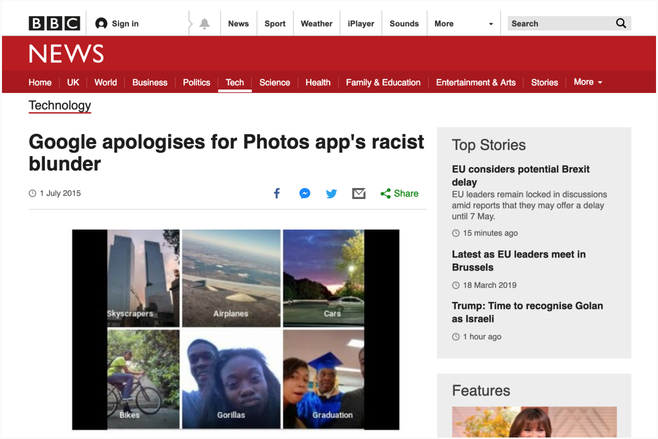
Evolutionary dataset optimisation: learning algorithm quality through evolution

Henry Wilde, Dr. Jonathan Gillard, Dr. Vincent Knight



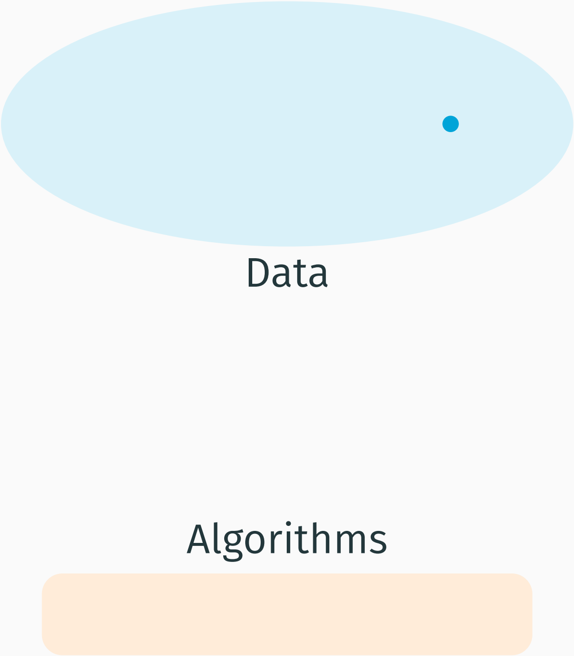
via: BBC News (`https://www.bbc.co.uk/news/technology-33347866`)

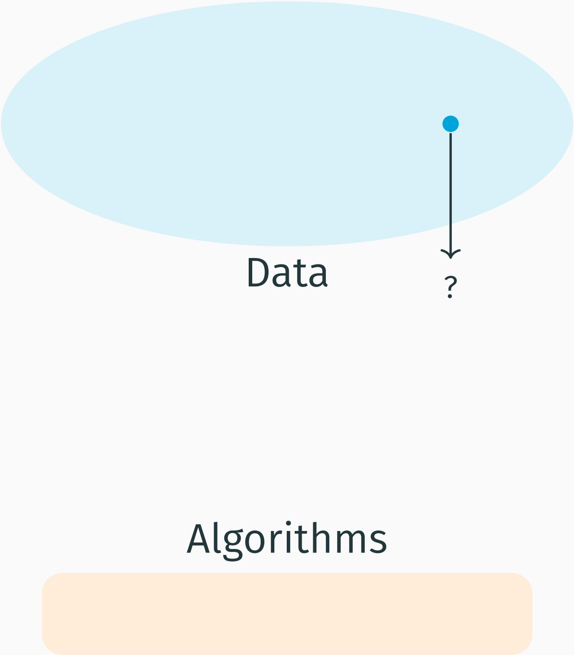
Reliability and frailty

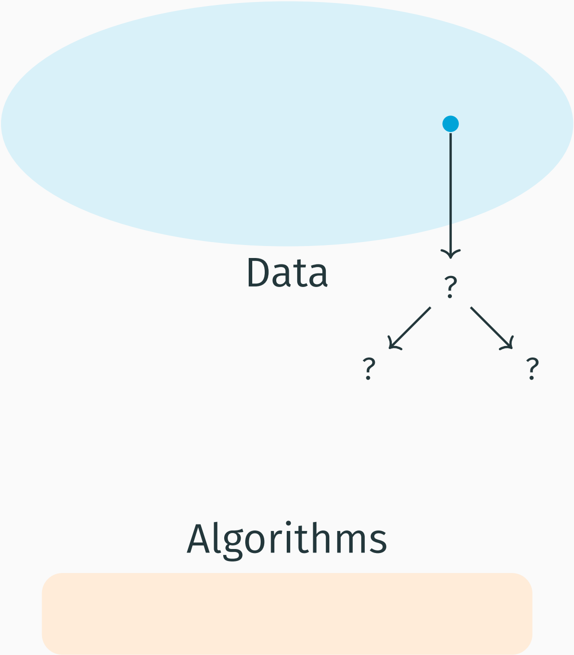
- R. Hyndman. *Prediction competitions*. 2014. URL: `https://robjhyndman.com/hyndsight/prediction-competitions/`
- A. Torralba and A. A. Efros. "Unbiased Look at Dataset Bias". In: *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*. 2011, pp. 1521–1528. DOI: `10.1109/CVPR.2011.5995347`

