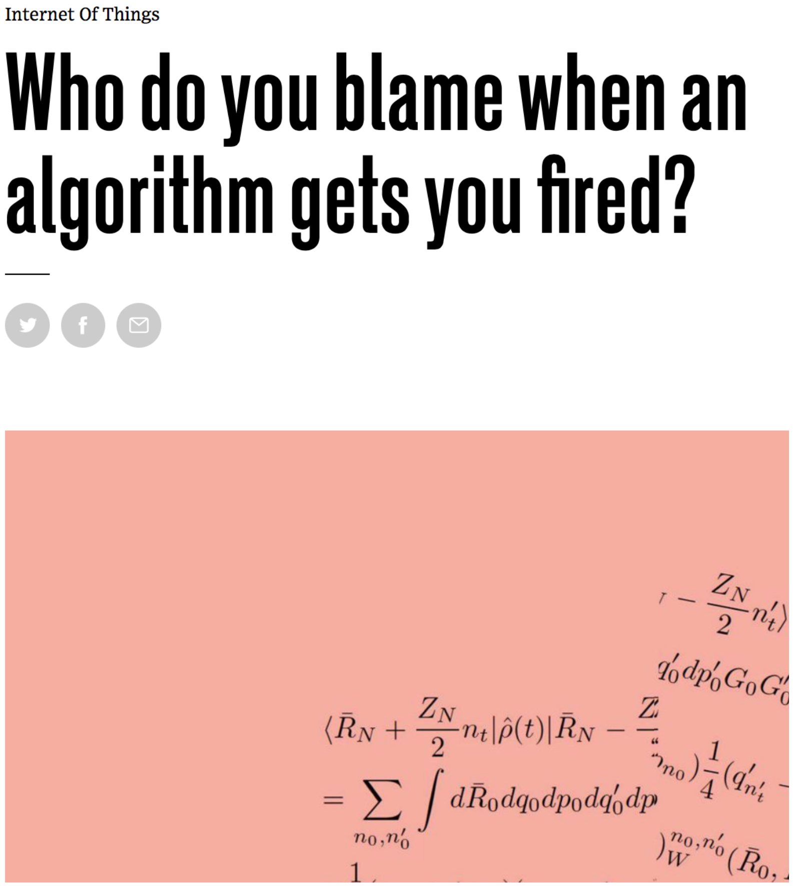
Fairness as a Program Property

Aws Albarghouthi
Loris D’Antoni
Samuel Drews
University of Wisconsin-Madison

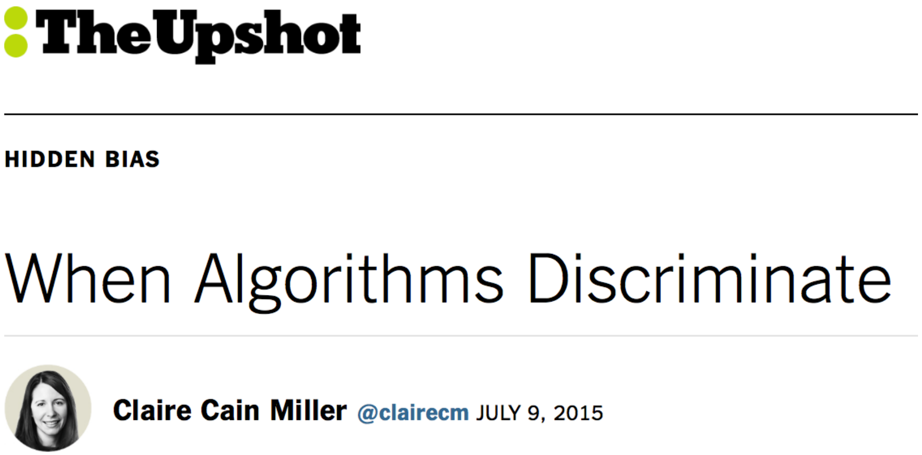
Aditya Nori
Microsoft Research



Internet Of Things

Who do you blame when an algorithm gets you fired?

$\langle \bar{R}_N + \frac{Z_N}{2} n_t | \hat{\rho}(t) | \bar{R}_N - \frac{Z_N}{2} n_t' \rangle$
$= \sum_{n_0, n_0'} \int d\bar{R}_0 dq_0 dp_0 dq_0' dp_0' G_0 G_0'$
$\frac{1}{4}(q_{n_t}'$
$)_W^{n_0, n_0'}(\bar{R}_0,$

