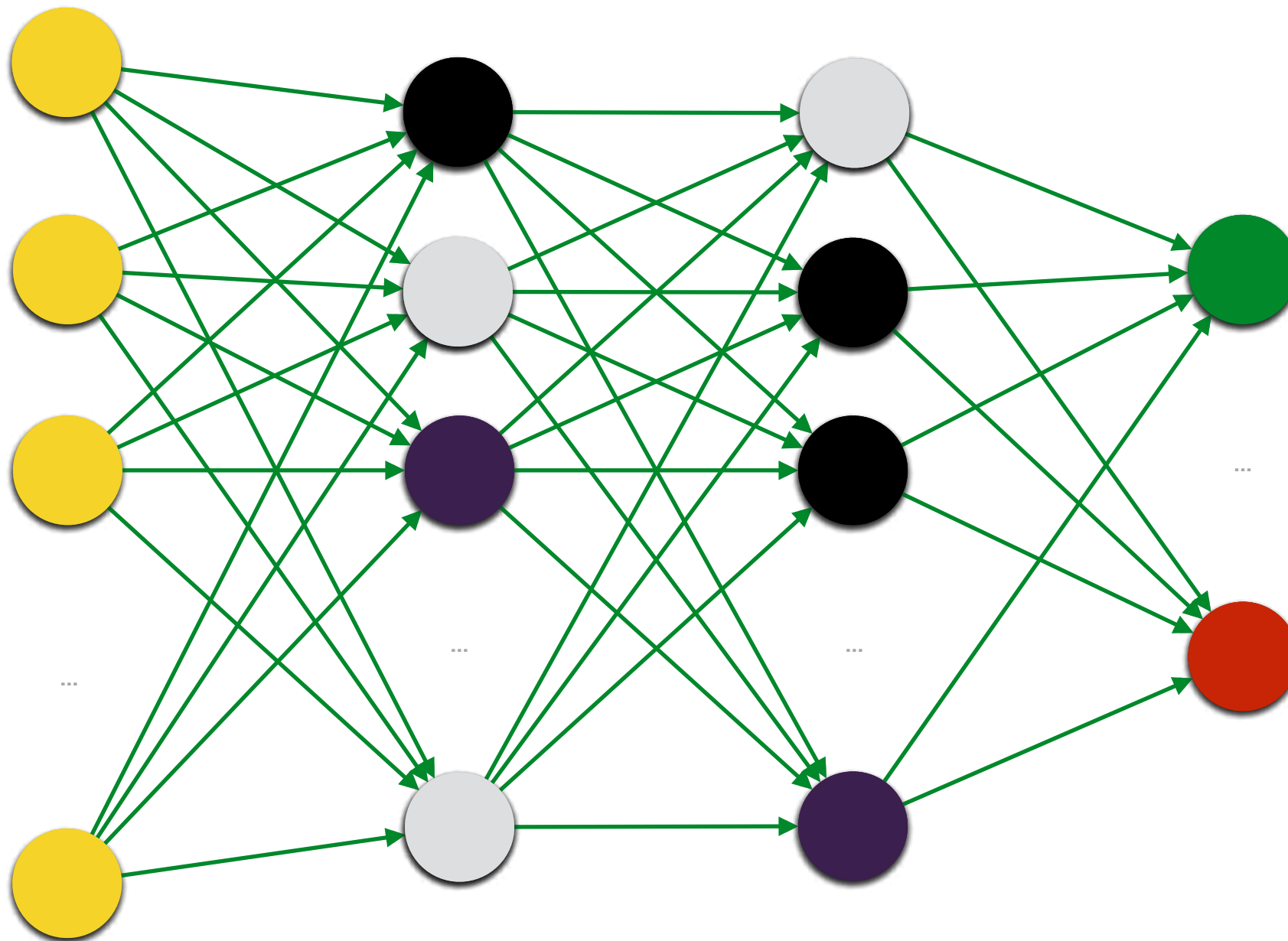
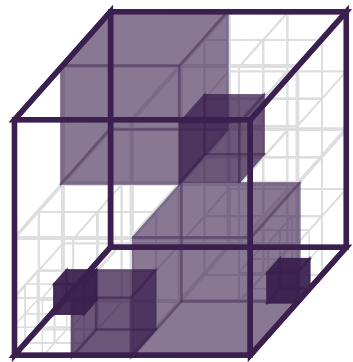
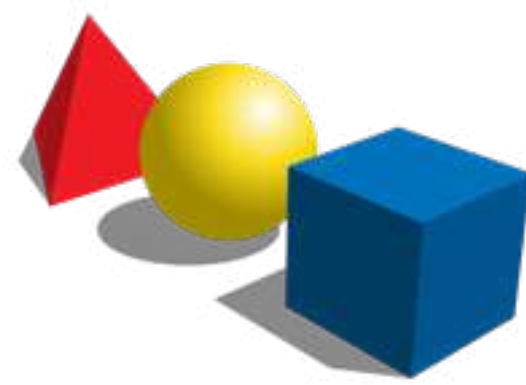
Forward and Backward Analysis

A Better Solution



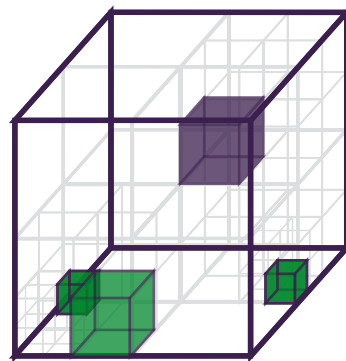
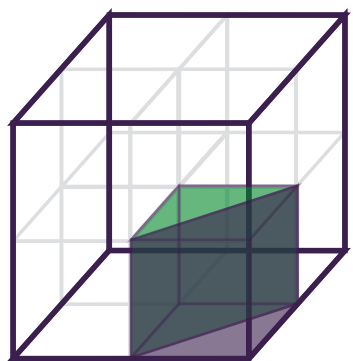
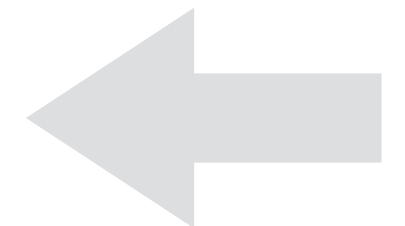
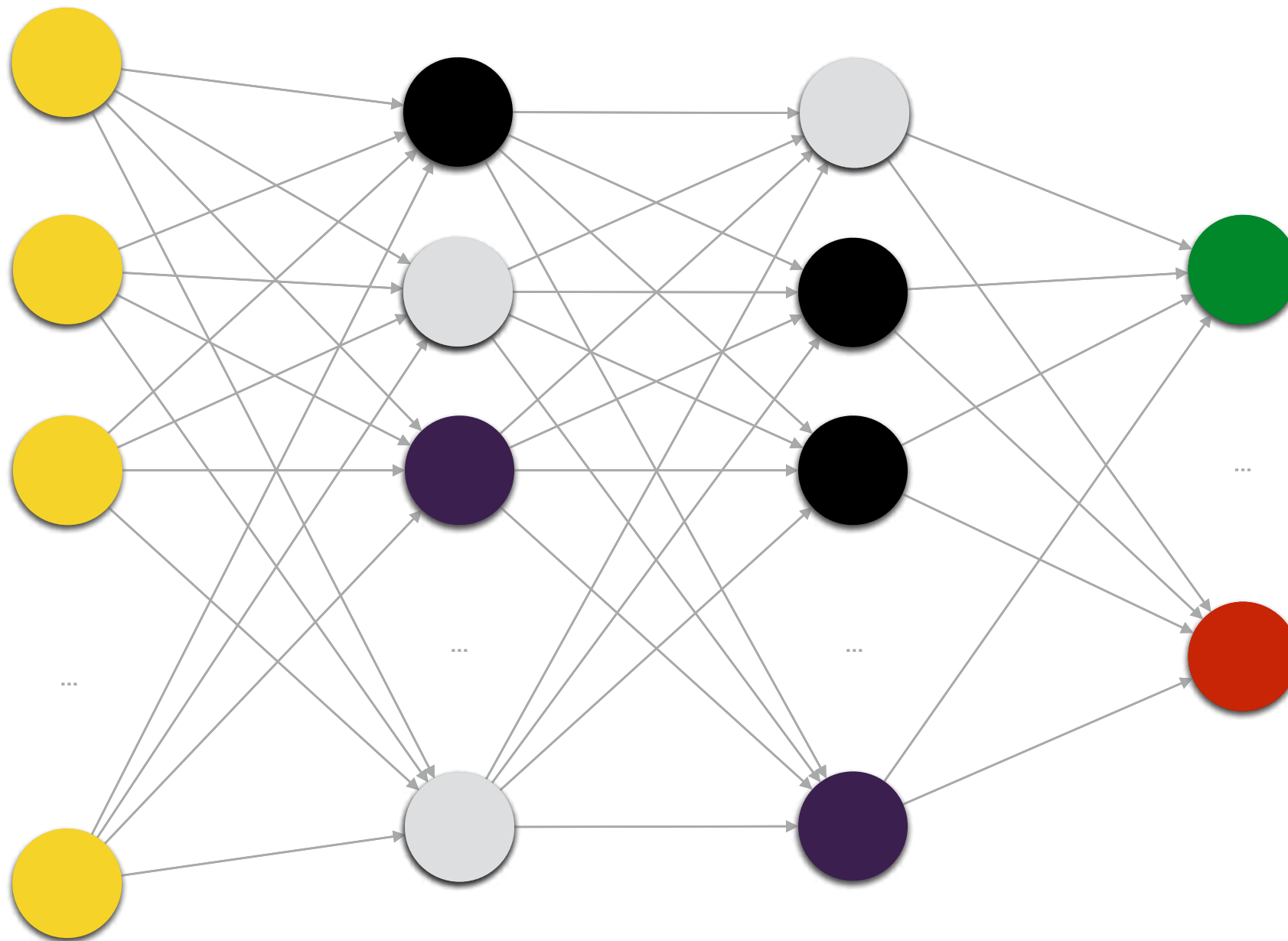
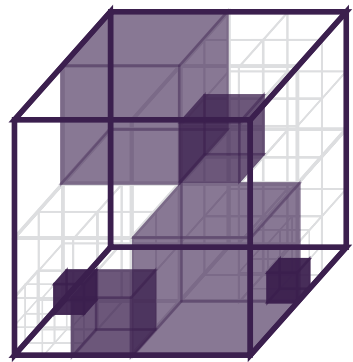
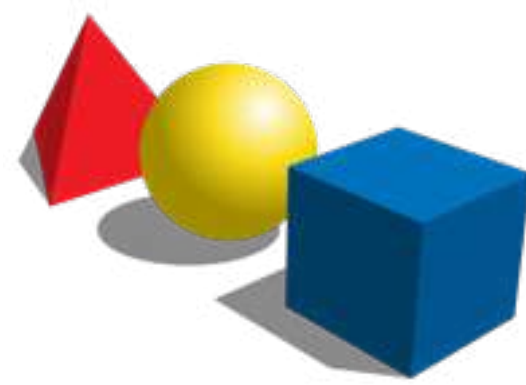
Forward and Backward Analysis