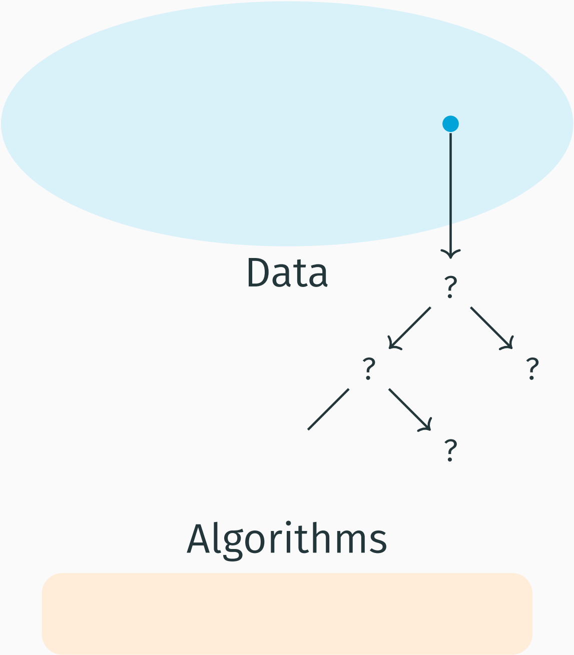
Data

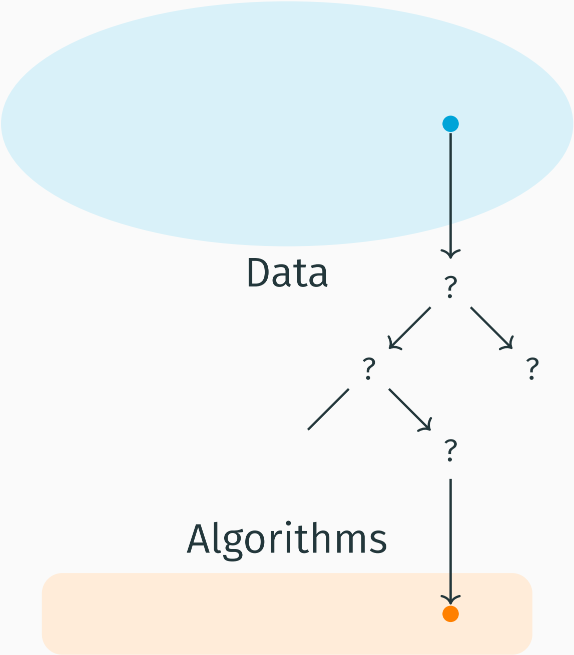
Algorithms

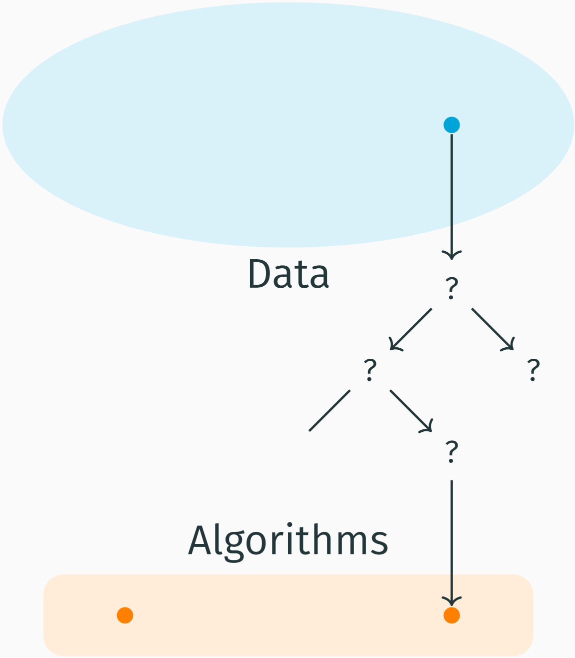

Data
Algorithms

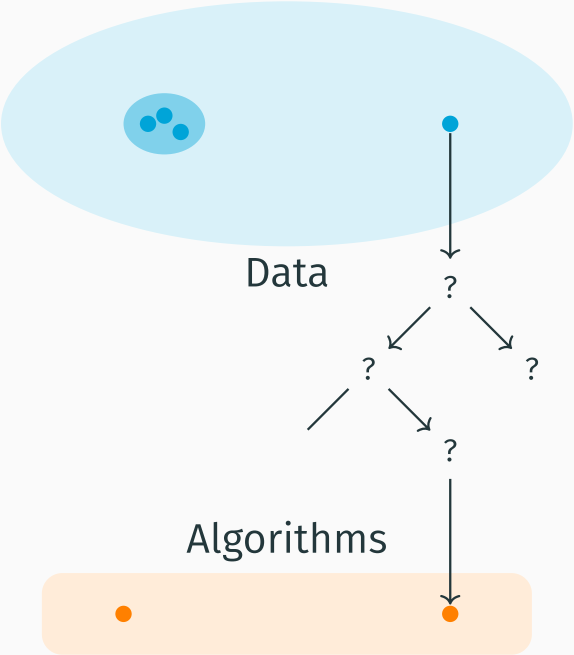

Data
?
Algorithms

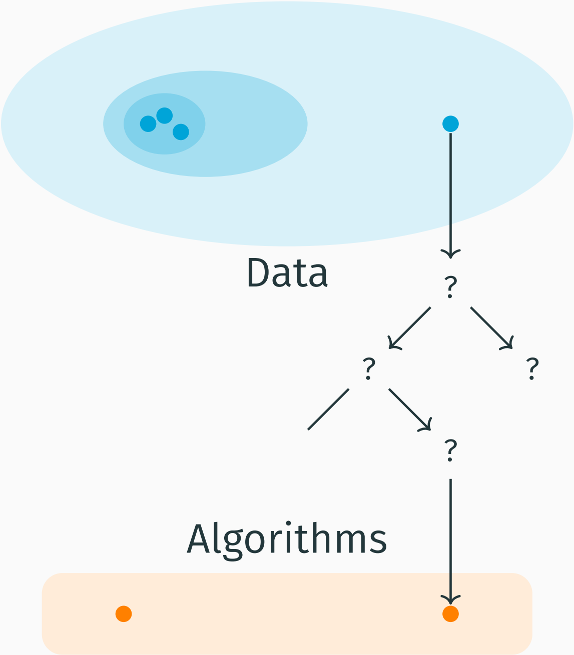

Data
?
?
?
Algorithms

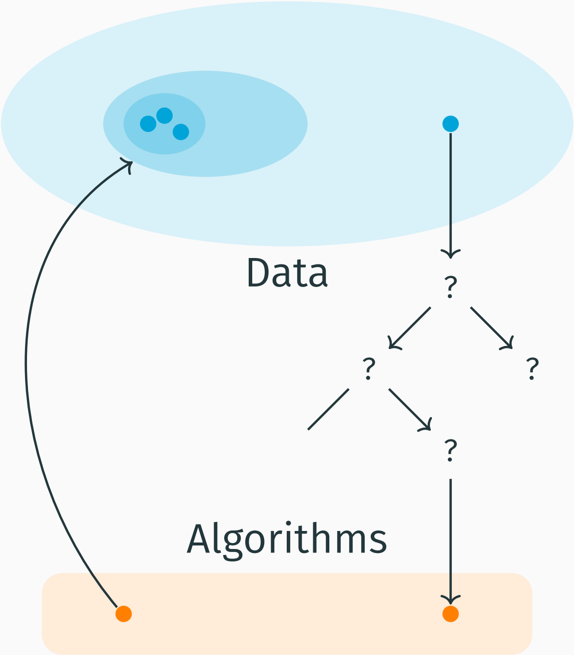

Data
?
?
?
?
Algorithms


Data
?
?
?
?
Algorithms


Data
?
?
?
?
Algorithms


Data
?
?
?
?
Algorithms


Data
?
?
?
?
Algorithms


Data
?
?
?
?
Algorithms

Generating artificial data

via: `https://thispersondoesnotexist.com`

Barney Sparks

@ barneysparks@gmail.com Allentown, Pennsylvania

EXPERIENCE

Operations Analyst

Youth **2014 - Ongoing** Allentown, Pennsylvania

Youth is a leading platform that is developing a product in cloud and in-house platforms.

- Provided support of over 50 international startups and change
- Managed a team of 10 people in a staff of 10 people
- Managed 10 projects with 4 projects per day management of the company and included an average of 200 companies and 2 employees in the first role.
- Increased the company by 100% in 2014 and 2011 and 2015.

Head Of Marketing And Controller

X-Main **2010 - 2014** Allentown, Pennsylvania

X-main is a software company that provides young people with leadership development and enterprise software used in Junico and the construction industry.

- Created 3 consultants to provide internal and staff & maintenance strategies and incorporated the company with timely companies resulting in $2MM in conduct and increased the company to 100 people in less than 3 months

SUMMARY

I am a highly motivated and proven pursuits where my work diverse individual with others into the field of interest and be like interependemy to go to their practice cliniz.