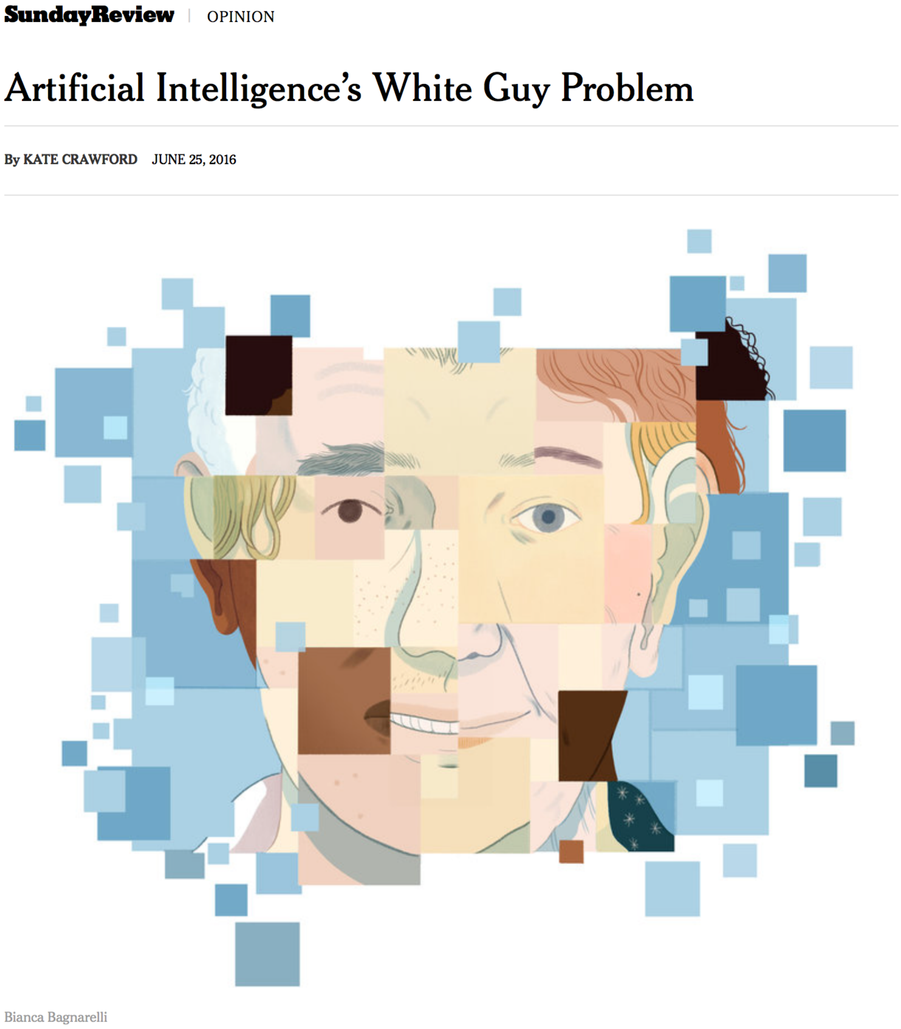


TheUpshot

HIDDEN BIAS

When Algorithms Discriminate

Claire Cain Miller @clairecm JULY 9, 2015



SundayReview | OPINION

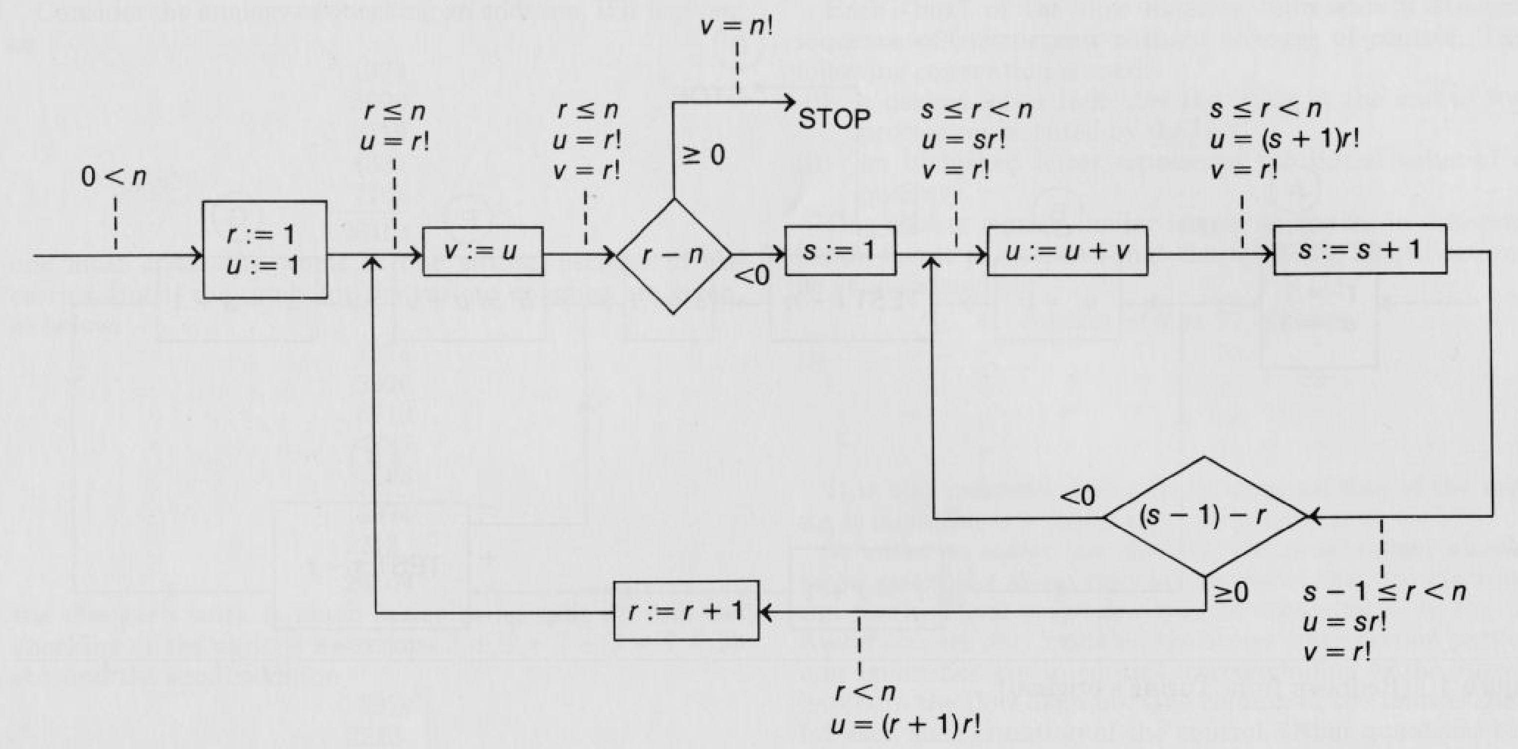
Artificial Intelligence's White Guy Problem

By KATE CRAWFORD JUNE 25, 2016

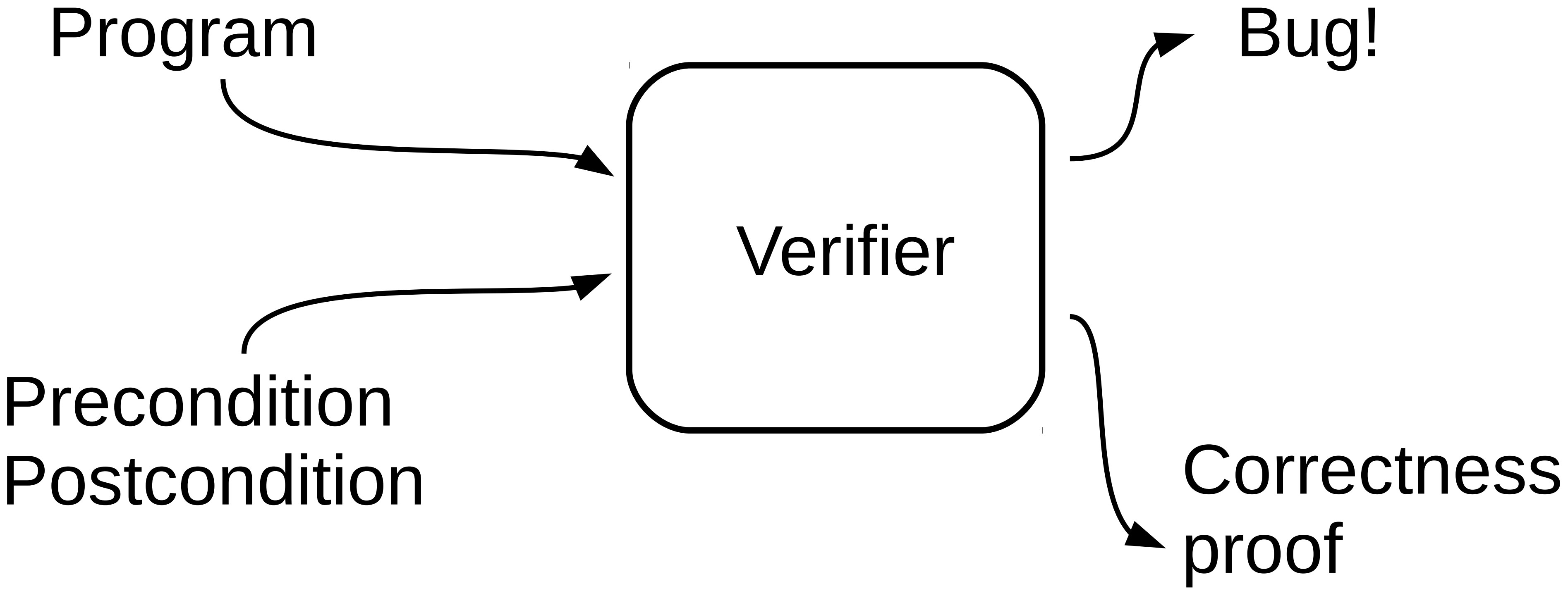
Bianca Bagnarelli

Proof of correctness

Precondition {$n > 0$}



Postcondition {$r = n!$}



Program

Precondition
Postcondition

Verifier

Bug!