A Better Solution



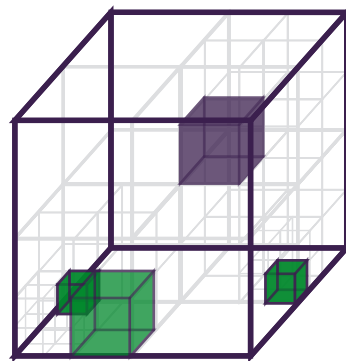
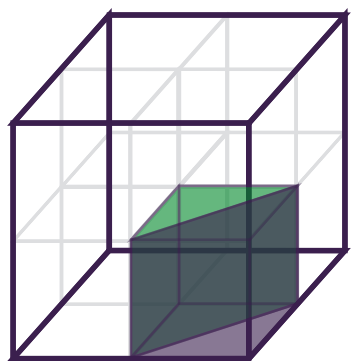
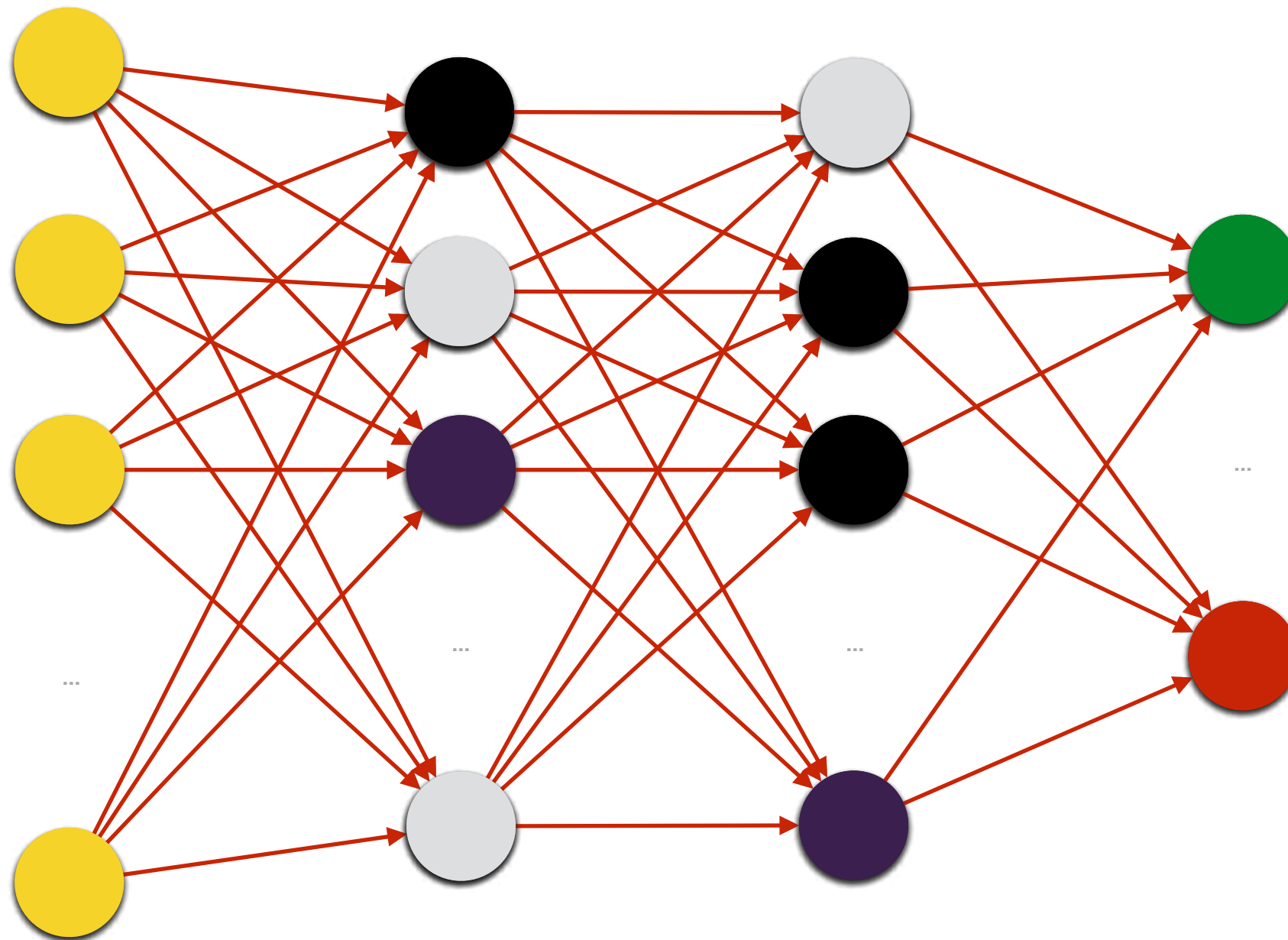
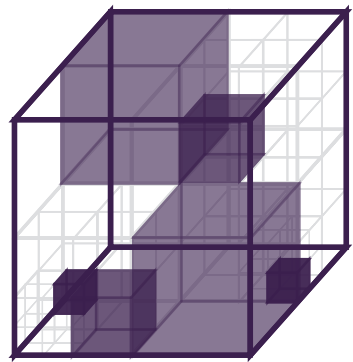
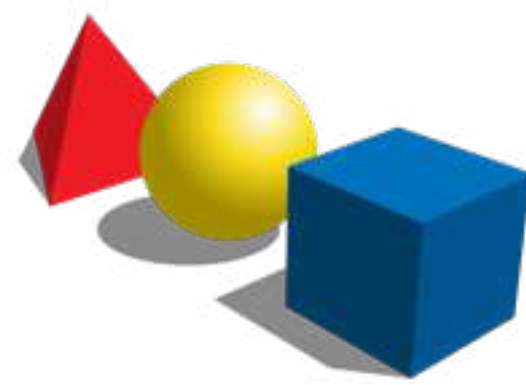
Forward and Backward Analysis

A Better Solution



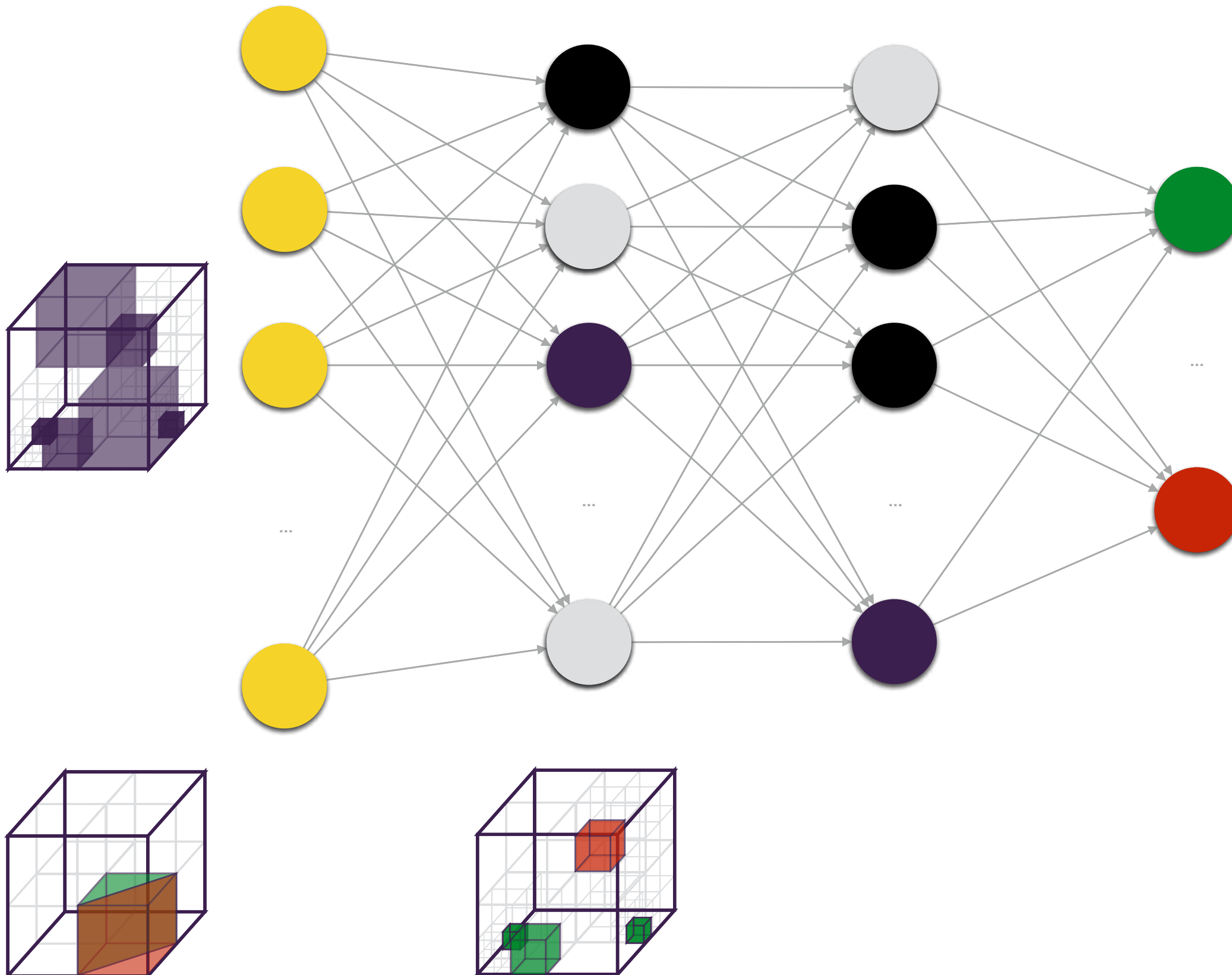
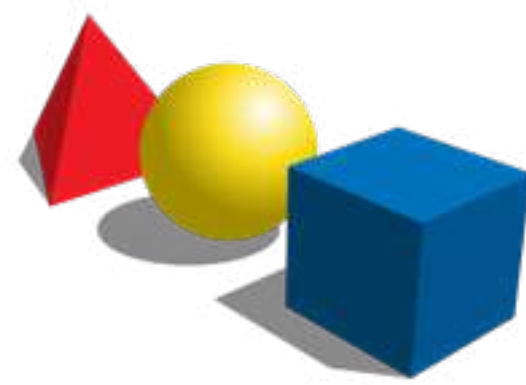
Forward and Backward Analysis

A Better Solution



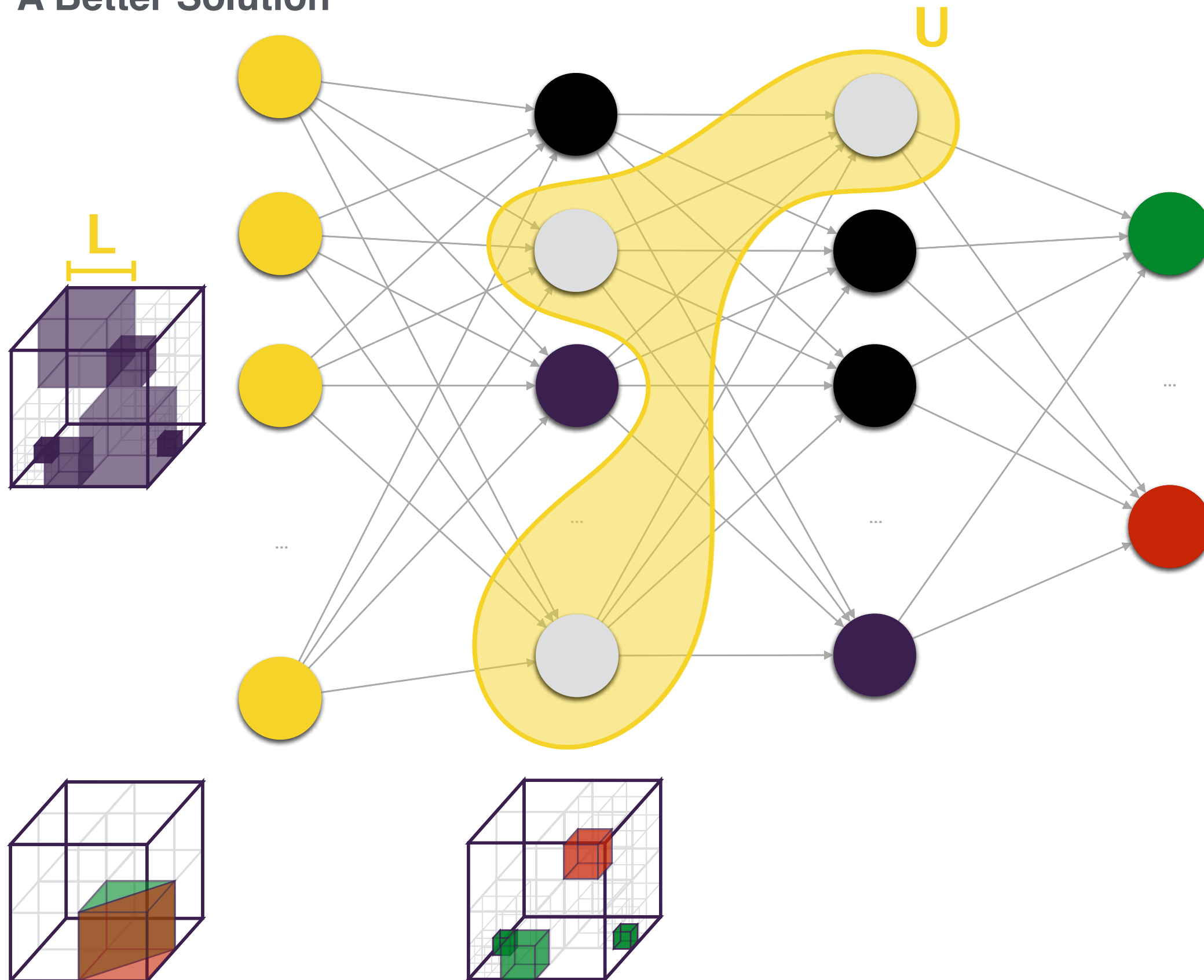
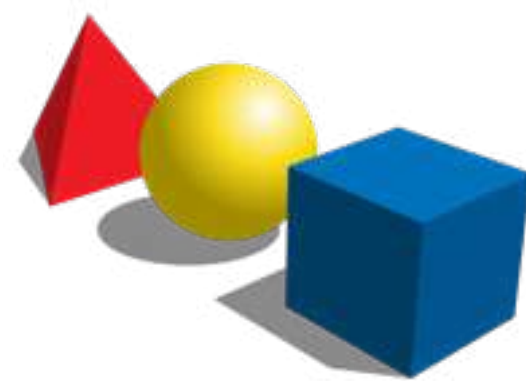
Forward and Backward Analysis

