UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

Sesión 14.1: Locality-sensitive hashing

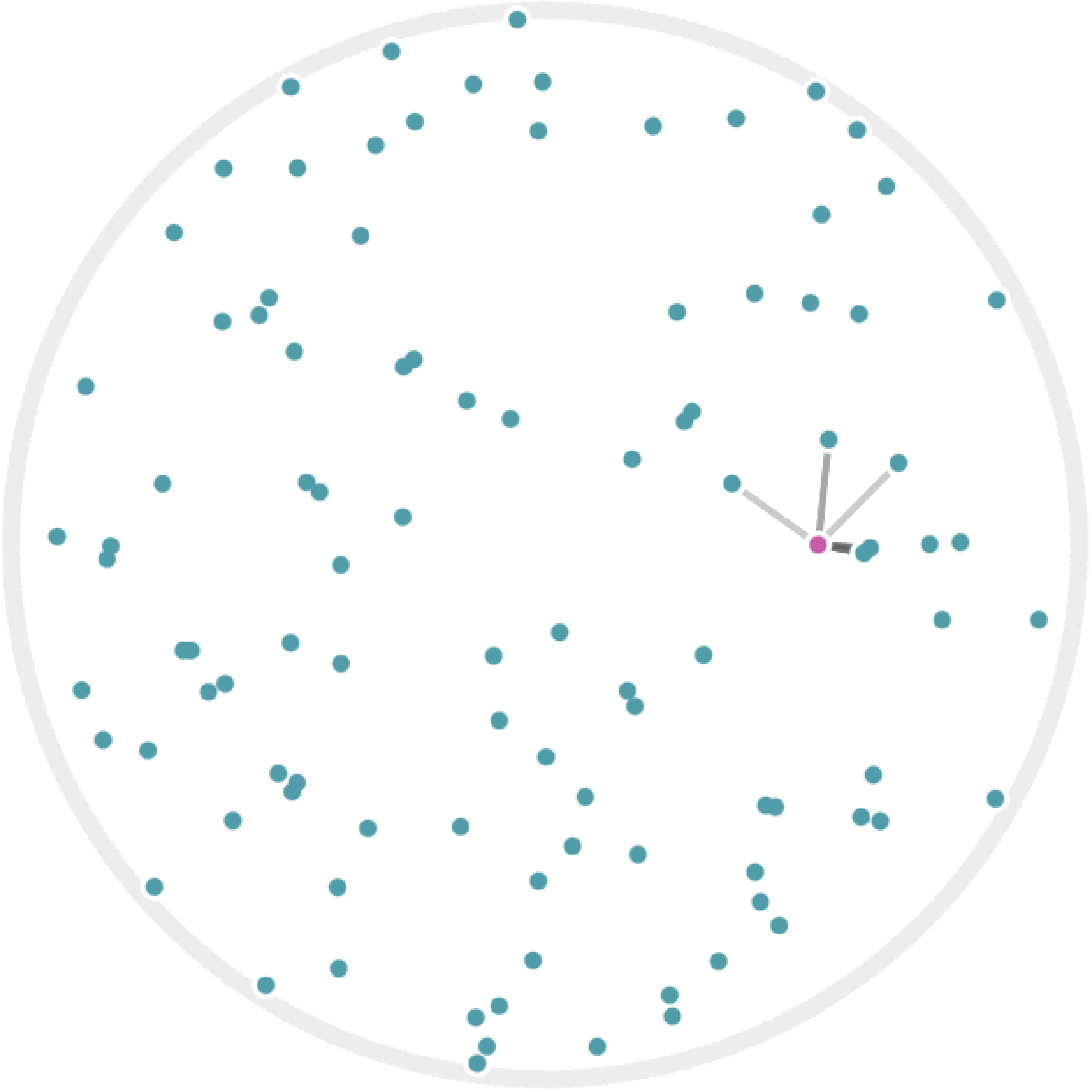
CS3102 EDA