Correctness
proof

Group Fairness

$$h \leftarrow \mathcal{D}(\boldsymbol{v})$$

Group Fairness

$$\{\boldsymbol{v} = (v_1, \ldots, v_s, \ldots)\}$$

sensitive feature (e.g. minority)

$$h \leftarrow \mathcal{D}(\boldsymbol{v})$$

Group Fairness

$$\{\boldsymbol{v} = (v_1, \ldots, v_s, \ldots)\}$$

sensitive feature (e.g. minority)

$$h \leftarrow \mathcal{D}(\boldsymbol{v})$$

$$\left\{ \frac{\Pr[h \mid v_s]}{\Pr[h \mid \neg v_s]} > 1 - \epsilon \right\}$$

Group Fairness

population model

$$\{v \sim \mathcal{M}\}$$

$$h \leftarrow \mathcal{D}(\boldsymbol{v})$$

$$\left\{ \frac{\Pr[h \mid v_s]}{\Pr[h \mid \neg v_s]} > 1 - \epsilon \right\}$$

Individual Fairness

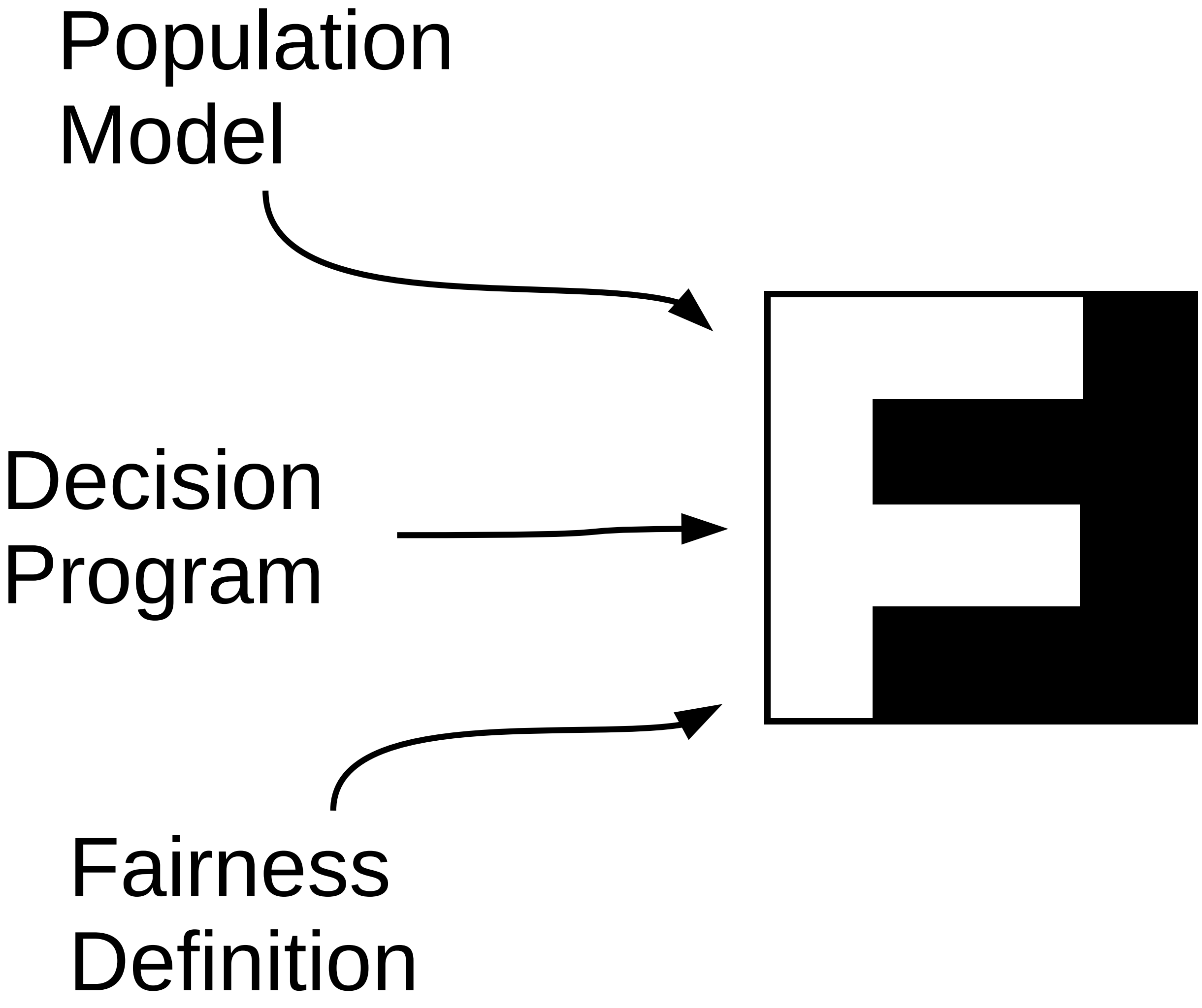
$$\{v_1, v_2 \sim \mathcal{M}\}$$

$$h_1 \leftarrow \mathcal{D}(v_1)$$

$$h_2 \leftarrow \mathcal{D}(v_2)$$

$$\{\Pr[h_1 \neq h_2 \mid v_1 \underset{\substack{\uparrow \\ \text{similarity}}}{\sim} v_2] < \epsilon\}$$