A Better Solution



Forward and Backward Analysis

A Better Solution



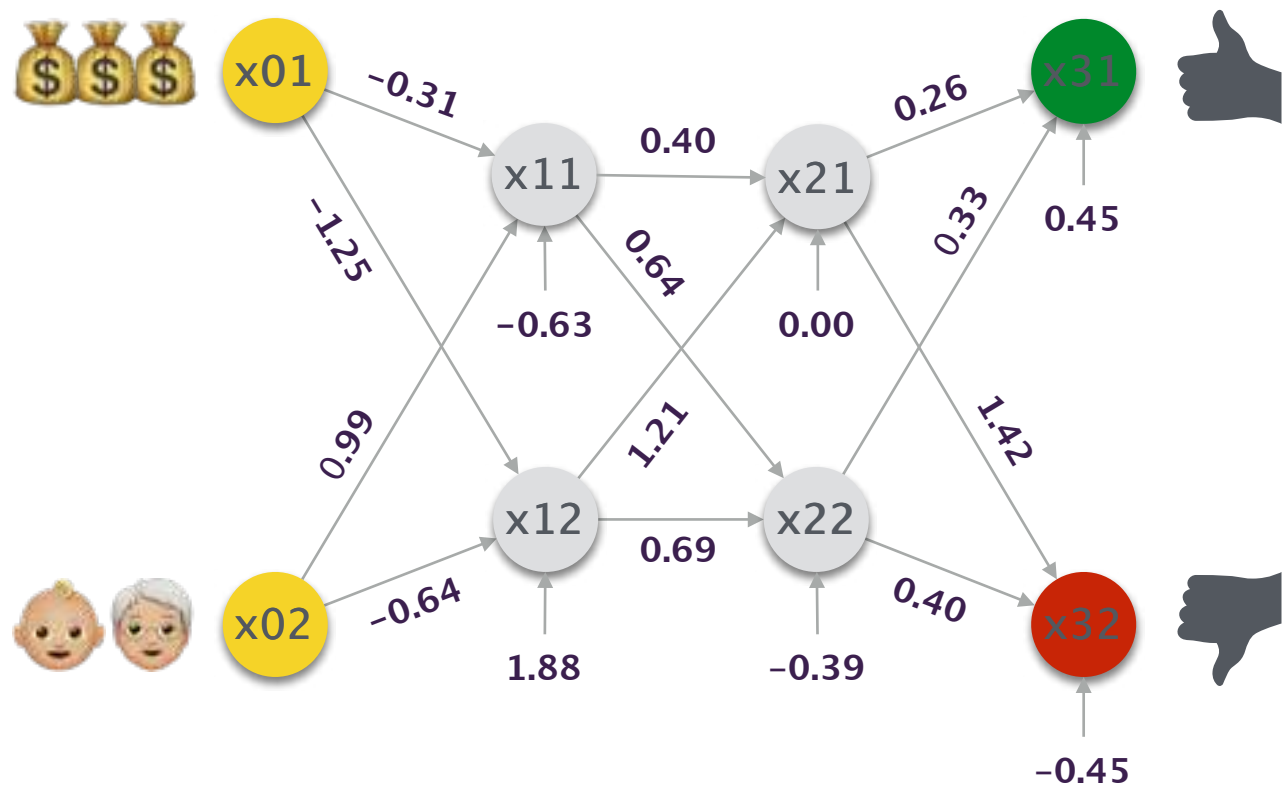
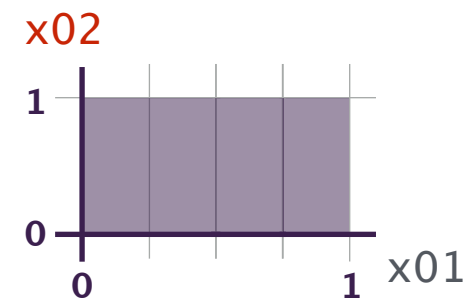
Toy Example

Activation Pattern-Based Analysis



$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

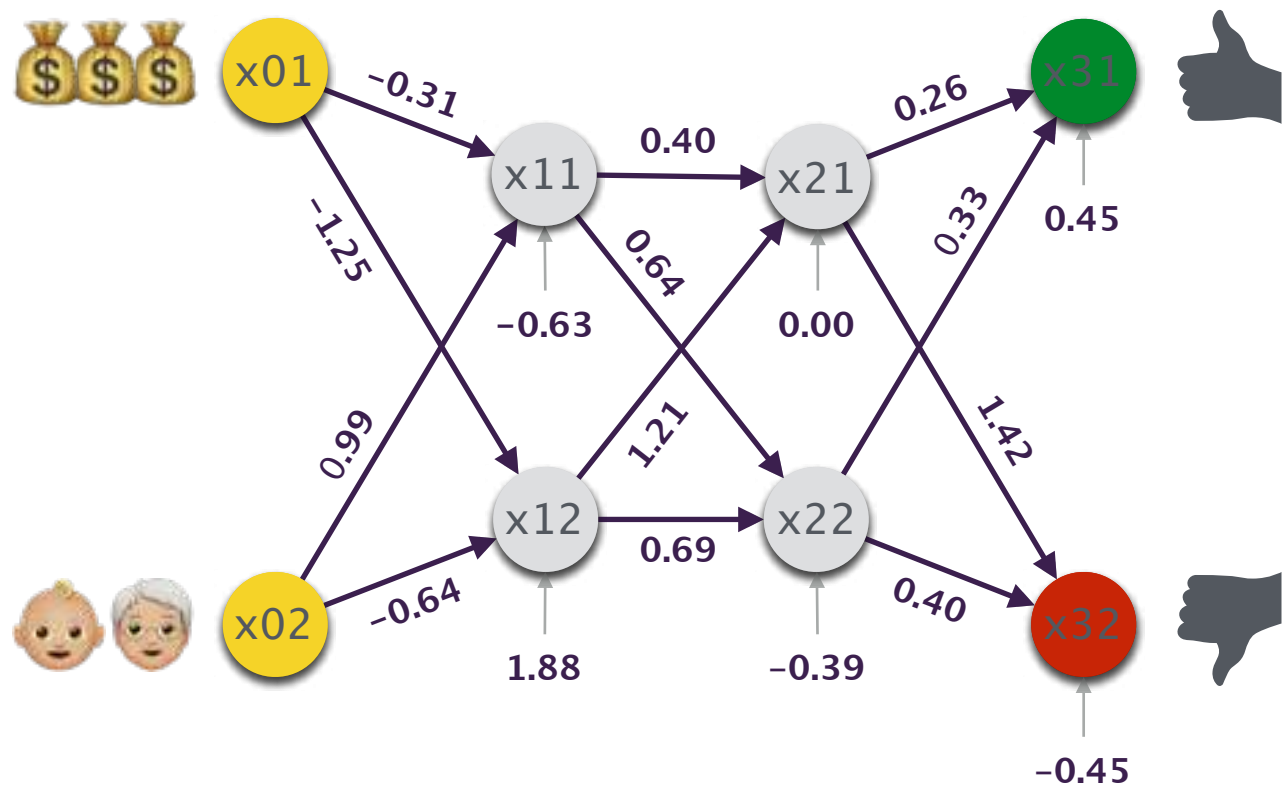
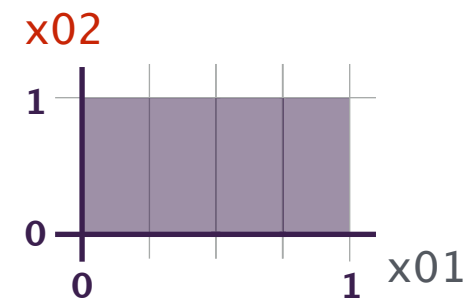

Toy Example

Activation Pattern-Based Analysis



$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

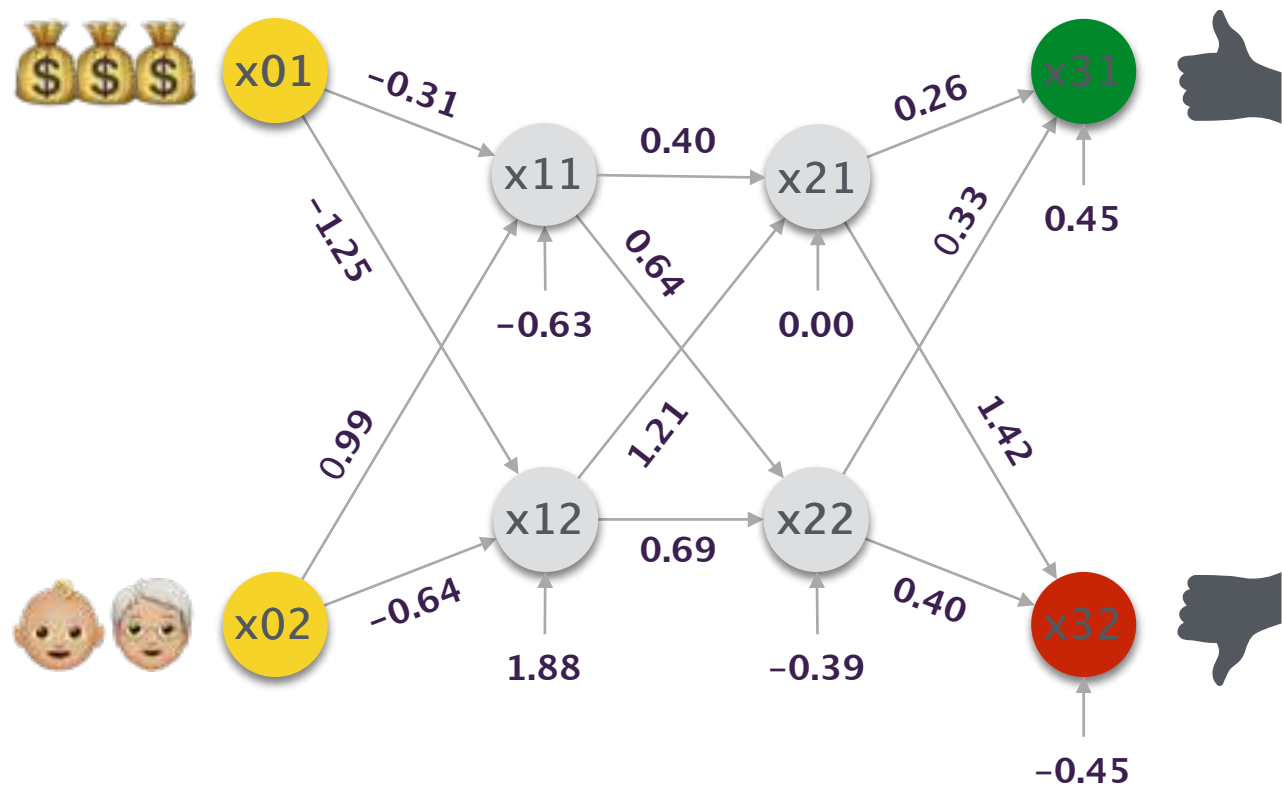
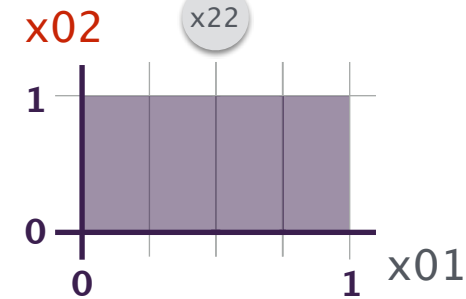
Toy Example

Activation Pattern-Based Analysis



$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

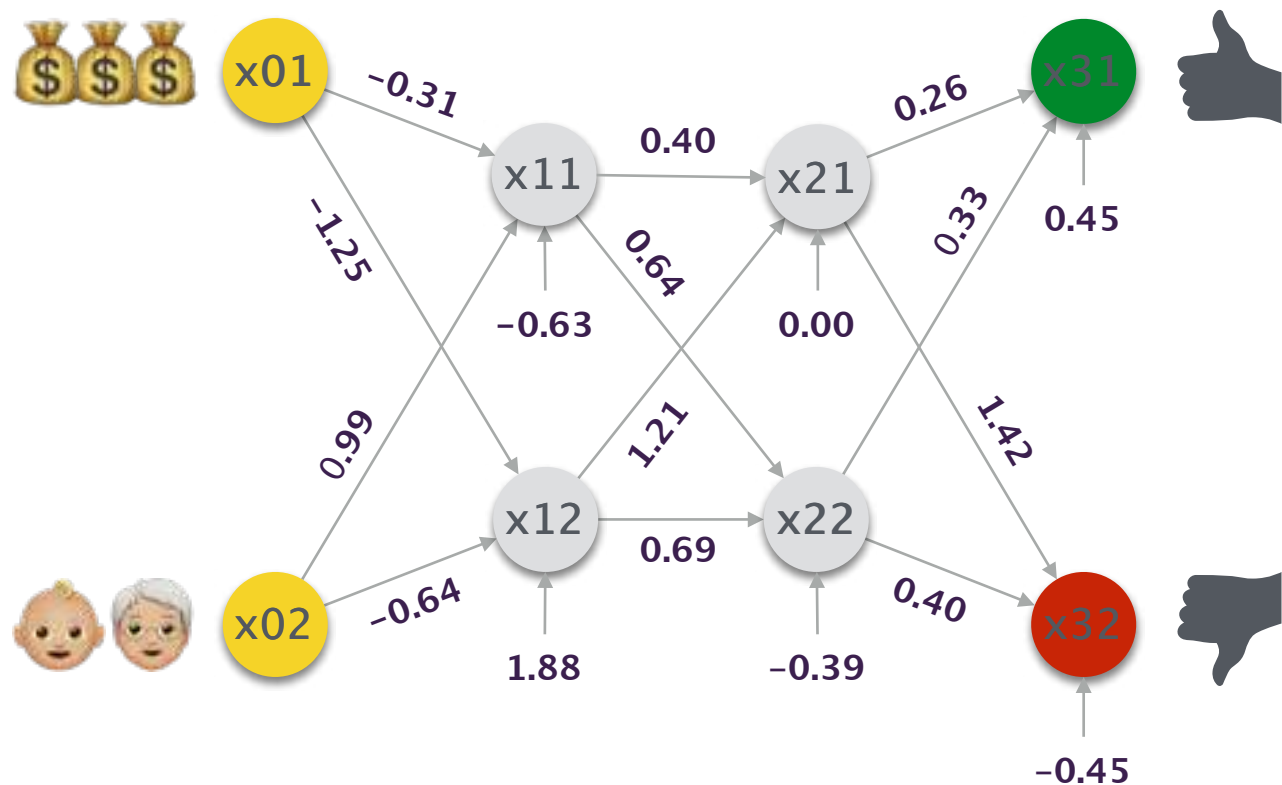
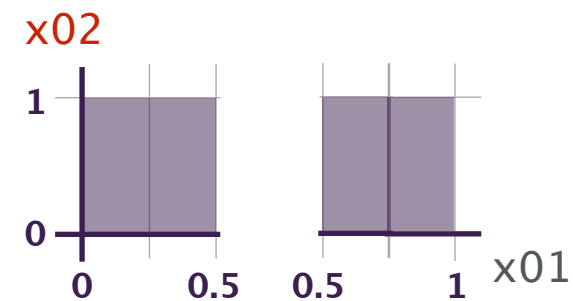
Toy Example