Índice

1. Locality-sensitive hashing (LSH)
2. MinHashing
3. Shazam

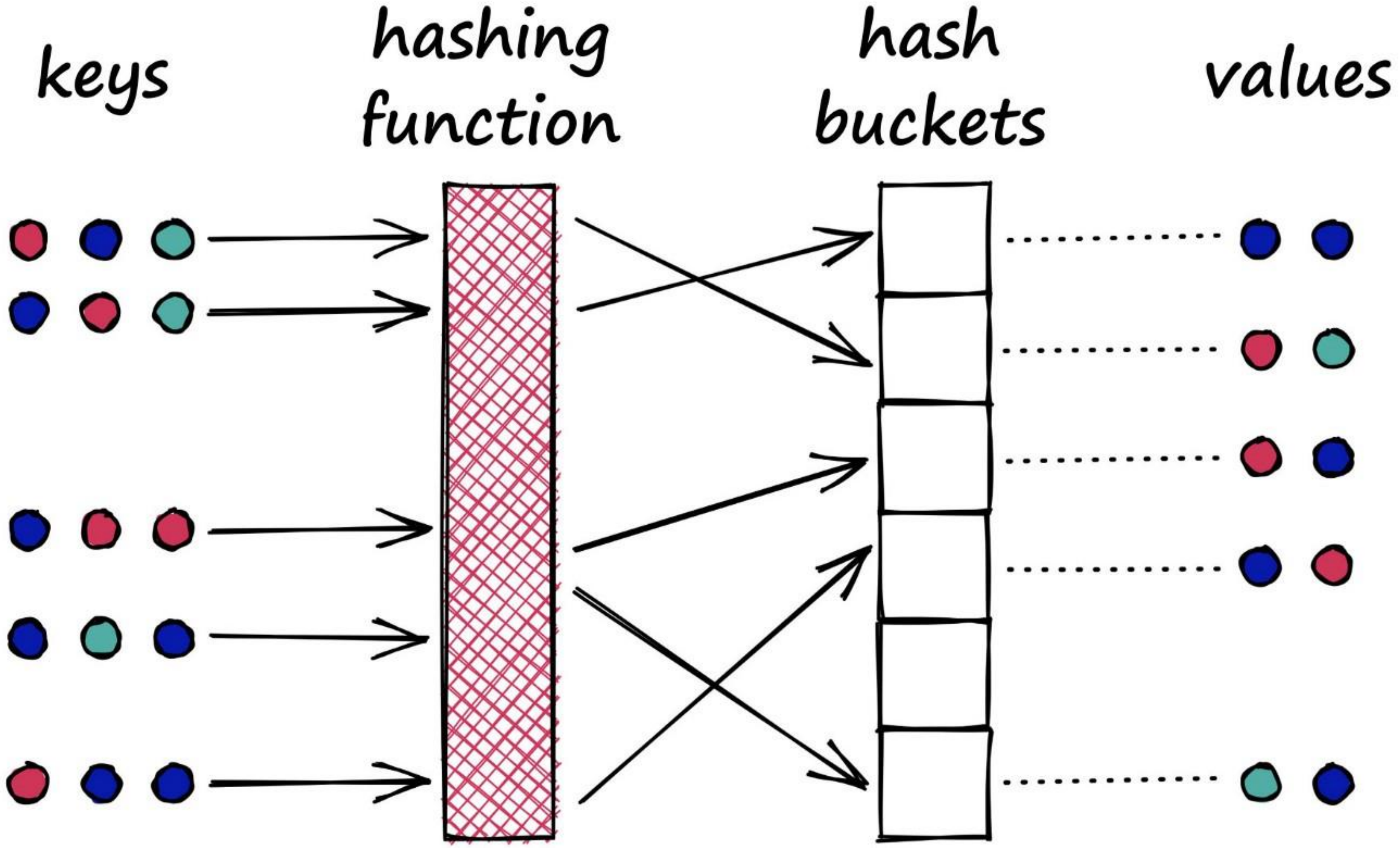
1. Locality-sensitive hashing (LSH)