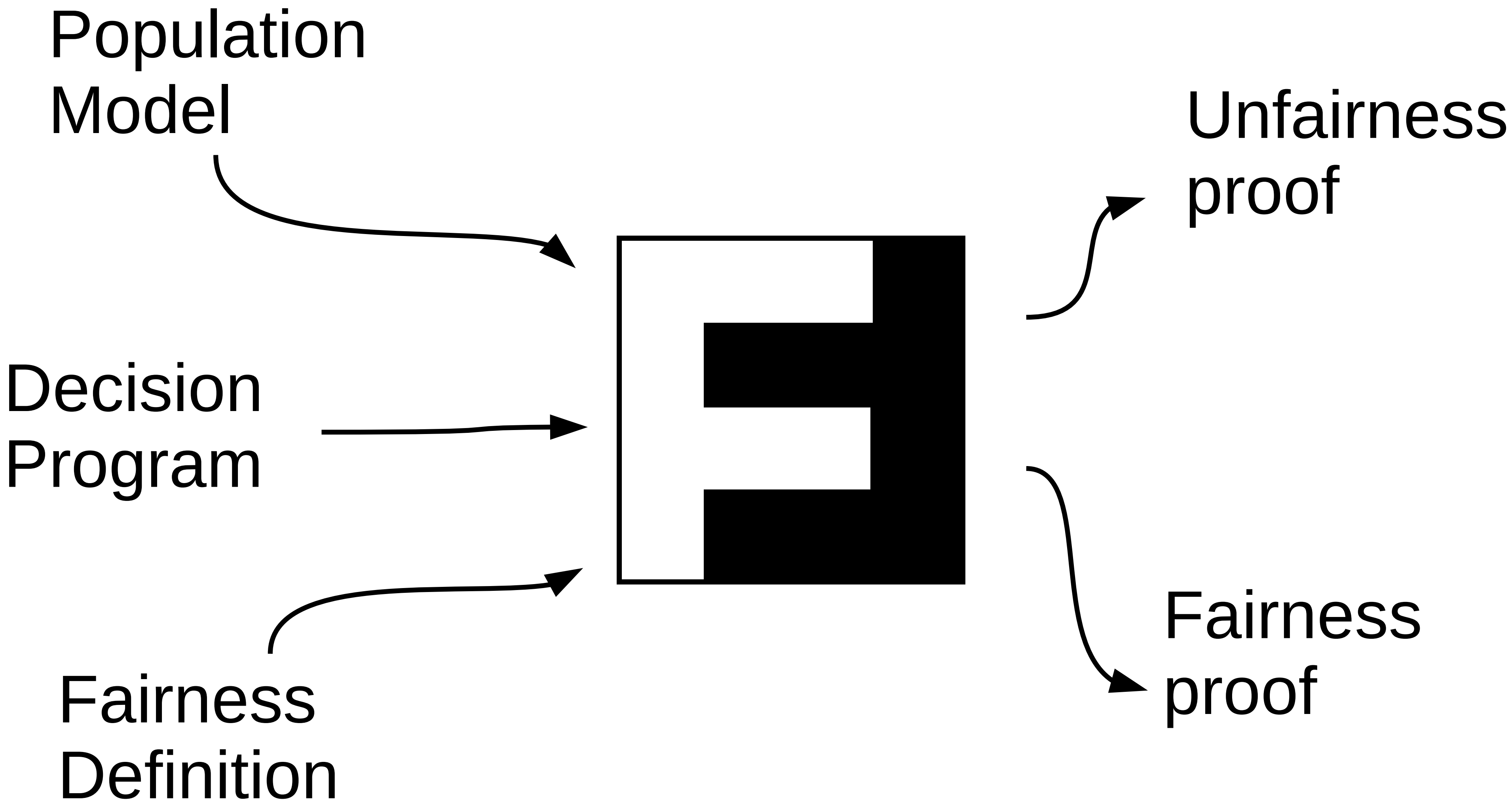
FairSquare



Population Model

Decision Program

Fairness Definition



Population Model

Decision Program

Fairness Definition

Unfairness proof

Fairness proof

$$\{v \sim \mathcal{M}\}$$

```
define dec(colRank, yExp)
  expRank ← yExp - colRank
  if (colRank <= 5)
    hire ← true
  elif (expRank > -5)
    hire ← true
  else
    hire ← false
  return hire
```

$$\left\{ \frac{\Pr[\texttt{hire} \mid \texttt{ethnicity > 10}]}{\Pr[\texttt{hire} \mid \texttt{ethnicity <= 10}]} > 1 - \epsilon \right\}$$

$$\{v \sim \mathcal{M}\}$$

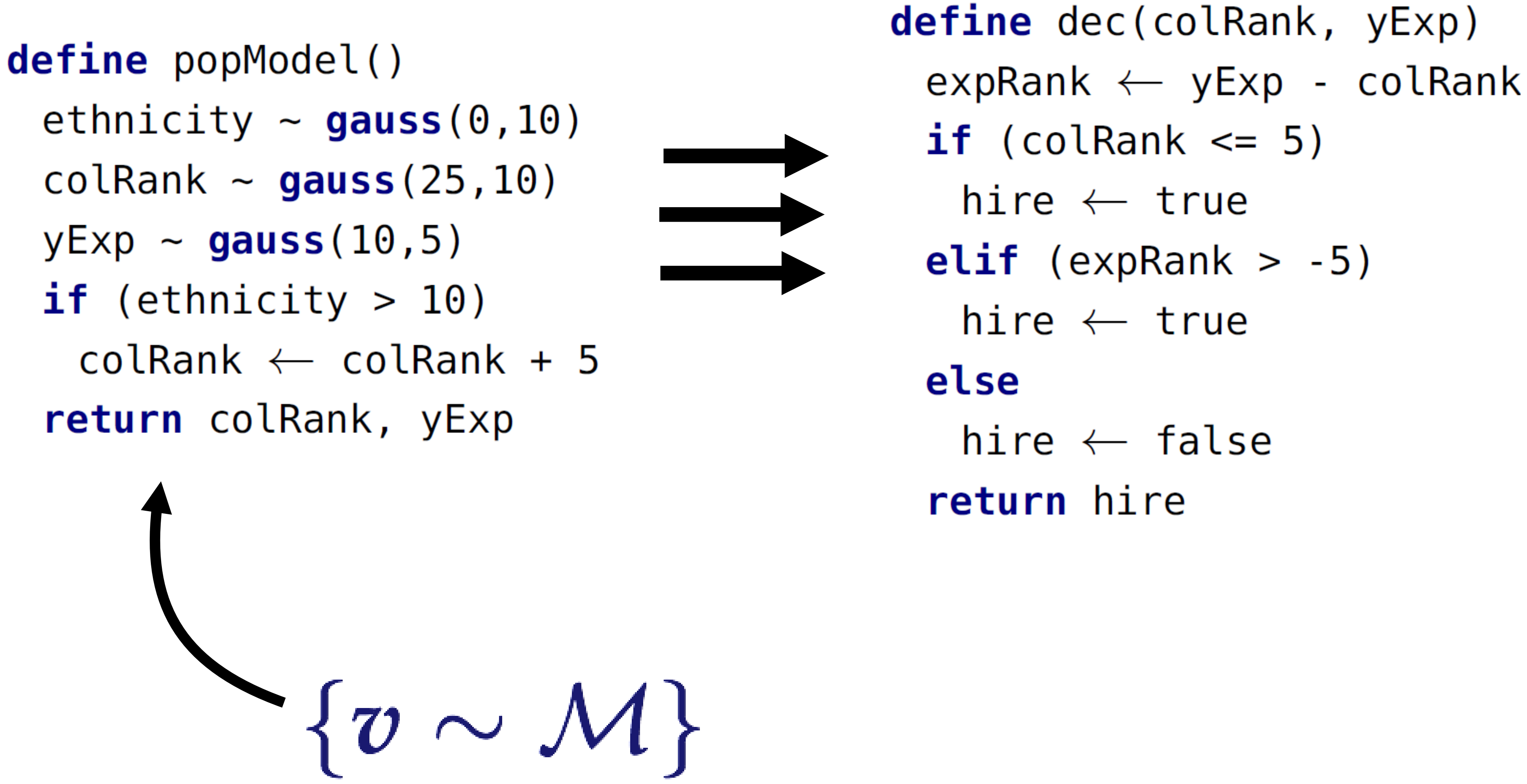
```
define dec(colRank, yExp)
  expRank ← yExp - colRank
  if (colRank <= 5)
    hire ← true
  elif (expRank > -5)
    hire ← true
  else
    hire ← false
  return hire
```

Code!

$$\left\{ \frac{\Pr[\texttt{hire} \mid \texttt{ethnicity > 10}]}{\Pr[\texttt{hire} \mid \texttt{ethnicity <= 10}]} > 1 - \epsilon \right\}$$

population model

decision-making program



```
define popModel()
  ethnicity ~ gauss(0,10)
  colRank ~ gauss(25,10)
  yExp ~ gauss(10,5)
  if (ethnicity > 10)
    colRank ← colRank + 5
  return colRank, yExp
```

```
define dec(colRank, yExp)
  expRank ← yExp - colRank
  if (colRank <= 5)
    hire ← true
  elif (expRank > -5)
    hire ← true
  else
    hire ← false
  return hire
```