TECHNOLOGIES

PHP | Ansi C | Perl | Oracle BRM Pipeline | Oracle BRM CMT | Java | Shell Script | Linux | Oracle BRM

EDUCATION

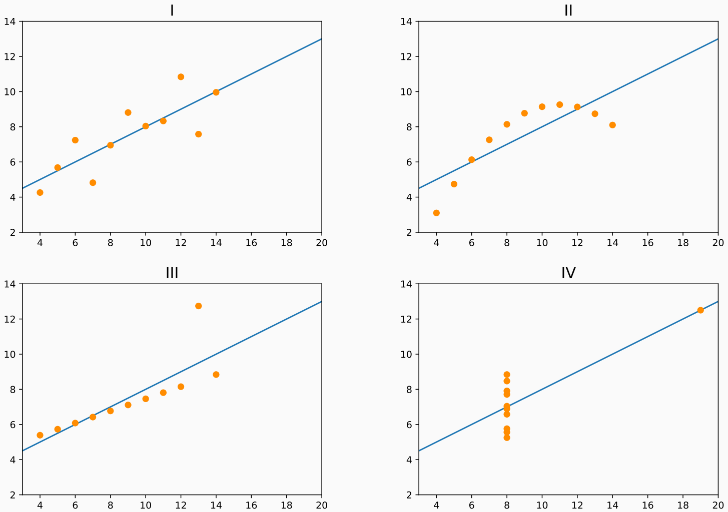
Unige, Faculty Of Medicine

University Of Geneva

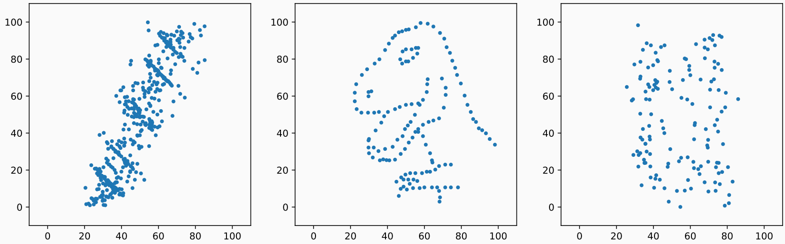
via: https://thisresumedoesnotexist.com

via: `https://thiscatdoesnotexist.com`

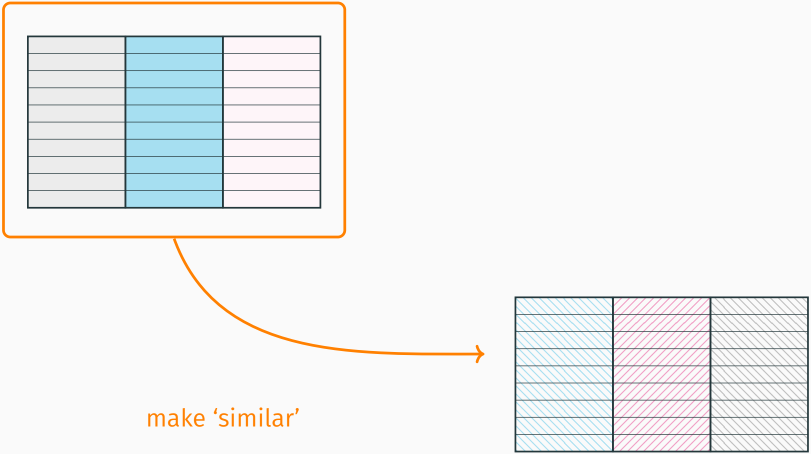
Anscombe’s quartet



The Datasaurus dozen

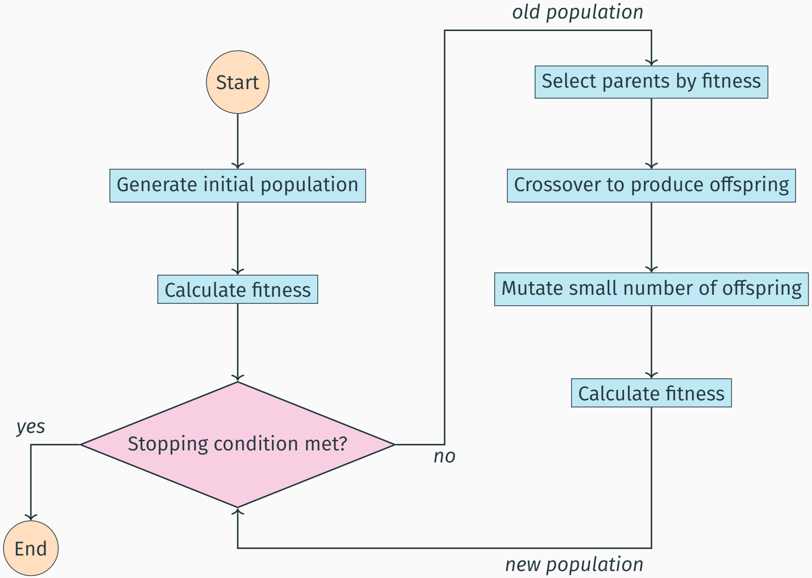


Original paper by @JustinMatejka


make ‘similar’

Given an algorithm, how can one find sets of data for which it performs well?