Activation Pattern-Based Analysis



$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

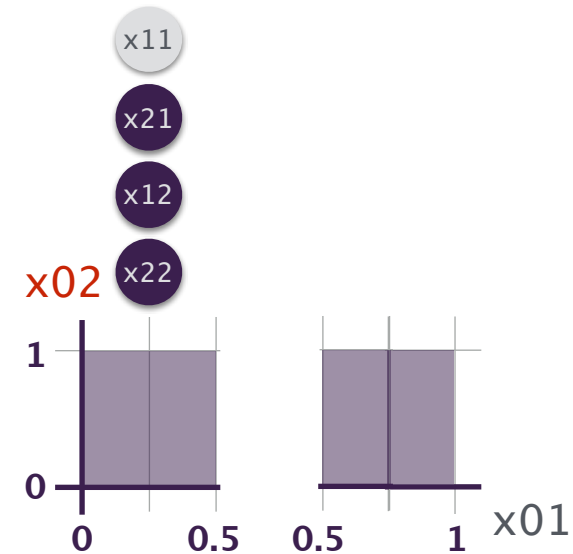
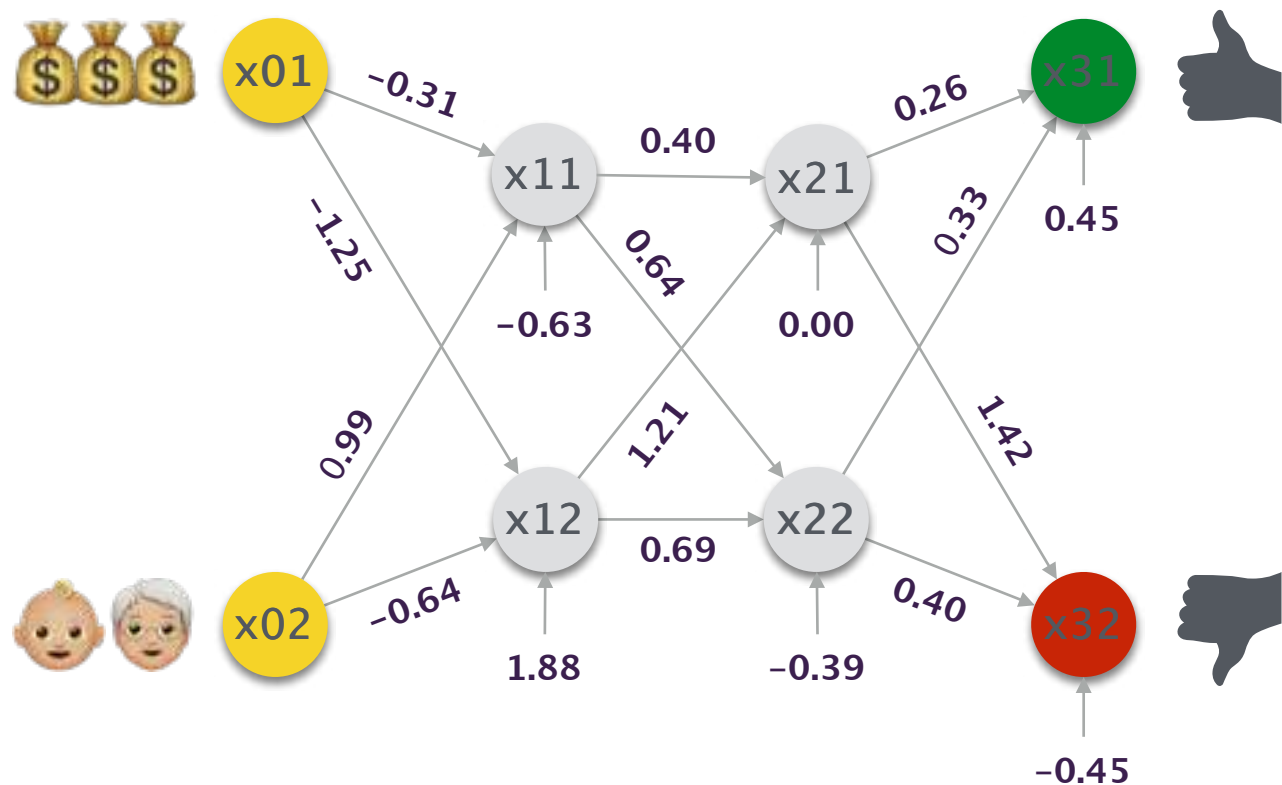
```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Toy Example

Activation Pattern-Based Analysis

$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

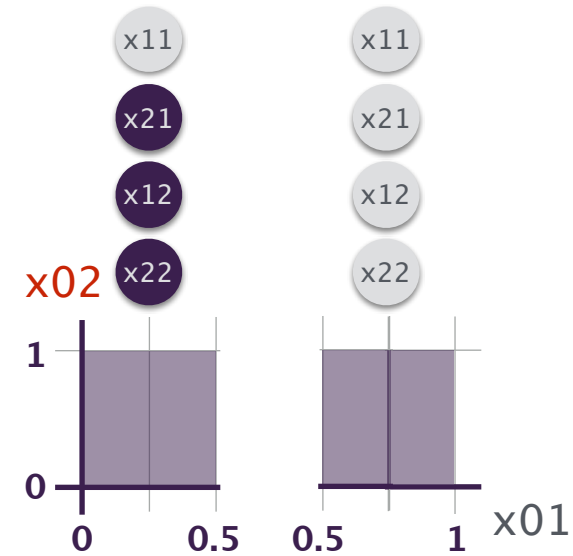
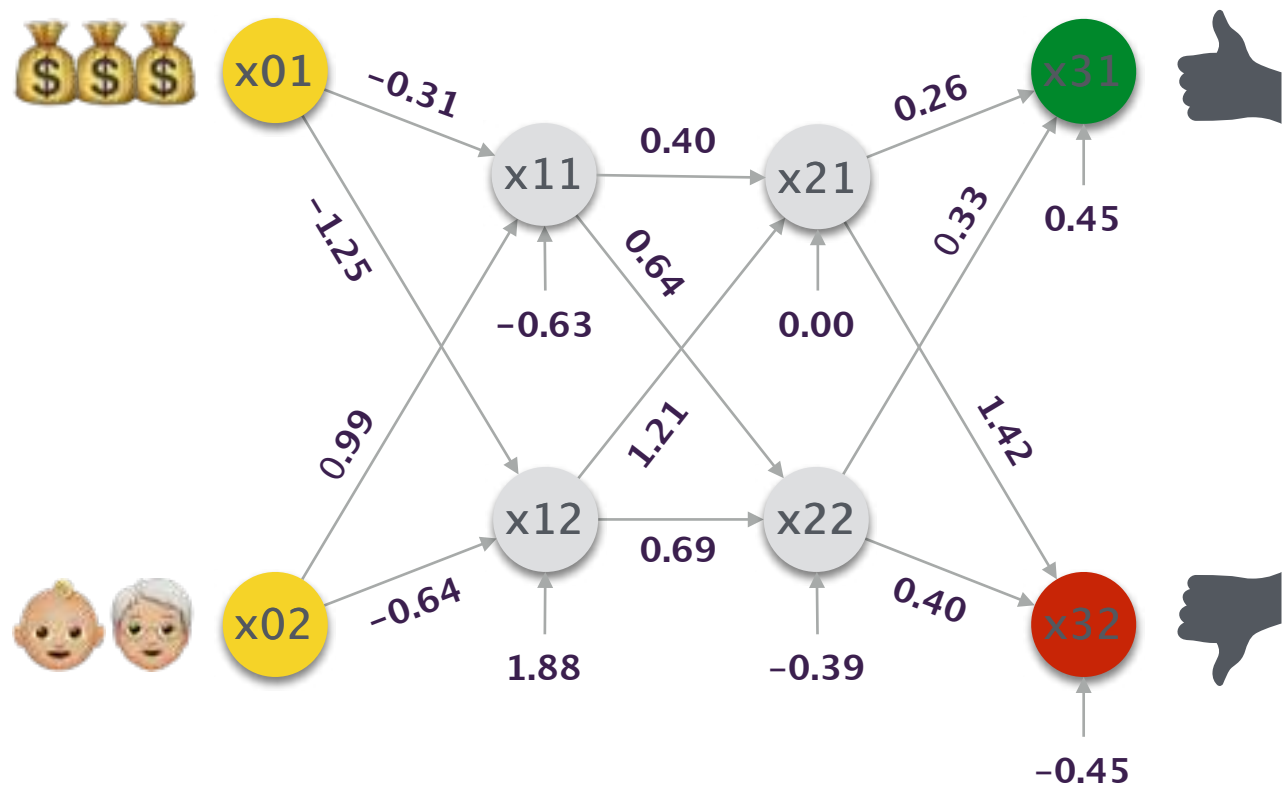
```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Toy Example

Activation Pattern-Based Analysis

$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

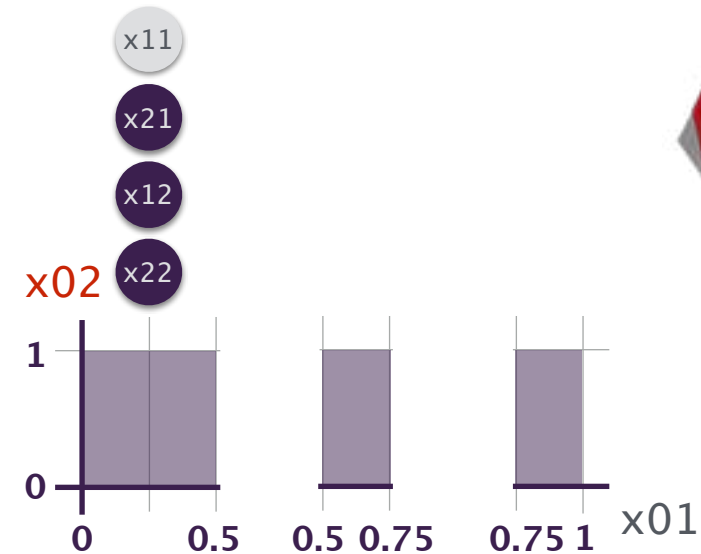
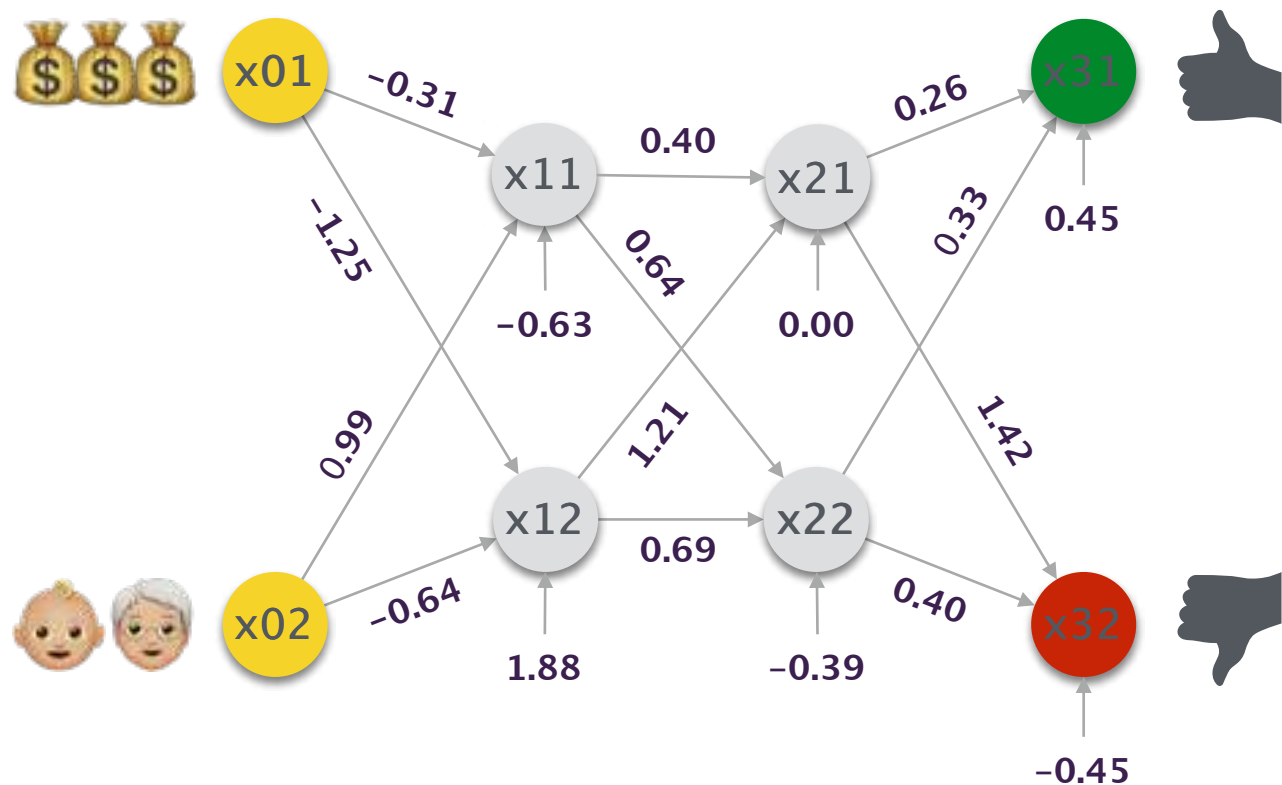
```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Toy Example

Activation Pattern-Based Analysis

$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

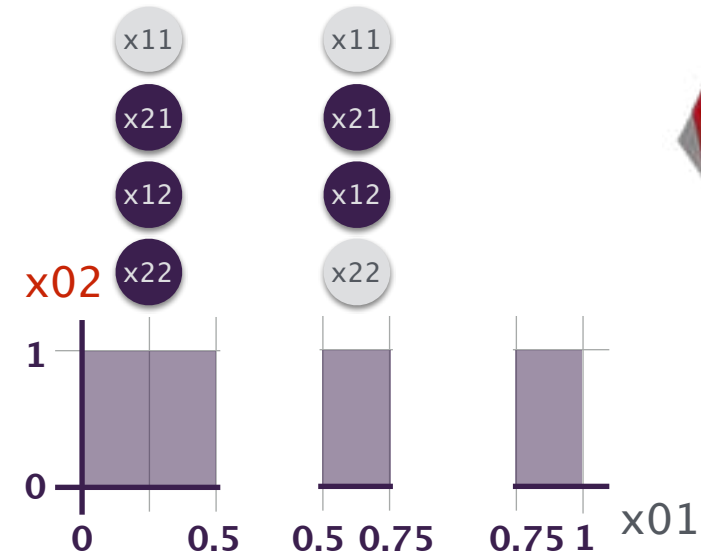
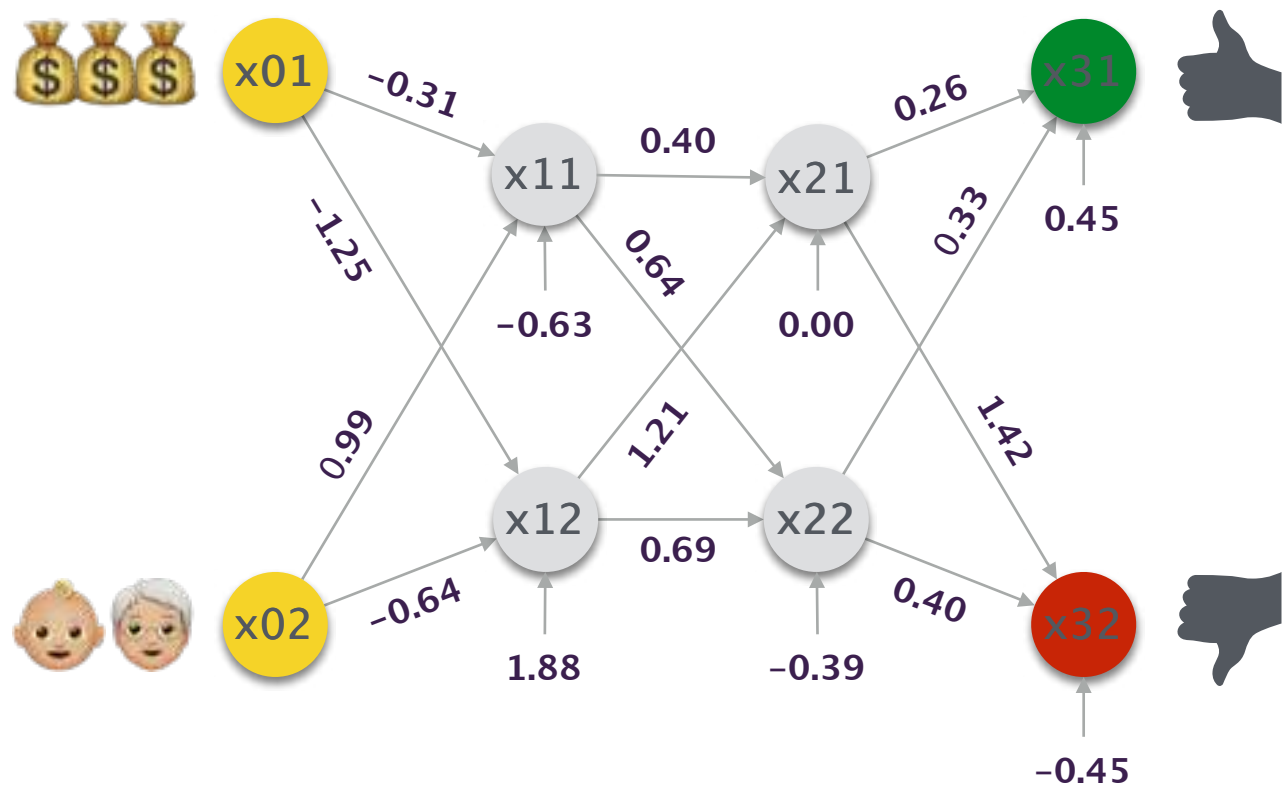
```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```