$\{v \sim \mathcal{M}\}$

population model

```
define popModel()
  ethnicity ~ gauss(0,10)
  colRank ~ gauss(25,10)
  yExp ~ gauss(10,5)
  if (ethnicity > 10)
    colRank ← colRank + 5
  return colRank, yExp
```

$$\{v \sim \mathcal{M}\}$$

decision-making program

```
define dec(colRank, yExp)
  expRank ← yExp - colRank
  if (colRank <= 5)
    hire ← true
  elif (expRank > -5)
    hire ← true
  else
    hire ← false
  return hire
```

```
dec(popModel())
```

`dec(popModel())` $\Pr[\texttt{hire} \wedge \texttt{min}]$

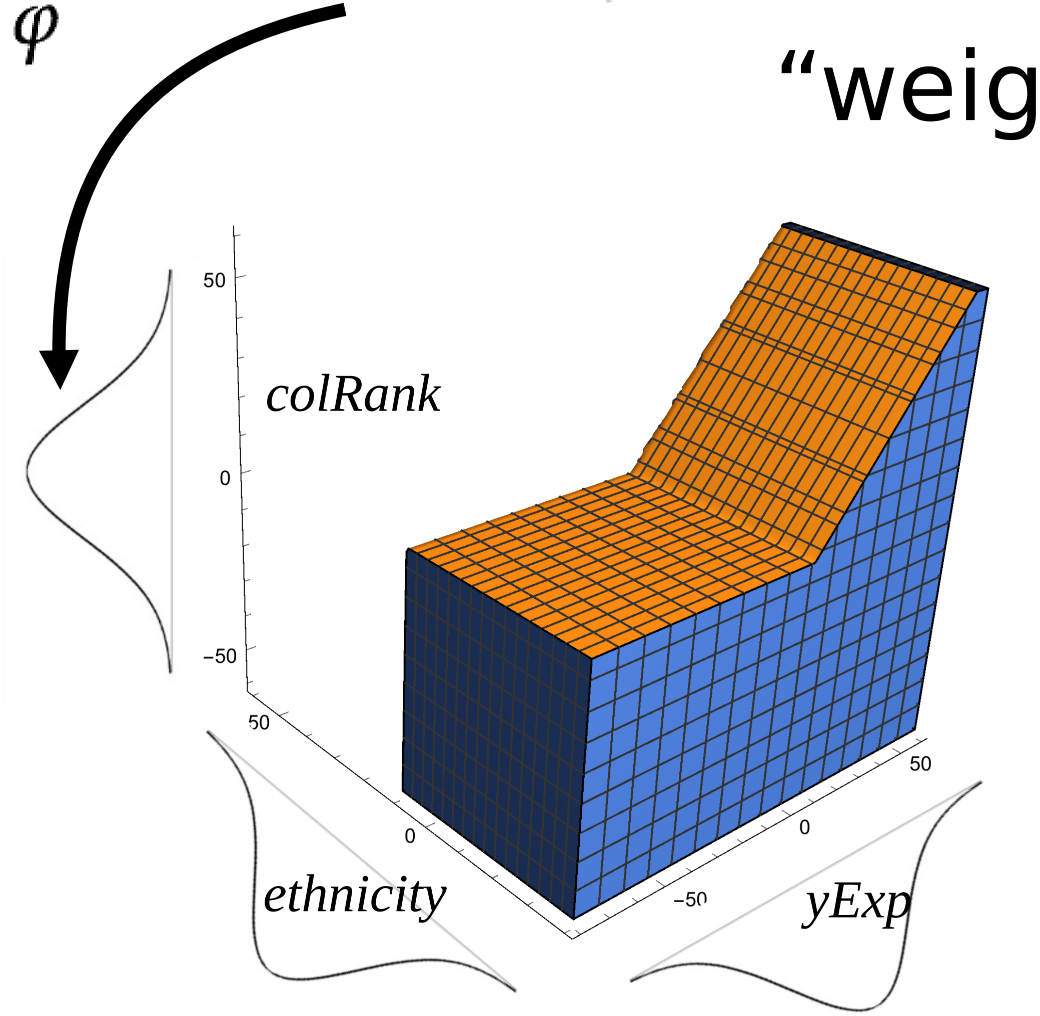
all possible execution paths Π

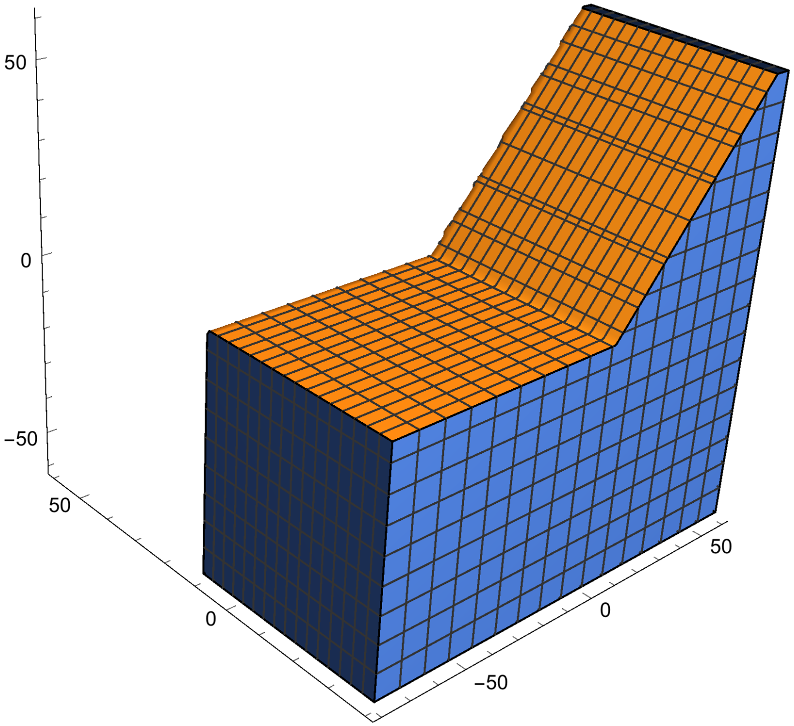
```
define popModel()
 ethnicity ~ gauss(0,10)
 colRank ~ gauss(25,10)
 yExp ~ gauss(10,5)
 if (ethnicity > 10)
  colRank ← colRank + 5
 return colRank, yExp
```