Evolutionary algorithms


old population
Start
Select parents by fitness
Generate initial population
Crossover to produce offspring
Calculate fitness
Mutate small number of offspring
Calculate fitness
yes
Stopping condition met?
no
End
new population

$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

$$\max \quad f : \mathbb{N}^2 \rightarrow \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

Population (25, 30) (12, 1) (11, 0) (20, 12) (24, 25)

$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49

$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49
Select parents	(25, 30)			(20, 12)	(24, 25)

$$\max \quad f:\mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49
Select parents	(25, 30)			(20, 12)	(24, 25)
Create offspring		(24, 30)	(25, 12)		

$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

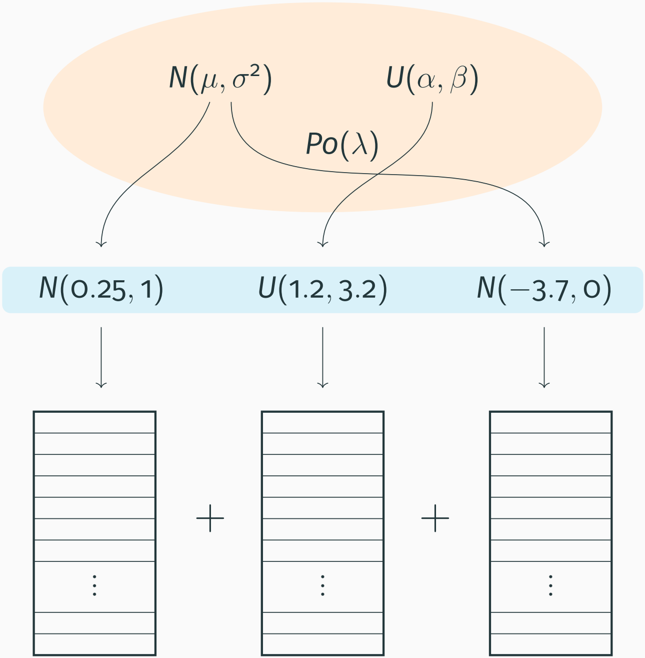
Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49
Select parents	(25, 30)			(20, 12)	(24, 25)
Create offspring		(24, 30)	(25, 12)		
Mutate offspring		(24, 30)	(25, 13)		

$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49
Select parents	(25, 30)			(20, 12)	(24, 25)
Create offspring		(24, 30)	(25, 12)		
Mutate offspring		(24, 30)	(25, 13)		
New generation	(25, 30)	(24, 30)	(25, 13)	(20, 12)	(24, 25)

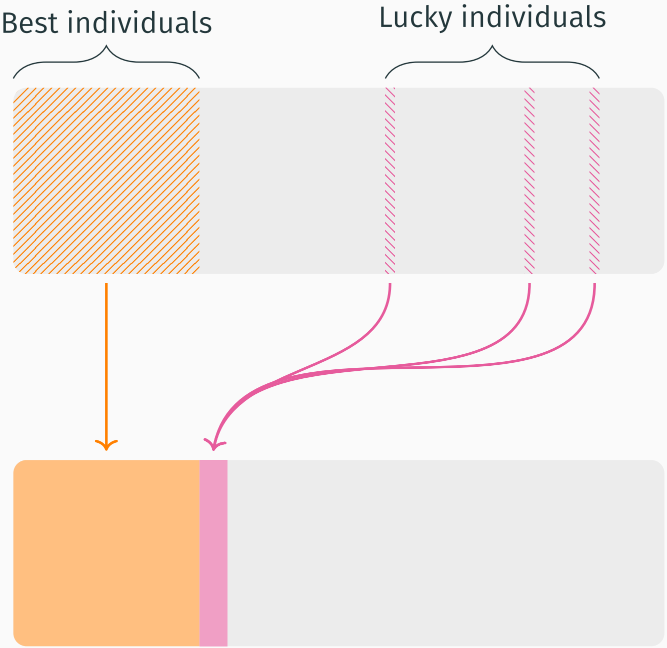
$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49
Select parents	(25, 30)			(20, 12)	(24, 25)
Create offspring		(24, 30)	(25, 12)		
Mutate offspring		(24, 30)	(25, 13)		
New generation	(25, 30)	(24, 30)	(25, 13)	(20, 12)	(24, 25)


$N(\mu, \sigma^2)$
$U(\alpha, \beta)$
$Po(\lambda)$
$N(0.25, 1)$
$U(1.2, 3.2)$
$N(-3.7, 0)$
+
+