Toy Example

Activation Pattern-Based Analysis

$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

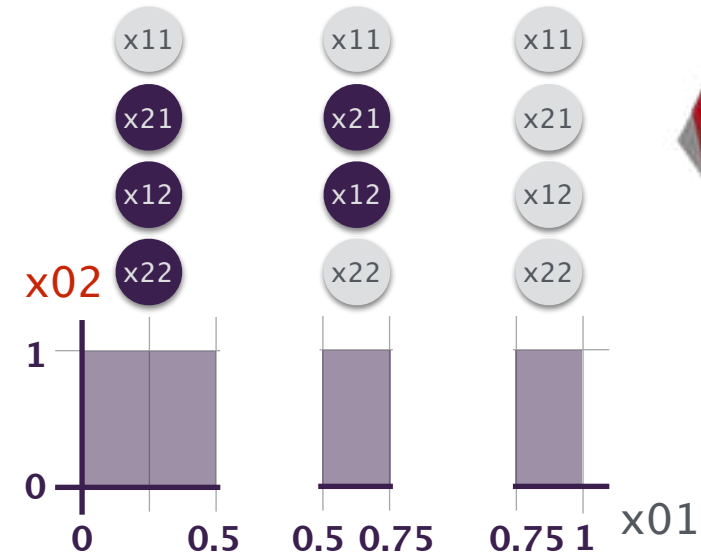
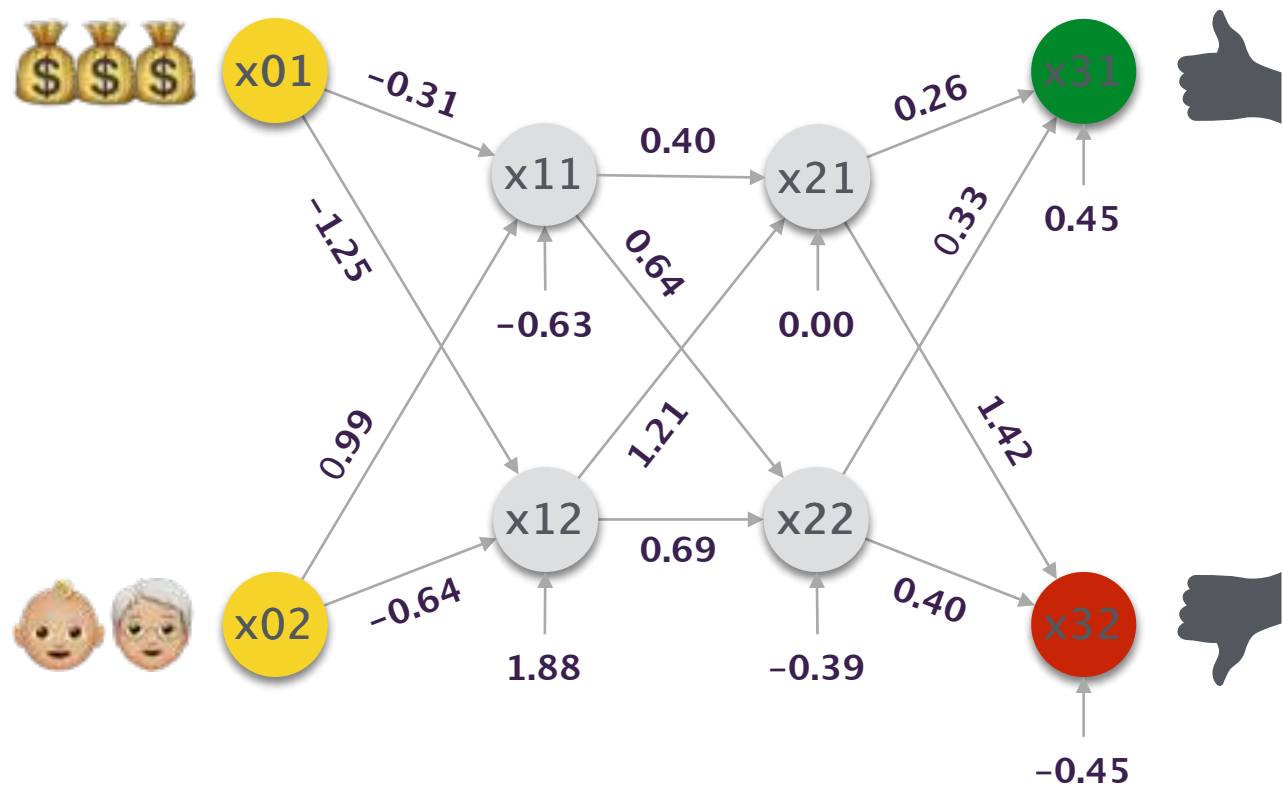
```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Toy Example

Activation Pattern-Based Analysis

$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

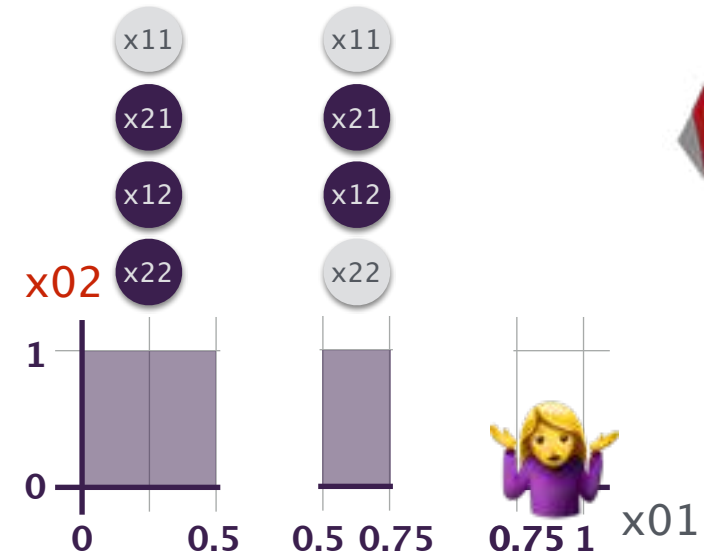
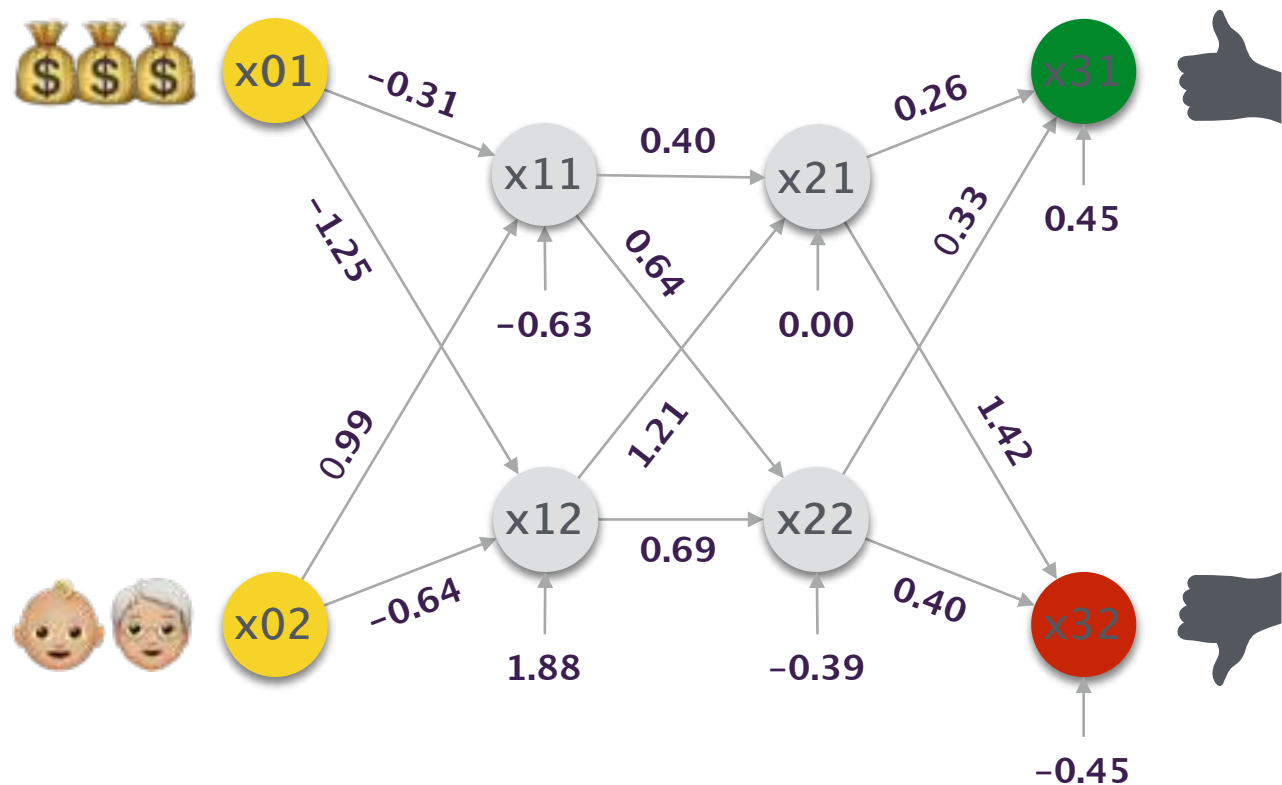
```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Toy Example