represent paths Π_{hm} as a region $\varphi \subseteq \mathbb{R}^3$

$$\int_{\varphi} p_e(e)p_c(c)p_y(y)\; de\; dp\; dy$$

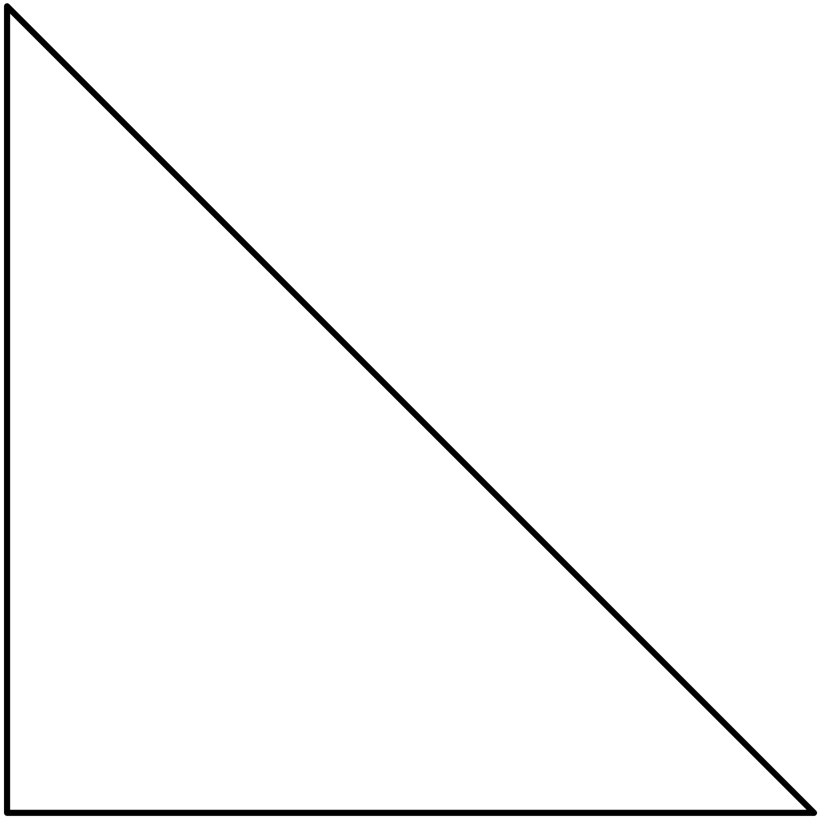
“weighted volume”