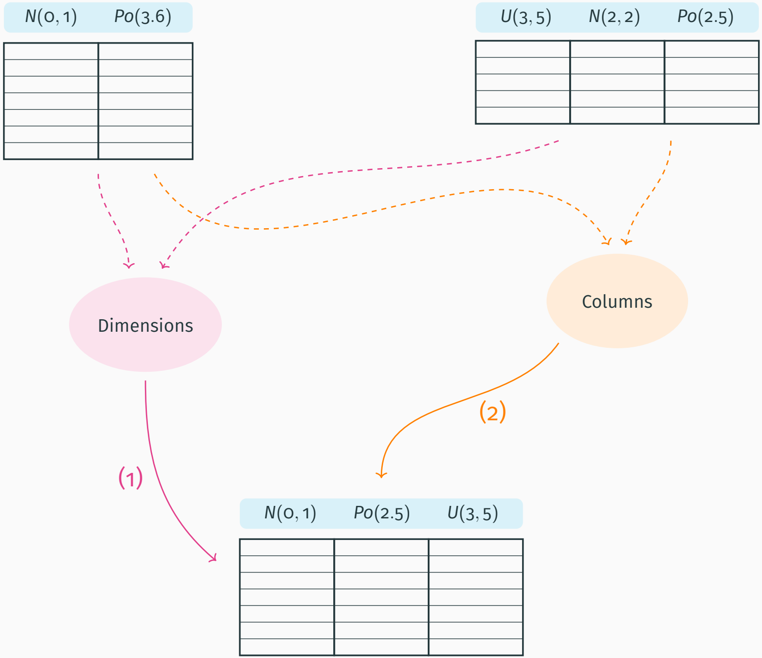
$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49
Select parents	(25, 30)			(20, 12)	(24, 25)
Create offspring		(24, 30)	(25, 12)		
Mutate offspring		(24, 30)	(25, 13)		
New generation	(25, 30)	(24, 30)	(25, 13)	(20, 12)	(24, 25)


Best individuals
Lucky individuals

$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

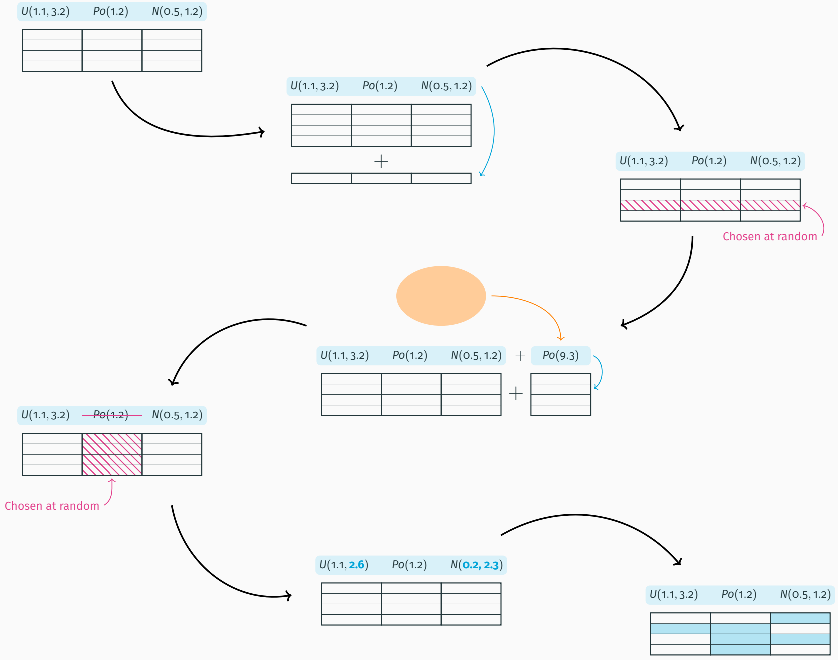
Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49
Select parents	(25, 30)			(20, 12)	(24, 25)
Create offspring		(24, 30)	(25, 12)		
Mutate offspring		(24, 30)	(25, 13)		
New generation	(25, 30)	(24, 30)	(25, 13)	(20, 12)	(24, 25)



$N(0,1)$ $Po(3.6)$
$U(3,5)$ $N(2,2)$ $Po(2.5)$
Dimensions
Columns
(1)
(2)
$N(0,1)$ $Po(2.5)$ $U(3,5)$

$$\max \quad f : \mathbb{N}^2 \to \mathbb{N}; \qquad f(x_1, x_2) = x_1 + x_2$$

Population	(25, 30)	(12, 1)	(11, 0)	(20, 12)	(24, 25)
Get fitness	55	13	11	42	49
Select parents	(25, 30)			(20, 12)	(24, 25)
Create offspring		(24, 30)	(25, 12)		
Mutate offspring		(24, 30)	(25, 13)		
New generation	(25, 30)	(24, 30)	(25, 13)	(20, 12)	(24, 25)