Activation Pattern-Based Analysis

$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

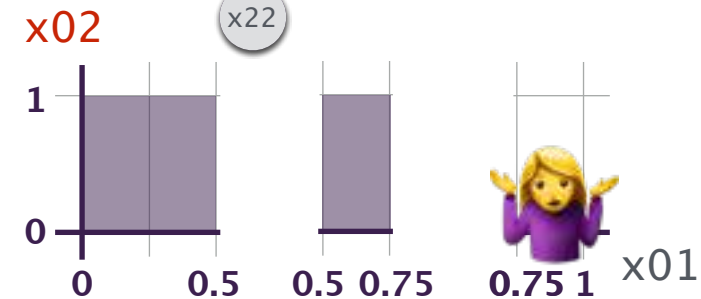
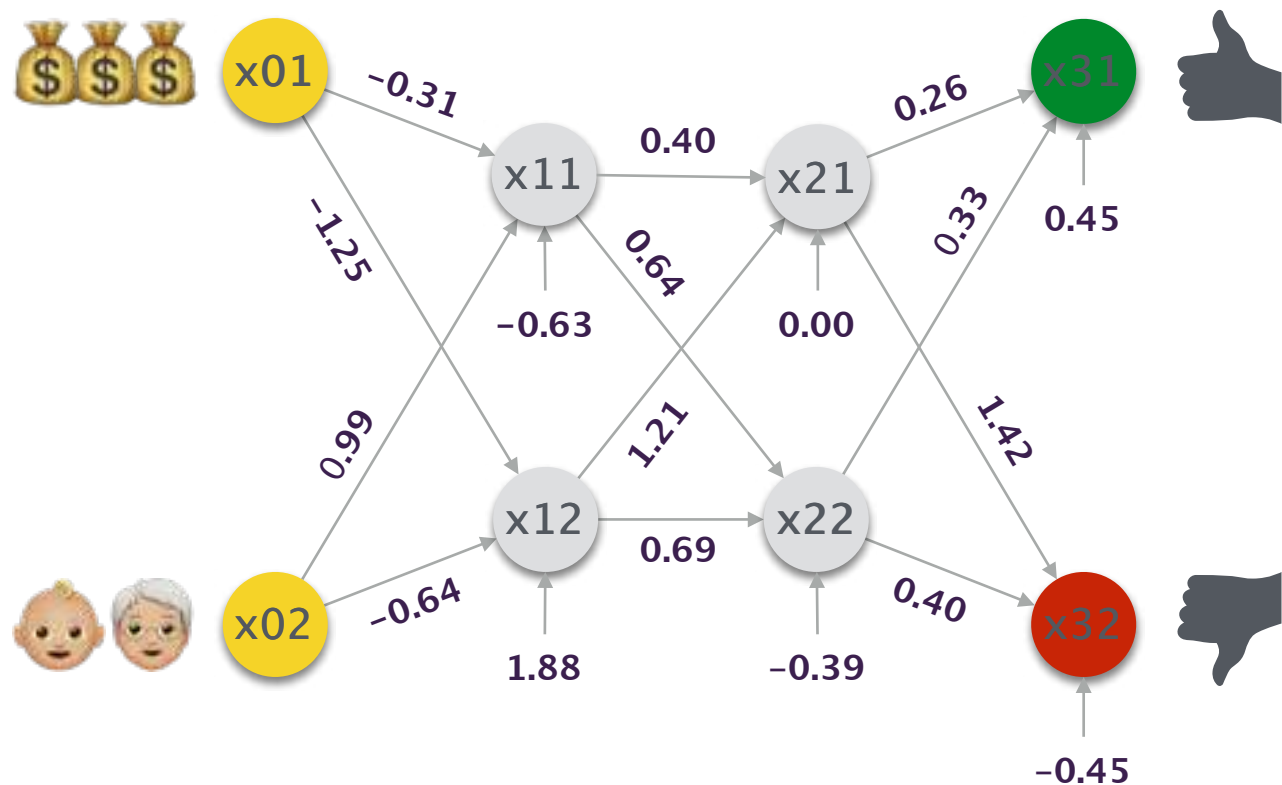
```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Toy Example

Activation Pattern-Based Analysis

$$L = 0.25$$

$$U = 2$$



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

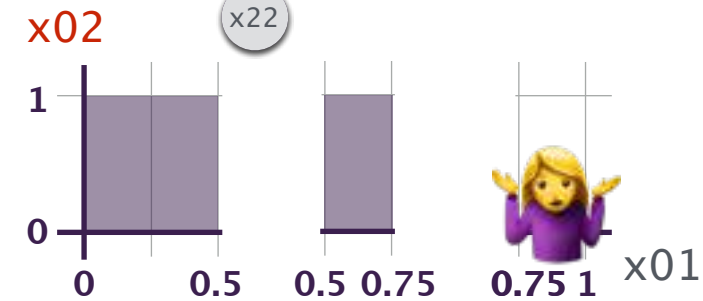
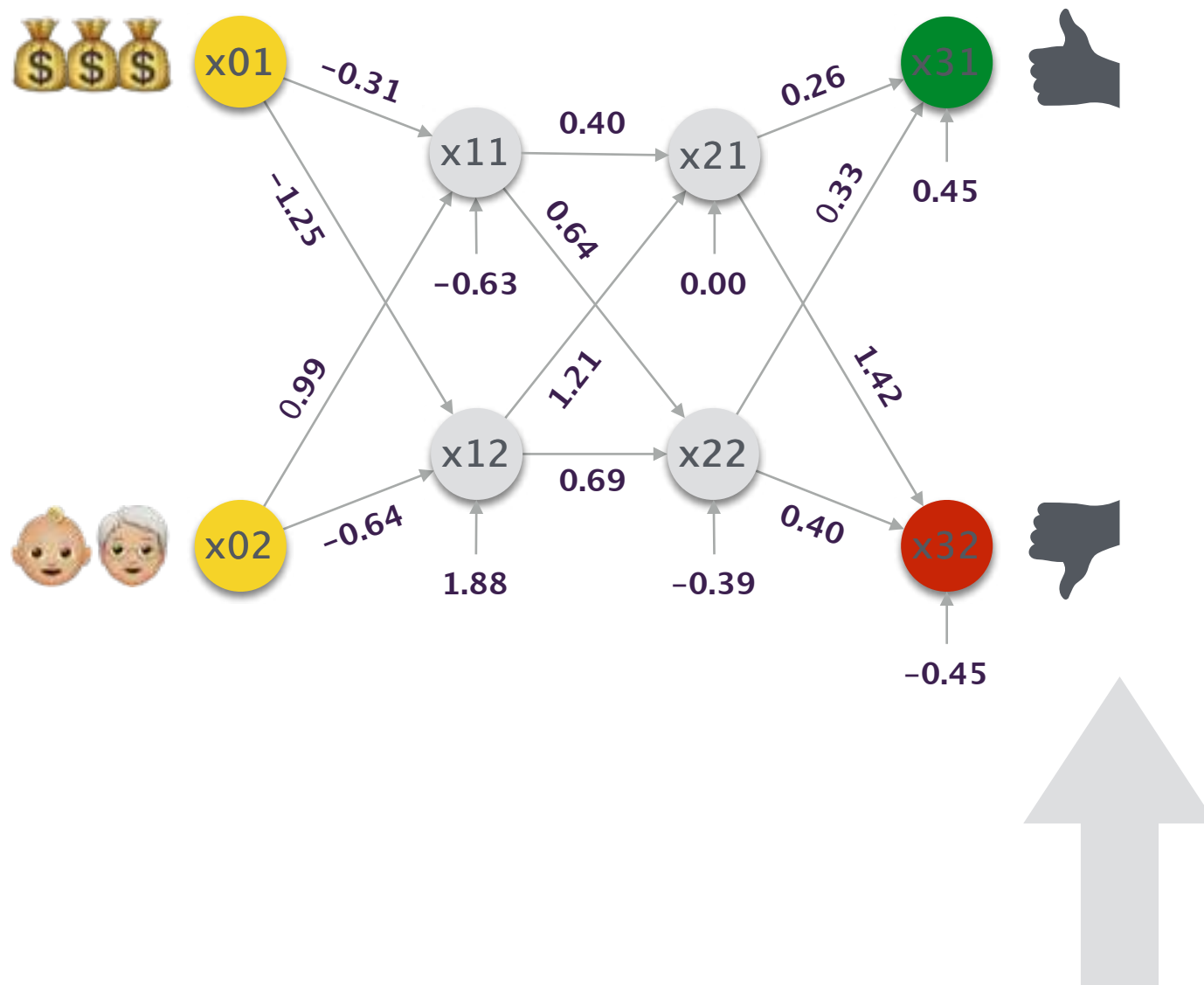
```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Toy Example

Activation Pattern-Based Analysis



```
x01 = float(input())
x02 = float(input())
```

```
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
```

```
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
```

```
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
```

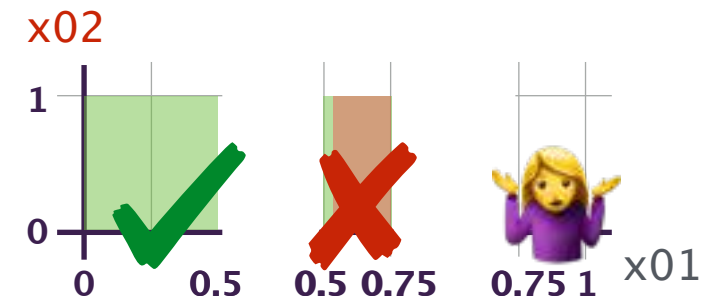
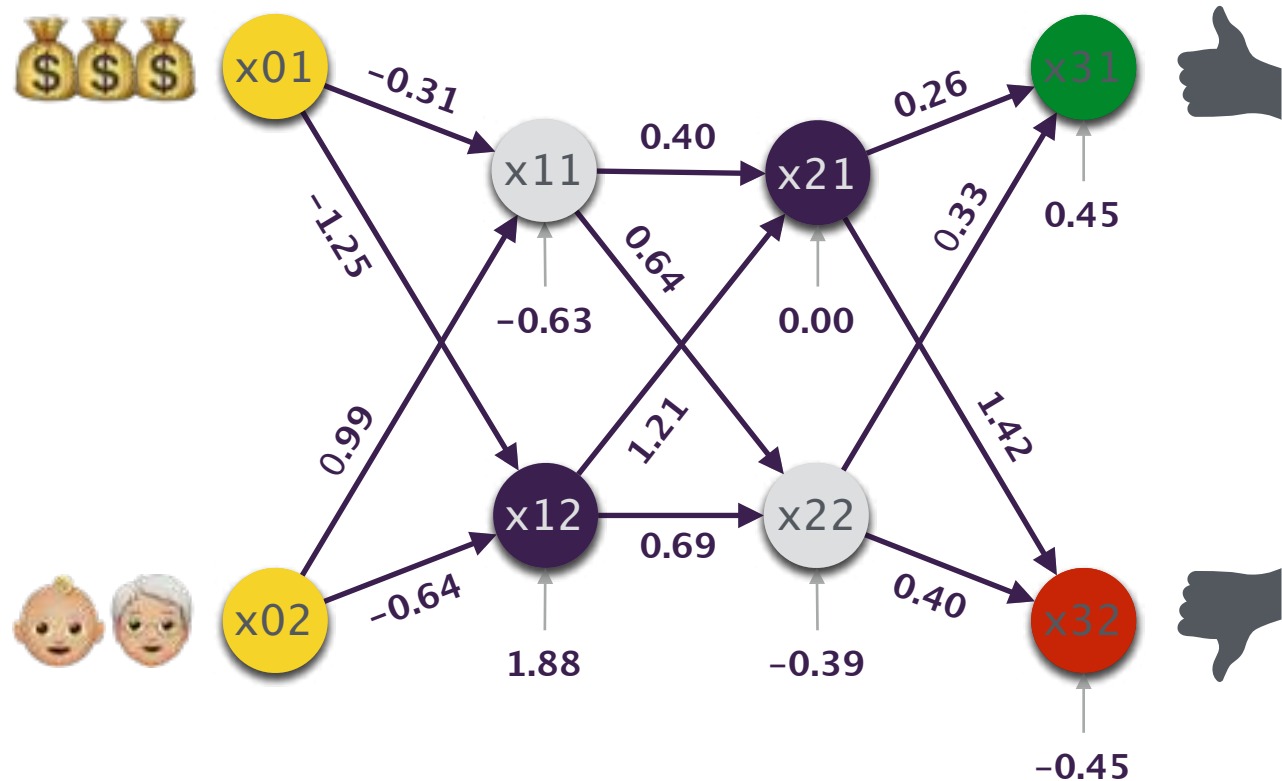
```
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
```

```
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
```

```
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
```

Toy Example

Activation Pattern-Based Analysis



```

x01 = float(input())
x02 = float(input())
x11 = -0.31 * x01 + 0.99 * x02 + (-0.63)
x12 = -1.25 * x01 + (-0.64) * x02 + 1.88
x11 = 0 if x11 < 0 else x11
x12 = 0 if x12 < 0 else x12
x21 = 0.40 * x11 + 1.21 * x12 + 0.00
x22 = 0.64 * x11 + 0.69 * x12 + (-0.39)
x21 = 0 if x21 < 0 else x21
x22 = 0 if x22 < 0 else x22
x31 = 0.26 * x21 + 0.33 * x22 + 0.45
x32 = 1.42 * x21 + 0.40 * x22 + (-0.45)
if x31 > x32:
    print('credit approved')
elif x32 < x31:
    print('credit denied')
    
```