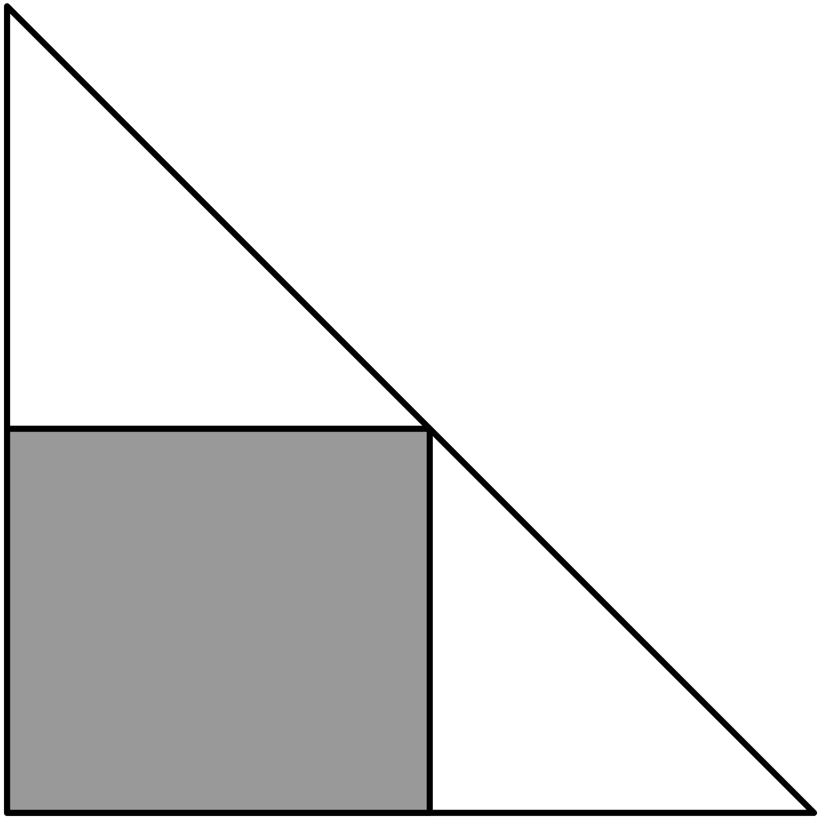


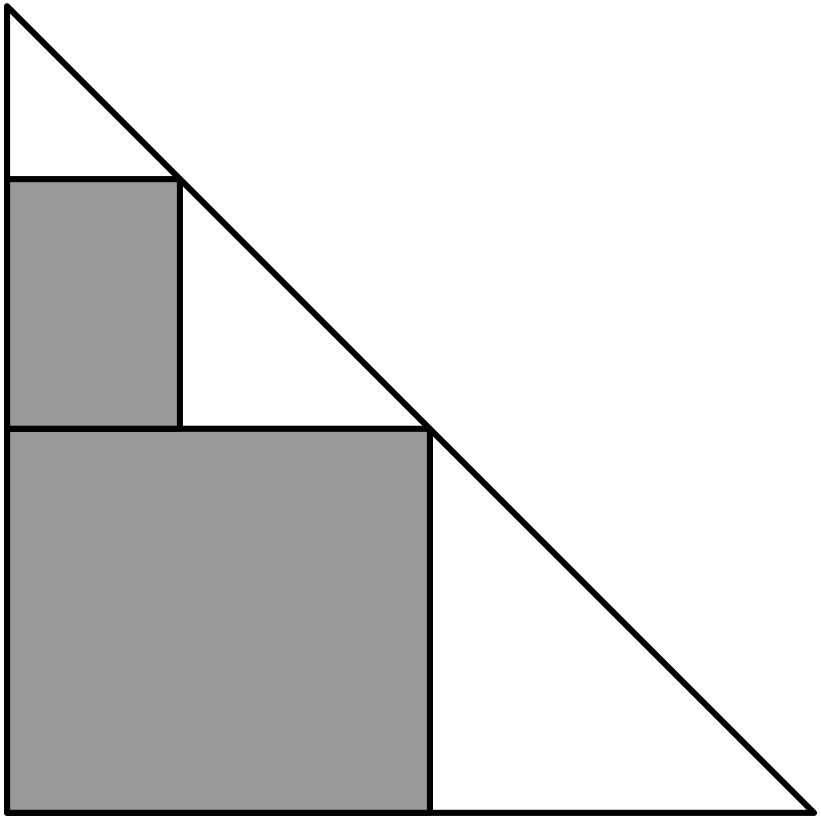


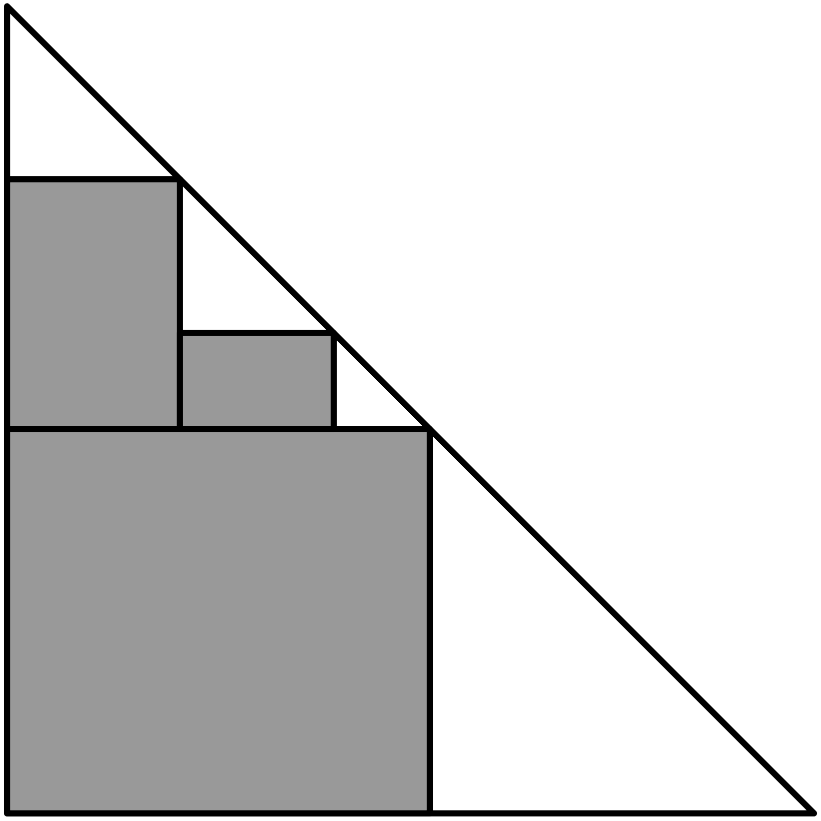
Programming
Languages
Magic

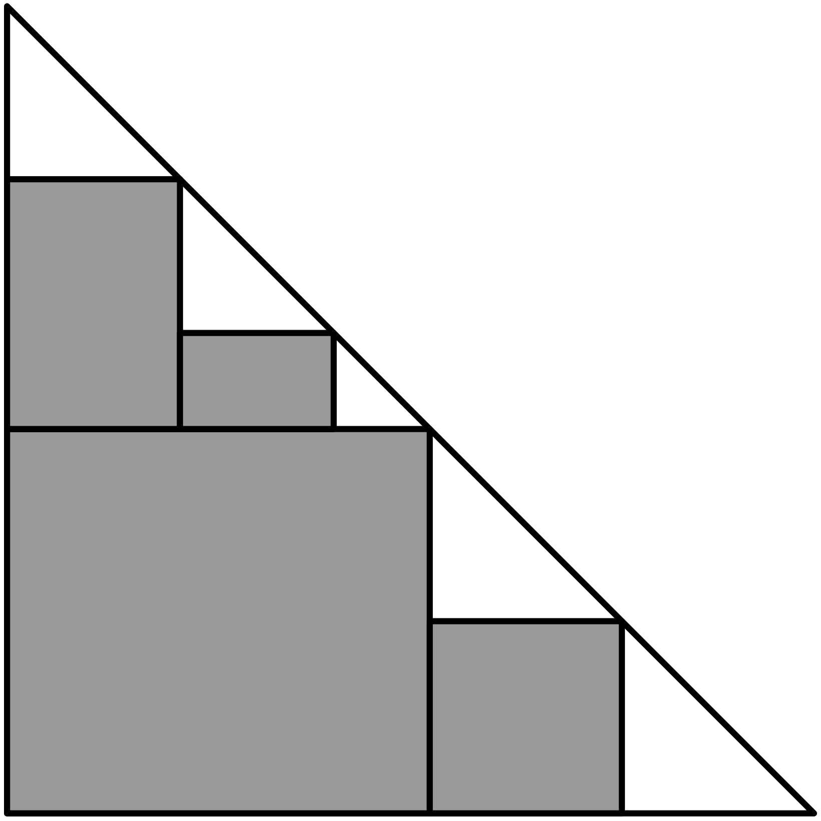
$$\varphi \subseteq \mathbb{R}^3$$







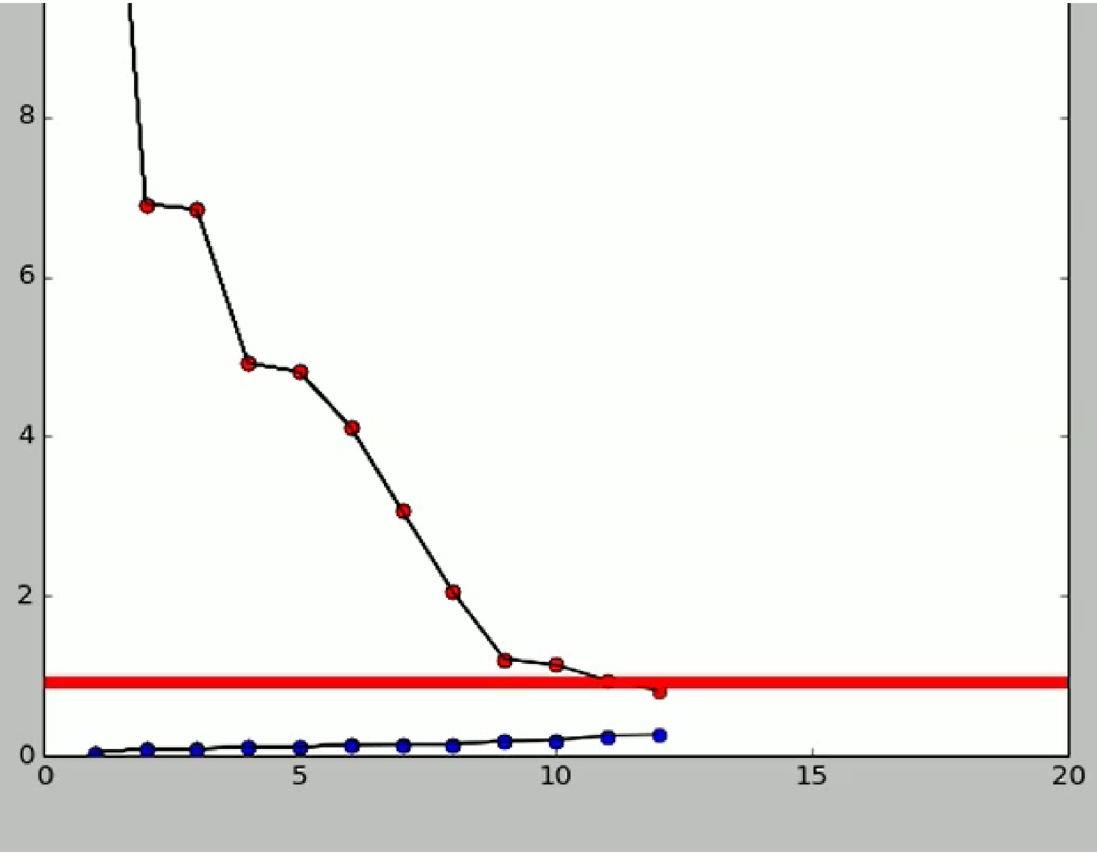


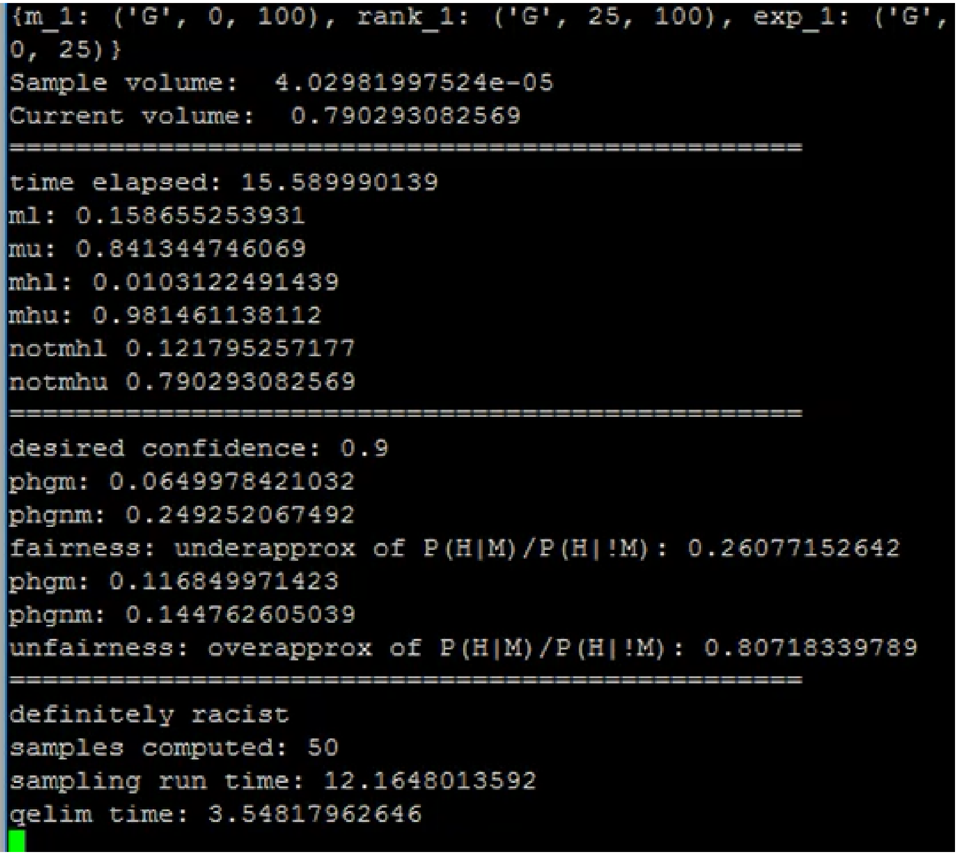


```
{
define popModel()
  ethnicity ~ gauss(0,10)
  colRank ~ gauss(25,10)
  yExp ~ gauss(10,5)
  if (ethnicity > 10)
    colRank ← colRank + 5
  return colRank, yExp
}
```

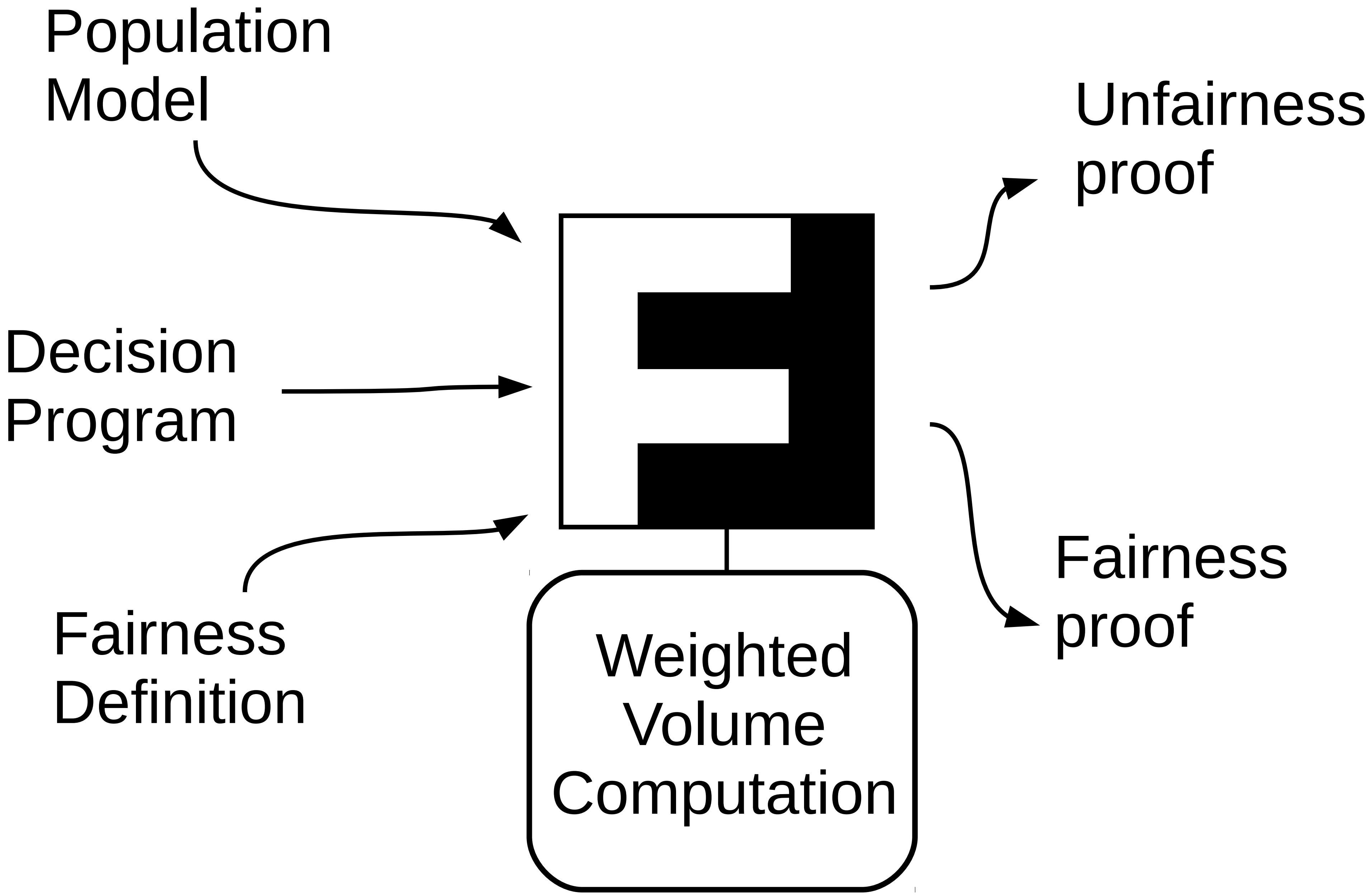
```
define dec(colRank, yExp)
  expRank ← yExp - colRank
  if (colRank <= 5)
    hire ← true
  elif (expRank > -5)
    hire ← true
  else
    hire ← false
  return hire
```

$$\left\{ \frac{\Pr[\texttt{hire} \mid \texttt{ethnicity > 10}]}{\Pr[\texttt{hire} \mid \texttt{ethnicity <= 10}]} > 0.9 \right\}$$





```
{m_1: ('G', 0, 100), rank_1: ('G', 25, 100), exp_1: ('G',
0, 25)}
Sample volume:  4.02981997524e-05
Current volume:  0.790293082569
================================================
time elapsed: 15.589990139
ml: 0.158655253931
mu: 0.841344746069
mhl: 0.0103122491439
mhu: 0.981461138112
notmhl 0.121795257177
notmhu 0.790293082569
================================================
desired confidence: 0.9
phgm: 0.0649978421032
phgnm: 0.249252067492
fairness: underapprox of P(H|M)/P(H|!M): 0.26077152642
phgm: 0.116849971423
phgnm: 0.144762605039
unfairness: overapprox of P(H|M)/P(H|!M): 0.80718339789
================================================
definitely racist
samples computed: 50
sampling run time: 12.1648013592
qelim time: 3.54817962646
```



Population Model

Decision Program

Fairness Definition

Weighted Volume Computation

Unfairness proof

Fairness proof