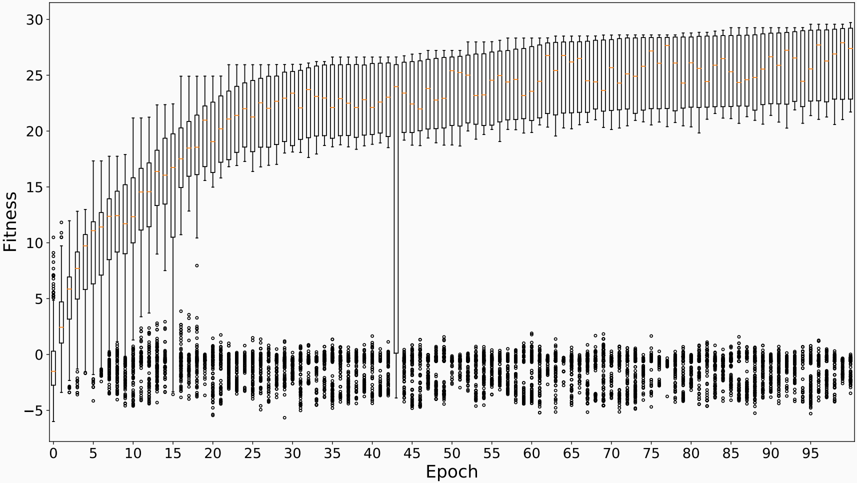


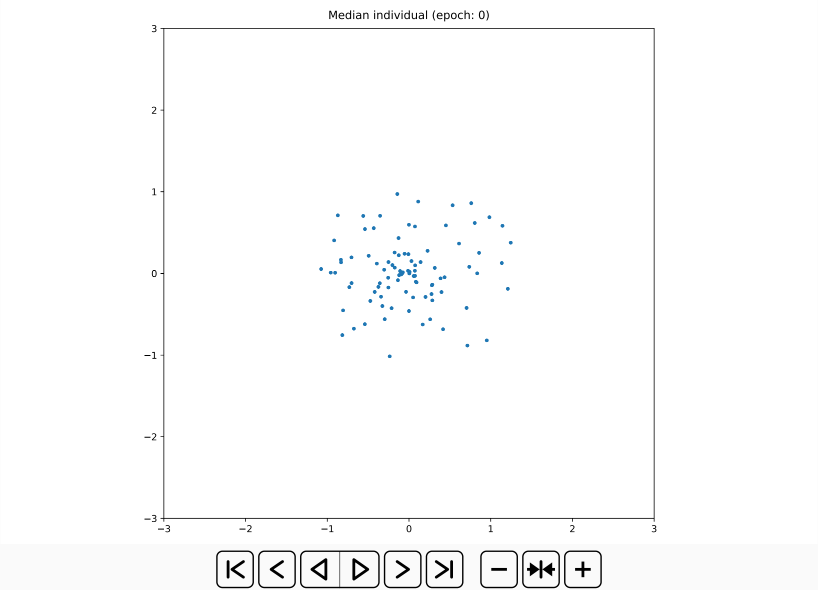
U(1.1, 3.2) Po(1.2) N(0.5, 1.2)
+
Chosen at random
Po(9.3)
Chosen at random
U(1.1, 2.6) Po(1.2) N(0.2, 2.3)

Some example use cases

Maximise

$$f : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, \qquad f(A, B) = Var(A) - \max_i |B_i - 1|$$


Fitness
Epoch
30
25
20
15
10
5
0
−5
0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95

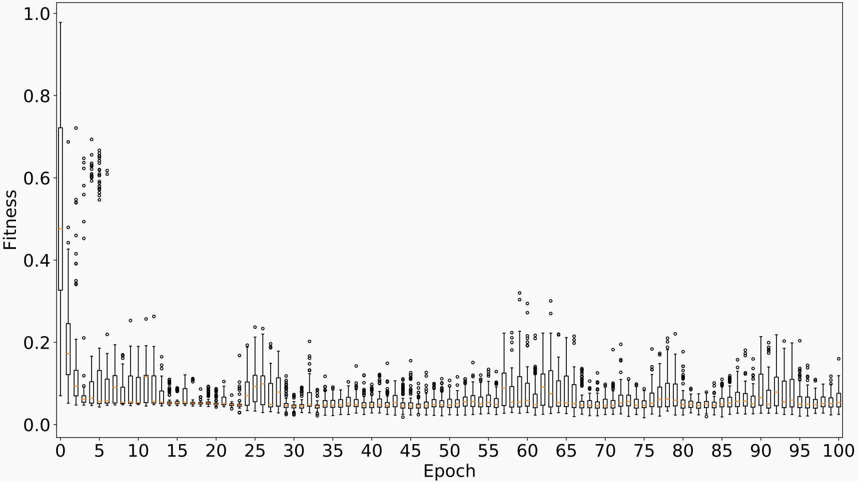

Median individual (epoch: 0)
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
3

Given:

- a large column, X;
- some sampling proportion, $p \in [0, 1]$;
- a number of samples to take, k

Minimise

The maximum sampled mean of X



1.0
0.8
0.6
0.4
0.2
0.0
Fitness
0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100
Epoch