Implementation

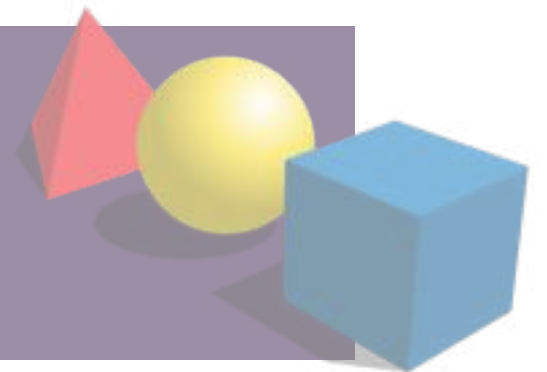
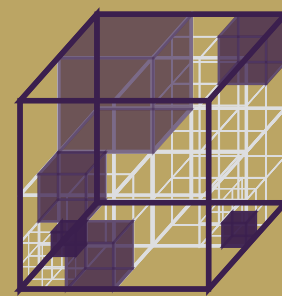
practical tools

targeting specific programs



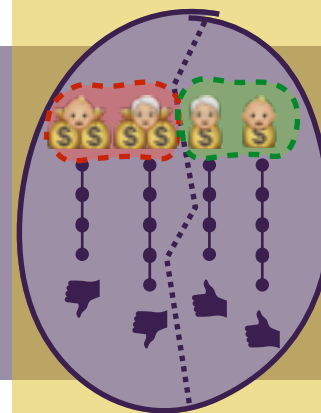
algorithmic approaches

to decide program properties



mathematical models

of the program behavior



Implementation

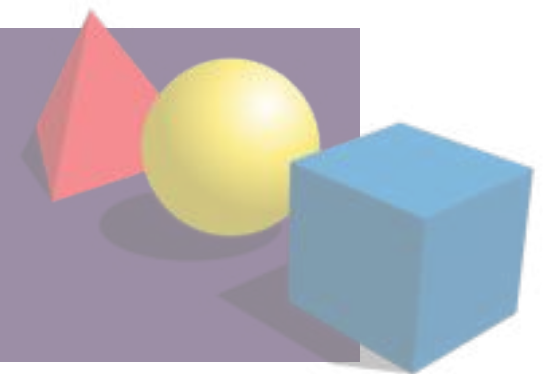
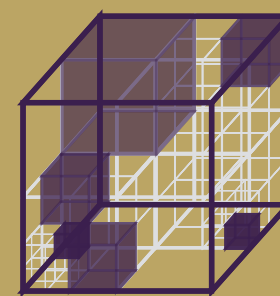
practical tools

targeting specific programs



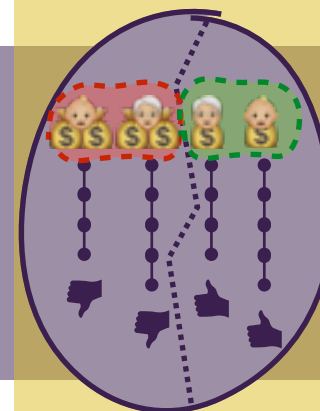
algorithmic approaches

to decide program properties



mathematical models

of the program behavior

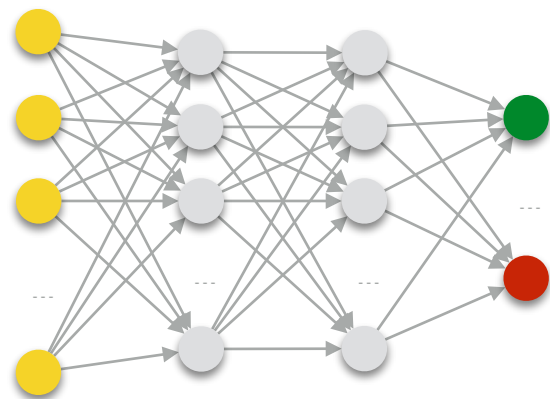


<https://github.com/caterinaurban/Libra>



Scalability-vs-Precision Tradeoff

Japanese Credit Screening Dataset

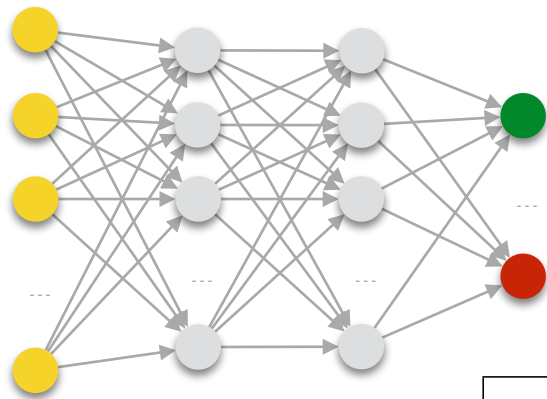


17 inputs
4 HL * 5 N
2 classes
86% accuracy

L	U	◆ BOXES					▲ SYMBOLIC					★ DEEPPOLY				
		INPUT	C	F		TIME	INPUT	C	F		TIME	INPUT	C	F		TIME
0.5	4	15.28%	37	0	0	8s	58.33%	79	8	20	1m 26s	69.79%	115	10	39	3m 18s
	6	17.01%	39	6	6	51s	69.10%	129	22	61	5m 41s	80.56%	104	23	51	7m 53s
	8	51.39%	90	28	85	12m 2s	82.64%	88	31	67	12m 35s	91.32%	84	27	56	19m 33s
	10	79.86%	89	34	89	34m 15s	93.06%	98	40	83	42m 32s	96.88%	83	29	58	43m 39s
0.25	4	59.09%	1115	20	415	54m 32s	95.94%	884	39	484	54m 31s	98.26%	540	65	293	14m 29s
	6	83.77%	1404	79	944	37m 19s	98.68%	634	66	376	23m 31s	99.70%	322	79	205	13m 25s
	8	96.07%	869	140	761	1h 7m 29s	99.72%	310	67	247	1h 3m 33s	99.98%	247	69	177	22m 52s
	10	99.54%	409	93	403	1h 35m 20s	99.98%	195	52	176	1h 2m 13s	100.00%	111	47	87	34m 56s
0.125	4	97.13%	12449	200	9519	3h 33m 48s	99.99%	1101	60	685	47m 46s	99.99%	768	81	415	19m 1s
	6	99.83%	5919	276	4460	3h 23m	100.00%	988	77	606	26m 47s	100.00%	489	80	298	16m 54s
	8	99.98%	1926	203	1568	2h 14m 25s	100.00%	404	73	309	46m 31s	100.00%	175	57	129	20m 11s
	10	100.00%	428	95	427	1h 39m 31s	100.00%	151	53	141	57m 32s	100.00%	80	39	62	28m 33s
0	4	100.00%	19299	295	15446	6h 13m 24s	100.00%	1397	60	885	40m 5s	100.00%	766	87	425	16m 41s
	6	100.00%	4843	280	3679	2h 24m 7s	100.00%	763	66	446	35m 24s	100.00%	401	81	242	32m 29s
	8	100.00%	1919	208	1567	2h 9m 59s	100.00%	404	73	309	45m 48s	100.00%	193	68	144	24m 16s
	10	100.00%	486	102	475	1h 41m 3s	100.00%	217	55	192	1h 2m 11s	100.00%	121	50	91	30m 53s

Seeded Bias

German Credit Dataset (L = 0)



17 inputs
4 HL * 5 N
2 classes
71% accuracy

17 inputs
4 HL * 5 N
2 classes
65% accuracy

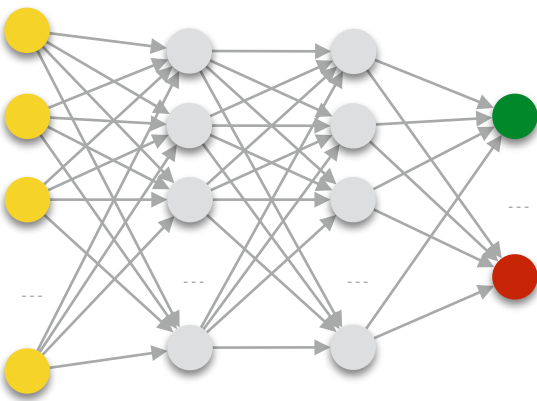
CREDIT	DEEPPOLY											
	FAIR DATA					BIASED DATA						
	U	BIAS	C	F		TIME	U	BIAS	C	F		TIME
≤ 1000	8	0.33%	170	21	25	3m 40s	8	0.79%	260	42	53	5m 42s
	6	0.17%	211	10	10	4m 5s	4	0.31%	218	9	20	1m 6s
	2	0.09%	176	4	5	14s	12	0.82%	271	53	61	18m 18s
	7	0.15%	212	9	9	1m 31s	4	0.42%	242	21	28	1m 36s
	3	0.23%	217	8	15	32s	10	0.95%	260	42	67	3m 2s
	12	0.30%	213	17	23	5m 45s	2	0.41%	226	20	26	1m 56s
	6	0.20%	193	11	11	52s	3	0.48%	228	19	34	39s
	5	0.16%	193	9	10	10s	1	0.09%	206	5	5	51s
MIN		0.09%			10s		0.09%				39s	
MEDIAN		0.19%			1m 12s		0.45%				1m 46s	
MAX		0.33%			5m 45s		0.95%				18m 18s	
> 1000	10	12.08%	321	85	150	10m 30s	11	27.59%	498	234	333	1h 16m 41s
	11	7.43%	329	75	125	22m 33s	7	30.77%	394	70	228	6m 34s
	2	2.21%	217	15	16	39s	7	33.17%	435	185	327	6h 51m 50s
	10	4.29%	239	24	33	4m 4s	6	16.45%	448	162	260	18m 25s
	4	9.73%	268	29	87	4m 0s	13	30.17%	418	141	332	43m 12s
	14	14.96%	403	116	231	1h 9m 45s	5	17.24%	460	91	217	12m 53s
	7	5.83%	313	92	115	4m 17s	8	19.23%	363	79	189	7m 24s
	9	4.61%	264	50	74	5m 38s	2	4.52%	331	45	95	4m 44s
MIN		2.21%			39s		4.52%				4m 44s	
MEDIAN		6.63%			4m 58s		23.41%				15m 39s	
MAX		14.96%			1h 9m 45s		31.17%				6h 51m 50s	

Bias Queries

ProPublica COMPAS Dataset (L = 0)



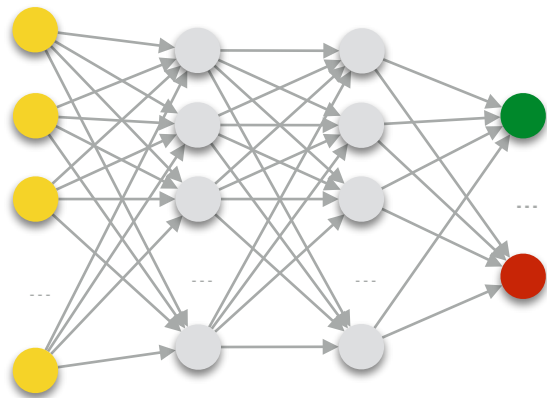
QUERY	DEEPPOLY											
	FAIR DATA					BIASED DATA						
	U	BIAS	C	F		TIME	U	BIAS	C	F		TIME
AGE < 25 RACE BIAS?	10	0.23%	71	18	20	1h 11m 43s	10	0.83%	43	15	33	2h 5m 5s
	10	0.75%	33	14	16	10m 33s	10	6.48%	63	25	34	8m 46s
	10	0.22%	34	17	22	52m 29s	10	1.15%	33	10	14	11m 58s
	10	0.24%	118	28	29	42m 2s	10	0.42%	31	13	30	10m 51s
	10	0.31%	117	49	54	1h 0m 2s	10	0.12%	37	11	16	18m 18s
	10	0.33%	59	18	21	53m 29s	10	2.27%	33	16	24	1h 4m 35s
	10	1.19%	39	17	23	9m 39s	10	3.41%	133	92	102	33m 43s
	10	2.12%	33	17	31	5m 18s	10	0.18%	33	12	17	14m 58s
MIN		0.22%			5m 18s		0.12%				8m 46s	
MEDIAN		0.32%			47m 16s		0.99%				16m 38s	
MAX		2.12%			1h 11m 43s		6.48%				2h 5m 5s	
MALE AGE BIAS?	10	3.86%	242	96	180	2h 30m 23s	10	5.22%	204	65	180	3h 25m 21s
	10	8.84%	100	45	77	19m 47s	10	12.38%	387	152	318	40m 49s
	10	8.14%	204	47	143	28m 12s	10	7.10%	181	63	142	20m 51s
	10	2.70%	563	168	232	1h 49m 9s	10	6.90%	96	23	95	1h 21m 37s
	10	4.65%	545	280	415	1h 33m 36s	10	6.14%	157	62	110	27m 43s
	10	5.77%	217	68	154	1h 35m 25s	10	8.10%	345	61	284	47m 9s
	10	7.76%	252	62	226	23m 10s	10	6.78%	251	141	223	50m 13s
	10	8.70%	267	90	266	53m 26s	10	12.88%	257	124	228	47m 46s
MIN		2.70%			19m 47s		5.22%				20m 51s	
MEDIAN		6.77%			1h 13m 31s		7.00%				47m 28s	
MAX		8.84%			2h 20m 23s		12.88%				3h 25m 21s	
CAUCASIAN PRIORS BIAS?	11	2.18%	106	21	53	2h 32m 44s	11	2.92%	86	26	69	2h 26m 20s
	7	3.66%	105	38	55	18m 26s	11	6.95%	108	33	71	15m 29s
	11	2.73%	100	32	57	39m 5s	14	4.43%	69	12	51	1h 47m 5s
	17	2.19%	101	28	57	16h 19m 14s	7	3.40%	83	21	82	20m 1s
	19	3.17%	86	30	53	52h 10m 2s	13	3.09%	96	24	58	1h 8m 4s
	11	2.45%	94	26	52	2h 18m 42s	14	5.79%	99	45	87	1h 51m 2s
	15	3.94%	87	29	52	2h 39m 18s	17	5.10%	110	73	94	17h 48m 22s
	15	5.36%	90	35	89	3h 41m 16s	14	3.99%	97	38	65	1h 21m 8s
MIN		2.18%			18m 26s		2.92%				15m 29s	
MEDIAN		2.95%			2h 36m 1s		4.21%				1h 34m 7s	
MAX		5.36%			52h 10m 2s		6.95%				17h 48m 22s	



19 inputs
4 HL * 5 N
3 classes
55% | 56% accuracy

Scalability wrt Neural Network Size

Adult Census Dataset (L = 0.5)



23 inputs
2 HL * 5 N
2 classes

23 inputs
4 HL * 3 N
2 classes

23 inputs
4 HL * 5 N
2 classes

23 inputs
4 HL * 10 N
2 classes

23 inputs
9 HL * 5 N
2 classes

M	U	BOXES				SYMBOLIC				DEEPPOLY			
		INPUT	C	F	TIME	INPUT	C	F	TIME	INPUT	C	F	TIME
10 ○ ● ⊕	4	88.26%	1482	77 1136	33m 55s	95.14%	1132	65 686	19m 5s	93.99%	1894	77 992	29m 55s
	6	99.51%	769	51 723	1h 10m 25s	99.93%	578	47 447	39m 8s	99.83%	1620	54 1042	1h 24m 24s
	8	100.00%	152	19 143	3h 47m 23s	100.00%	174	18 146	1h 51m 2s	100.00%	1170	26 824	8h 2m 27s
	10	100.00%	1	1 1	55m 58s	100.00%	1	1 1	56m 8s	100.00%	1	1 1	56m 43s
12 △ ▲ 人	4	49.83%	719	9 329	13m 43s	72.29%	1177	11 559	24m 9s	60.52%	1498	14 423	10m 32s
	6	72.74%	1197	15 929	2h 6m 49s	98.54%	333	7 195	20m 46s	66.46%	1653	17 594	15m 44s
	8	98.68%	342	9 284	1h 46m 43s	98.78%	323	9 190	1h 27m 18s	70.87%	1764	18 724	2h 19m 11s
	10	99.06%	313	7 260	1h 21m 47s	99.06%	307	5 182	1h 13m 55s	80.76%	1639	18 1007	3h 22m 11s
20 ◇ ◆ ㄧ	4	38.92%	1044	18 39	2m 6s	51.01%	933	31 92	15m 28s	49.62%	1081	34 79	3m 2s
	6	46.22%	1123	62 255	20m 51s	61.60%	916	67 405	44m 40s	59.20%	1335	90 356	22m 13s
	8	64.24%	1111	96 792	2h 24m 51s	74.27%	1125	78 780	3h 26m 20s	69.69%	1574	127 652	5h 6m 7s
	10	85.90%	1390	71 1339	>13h	89.27%	1435	60 1157	>13h	76.25%	1711	148 839	4h 36m 23s
40 □ ■ ㄧ	4	0.35%	10	0 0	1m 39s	34.62%	768	1 1	6m 56s	26.39%	648	2 3	10m 11s
	6	0.35%	10	0 0	1m 38s	34.76%	817	4 5	43m 53s	26.74%	592	8 10	1h 23m 11s
	8	0.42%	12	1 2	14m 37s	35.56%	840	21 28	2h 48m 15s	27.74%	686	32 42	2h 43m 2s
	10	0.80%	23	10 13	1h 48m 43s	37.19%	880	50 75	11h 32m 21s	30.56%	699	83 121	>13h
45 ◇ ㄧ *	4	1.74%	50	0 0	1m 38s	41.98%	891	14 49	10m 14s	36.60%	805	6 8	2m 47s
	6	2.50%	72	3 22	4m 35s	45.00%	822	32 143	45m 42s	38.06%	847	25 50	5m 7s
	8	9.83%	282	25 234	25m 30s	47.78%	651	46 229	1h 14m 5s	42.53%	975	74 180	25m 1s
	10	18.68%	522	33 488	1h 51m 24s	49.62%	714	51 294	3h 23m 20s	48.68%	1087	110 373	1h 58m 34s

Scalability wrt Queried Input Space Size

Adult Census Dataset (L = 0.25, U = 0.1 * IMI)



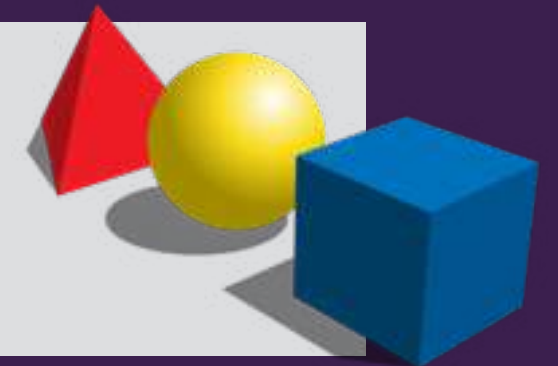
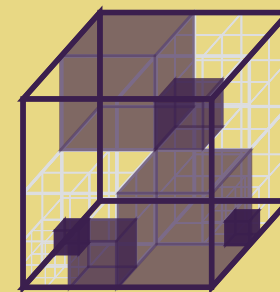
M	QUERY	BOXES				SYMBOLIC				DEEPPOLY						
		INPUT	C	F		TIME	INPUT	C	F		TIME	INPUT	C	F		TIME
80	F 0.009%	99.931% 0.009%	11	0	0	3m 5s	99.961% 0.009%	17	0	0	3m 2s	99.957% 0.009%	10	0	0	2m 36s
	E 0.104%	99.583% 0.104%	61	0	0	3m 6s	99.783% 0.104%	89	0	0	3m 10s	99.753% 0.104%	74	0	0	2m 44s
	D 1.042%	97.917% 1.020%	151	0	0	2m 56s	99.258% 1.034%	297	0	0	3m 41s	98.984% 1.031%	477	0	0	2m 58s
	C 8.333%	83.503% 6.958%	506	2	3	2h 1m	95.482% 7.956%	885	25	34	>13h	93.225% 7.768%	1145	23	33	12h 57m 37s
	B 50%	25.634% 12.817%	5516	7	11	1h 28m 6s	76.563% 38.281%	4917	123	182	>13h	63.906% 31.953%	7139	117	152	>13h
	A 100%	0.052% 0.052%	12	0	0	25m 51s	61.385% 61.385%	5156	73	102	10h 25m 2s	43.698% 43.698%	4757	68	88	>13h
320	F 0.009%	99.931% 0.009%	6	0	0	3m 15s	99.944% 0.009%	9	0	0	3m 35s	99.931% 0.009%	6	0	0	3m 30s
	E 0.104%	99.583% 0.104%	121	0	0	3m 39s	99.627% 0.104%	120	0	0	6m 34s	99.583% 0.104%	31	0	0	4m 22s
	D 1.042%	97.917% 1.020%	151	0	0	6m 18s	98.247% 1.024%	597	0	0	21m 9s	97.917% 1.020%	301	0	0	9m 35s
	C 8.333%	83.333% 6.944%	120	0	0	30m 37s	88.294% 7.358%	755	0	0	1h 36m 35s	83.342% 6.945%	483	0	0	52m 29s
	B 50%	25.000% 12.500%	5744	0	0	2h 24m 36s	46.063% 23.032%	4676	0	0	7h 25m 57s	25.074% 12.537%	5762	4	4	>13h
	A 100%	0.000% 0.000%	0	0	0	2h 54m 25s	24.258% 24.258%	2436	0	0	9h 41m 36s	0.017% 0.017%	4	0	0	5h 3m 33s
1280	F 0.009%	99.931% 0.009%	11	0	0	7m 35s	99.948% 0.009%	10	0	0	24m 42s	99.931% 0.009%	6	0	0	7m 6s
	E 0.104%	99.583% 0.104%	31	0	0	15m 49s	99.674% 0.104%	71	0	0	51m 52s	99.583% 0.104%	31	0	0	15m 14s
	D 1.042%	97.917% 1.020%	151	0	0	1h 49s	98.668% 1.028%	557	0	0	3h 31m 45s	97.917% 1.020%	301	0	0	1h 3m 33s
	C 8.333%	83.333% 6.944%	481	0	0	7h 11m 39s	—	—	—	—	>13h	83.333% 6.944%	481	0	0	7h 12m 57s
	B 50%	—	—	—	—	>13h	—	—	—	—	>13h	—	—	—	—	>13h
	A 100%	—	—	—	—	>13h	—	—	—	—	>13h	—	—	—	—	>13h

practical tools
targeting specific programs

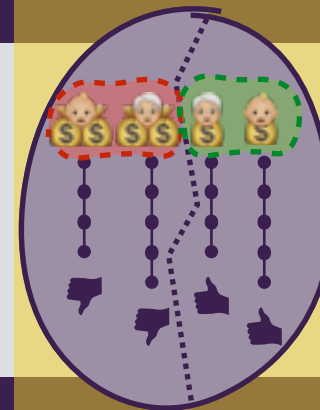
[M]	QUERY	INPUT	[C]	[F]	TIME	INPUT	[C]	[F]	TIME	INPUT	[C]	[F]	TIME			
80	F	99.9315	11	0	0	3m 5s	99.9615	17	0	0	3m 2s	99.9575	10	0	0	2m 4s
	E	99.5835	61	0	0	3m 6s	99.7835	89	0	0	3m 10s	99.7335	74	0	0	2m 4s
	D	97.9175	151	0	0	2m 5s	99.2585	297	0	0	3m 41s	98.9845	477	0	0	2m 5s
	C	83.5635	506	2	3	2h 1m	95.4825	885	25	34	+13h	93.2235	1145	23	33	12h 57m 37s
	B	25.6345	5516	7	11	1h 28m 6s	76.5635	4917	123	182	+13h	63.9065	7139	117	152	+13h
	A	0.6525	12	0	0	25m 51s	61.3855	5136	73	102	10h 25m 2s	43.6985	4757	68	88	+13h
	F	99.9315	6	0	0	3m 15s	99.9445	9	0	0	3m 35s	99.9315	6	0	0	3m 30s
	E	99.5835	121	0	0	3m 39s	99.6275	120	0	0	6m 34s	99.5835	31	0	0	4m 22s
	D	97.9175	151	0	0	6m 18s	98.2475	597	0	0	21m 9s	97.9175	301	0	0	9m 35s
	C	83.3335	120	0	0	30m 37s	88.2945	755	0	0	1h 36m 35s	85.3425	483	0	0	52m 29s
320	B	25.0005	5744	0	0	2h 24m 3s	46.9635	4676	0	0	7h 25m 57s	25.0745	5762	4	4	+13h
	A	0.0005	0	0	0	2h 54m 25s	24.2585	2436	0	0	9h 41m 36s	0.0175	4	0	0	5h 3m 33s
	F	99.9315	11	0	0	7m 35s	99.9445	10	0	0	24m 42s	99.9315	6	0	0	7m 6s
	E	99.5835	31	0	0	15m 49s	99.6745	71	0	0	51m 52s	99.5835	31	0	0	15m 14s
	D	97.9175	151	0	0	1h 49s	98.6685	537	0	0	3h 31m 45s	97.9175	301	0	0	1h 3m 33s
	C	83.3335	481	0	0	7h 11m 39s	--	--	--	+13h	83.3335	481	0	0	7h 12m 57s	
	B	--	--	--	--	+13h	--	--	--	+13h	--	--	--	--	+13h	
	A	--	--	--	--	+13h	--	--	--	+13h	--	--	--	--	+13h	
	F	99.9315	11	0	0	7m 35s	99.9445	10	0	0	24m 42s	99.9315	6	0	0	7m 6s
	E	99.5835	31	0	0	15m 49s	99.6745	71	0	0	51m 52s	99.5835	31	0	0	15m 14s
1280	D	97.9175	151	0	0	1h 49s	98.6685	537	0	0	3h 31m 45s	97.9175	301	0	0	1h 3m 33s
	C	83.3335	481	0	0	7h 11m 39s	--	--	--	+13h	83.3335	481	0	0	7h 12m 57s	
	B	--	--	--	--	+13h	--	--	--	+13h	--	--	--	--	+13h	
	A	--	--	--	--	+13h	--	--	--	+13h	--	--	--	--	+13h	
	F	99.9315	11	0	0	7m 35s	99.9445	10	0	0	24m 42s	99.9315	6	0	0	7m 6s
	E	99.5835	31	0	0	15m 49s	99.6745	71	0	0	51m 52s	99.5835	31	0	0	15m 14s
	D	97.9175	151	0	0	1h 49s	98.6685	537	0	0	3h 31m 45s	97.9175	301	0	0	1h 3m 33s
	C	83.3335	481	0	0	7h 11m 39s	--	--	--	+13h	83.3335	481	0	0	7h 12m 57s	
	B	--	--	--	--	+13h	--	--	--	+13h	--	--	--	--	+13h	
	A	--	--	--	--	+13h	--	--	--	+13h	--	--	--	--	+13h	



algorithmic approaches
to decide program properties



mathematical models
of the program behavior



<https://github.com/caterinaurban/Libra>



QUESTIONS?