$U(-0.06, 0.91)$

Metadata

X

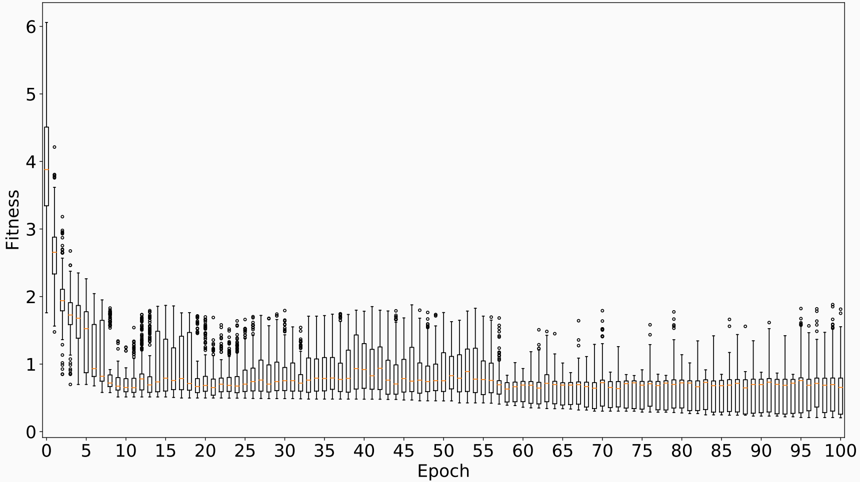
	0
0	-0.142152
1	0.0433848
2	0.0727756
...	...
45	0.0670734
46	0.0866879
47	0.0137398

Dataset

Given a set of k dissimilarity measures:

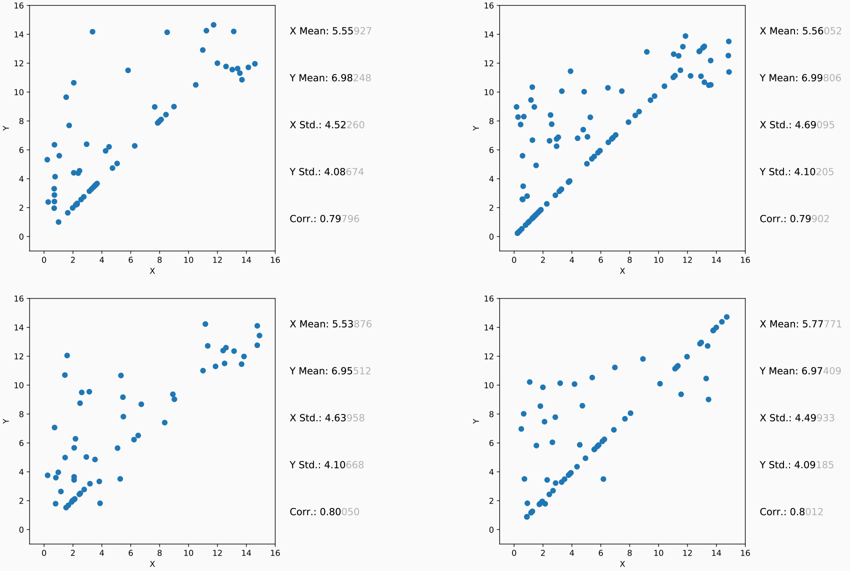
$$f_1, \ldots, f_k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$$

Minimise their sum



Fitness
Epoch
0
1
2
3
4
5
6
0
5
10
15
20
25
30
35
40
45
50
55
60
65
70
75
80
85
90
95
100

X Mean: 5 *Y* Mean: 7 *X* Std.: 4.7 *Y* Std.: 4.1 Corr.: 0.8



edo.readthedocs.io

Henry Wilde

Twitter: @daffidwilde

Email: wildehd@cardiff.ac.uk

Repository: `github.com/daffidwilde/edo`

Documentation: `edo.readthedocs.io`

Paper in preparation:

“Evolutionary Dataset Optimisation: understanding algorithm quality through evolution”

- A fitness function, f, which acts on a single dataset
- A population size, $N \in \mathbb{N}$
- A maximum number of iterations, $M \in \mathbb{N}$
- A selection parameter to detail the proportion of the fittest individuals to carry forward, $b \in [0, 1]$
- A mutation probability, $p_m \in [0, 1]$

- Limits on the number of rows a dataset can have:

$$R \in \left\{(r_{\min}, r_{\max}) \in \mathbb{N}^2 \mid r_{\min} \leq r_{\max}\right\}$$

- Limits on the number of columns a dataset can have:

$$C := \left(C_1, \ldots, C_{|\mathcal{P}|}\right) \text{ where } C_j \in \left\{(c_{\min}, c_{\max}) \in (\mathbb{N} \cup \{\infty\})^2 \mid c_{\min} \leq c_{\max}\right\}$$

  for each $j = 1, \ldots, |\mathcal{P}|$
- A set of probability distribution families, $\mathcal{P}$. Each family in this set has some parameter limits which form a part of the overall search space
- A probability vector to sample distributions from $\mathcal{P}$, $w = \left(w_1, \ldots, w_{|\mathcal{P}|}\right)$
- A second selection parameter, $l \in [0, 1]$, to allow for a small proportion of "lucky" individuals to be carried forward
- A shrink factor, $s \in [0, 1]$. The relative size of a component of the search space to be retained after adjustment