Locality-sensitive *hashing (LSH)*

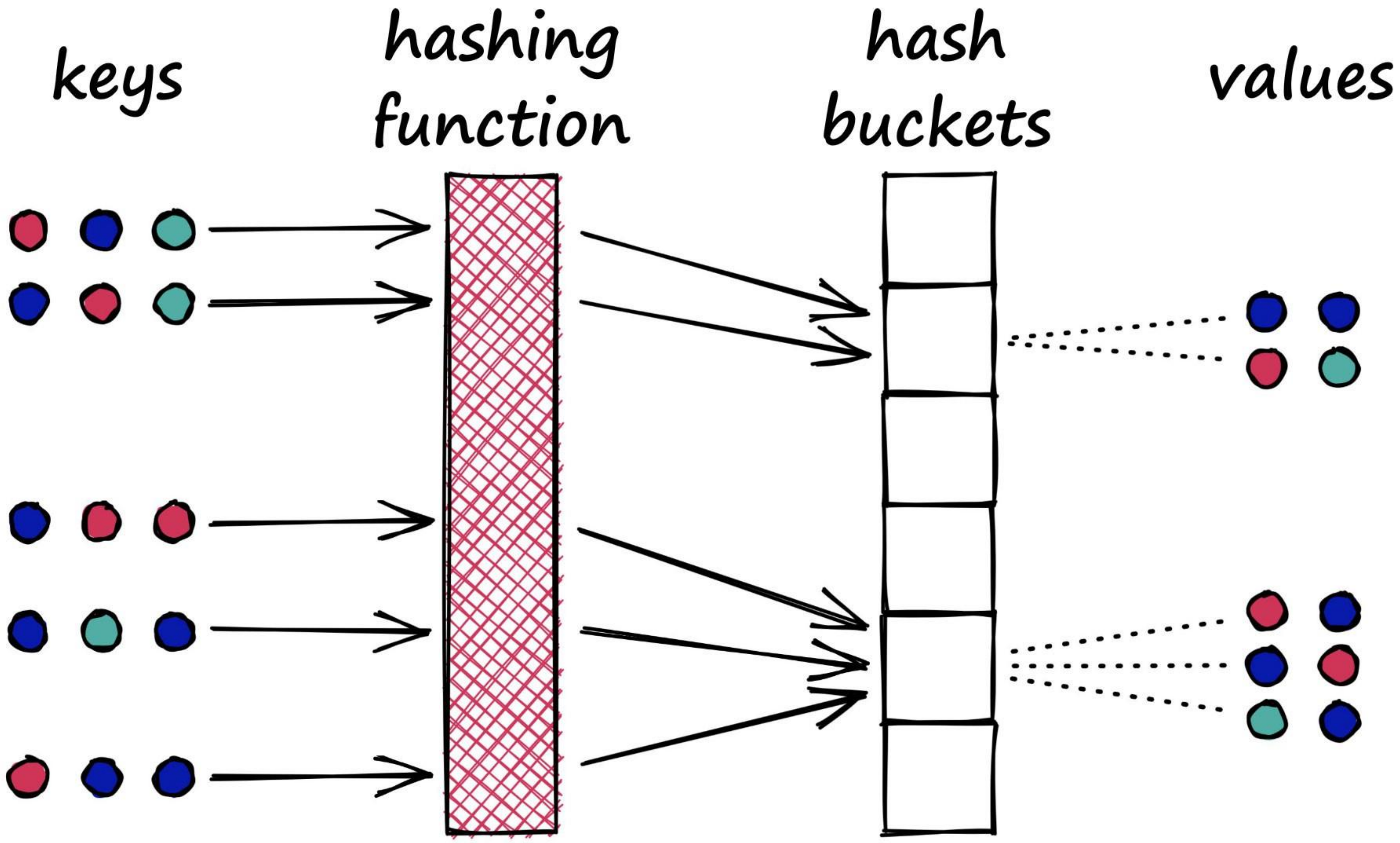


Piotr Indyk and Rajeev Motwani (1998) “Approximate nearest neighbors: towards removing the curse of dimensionality”. Proceedings of the thirtieth annual ACM symposium on Theory of computing. p. 604-613.

Hash *Function*



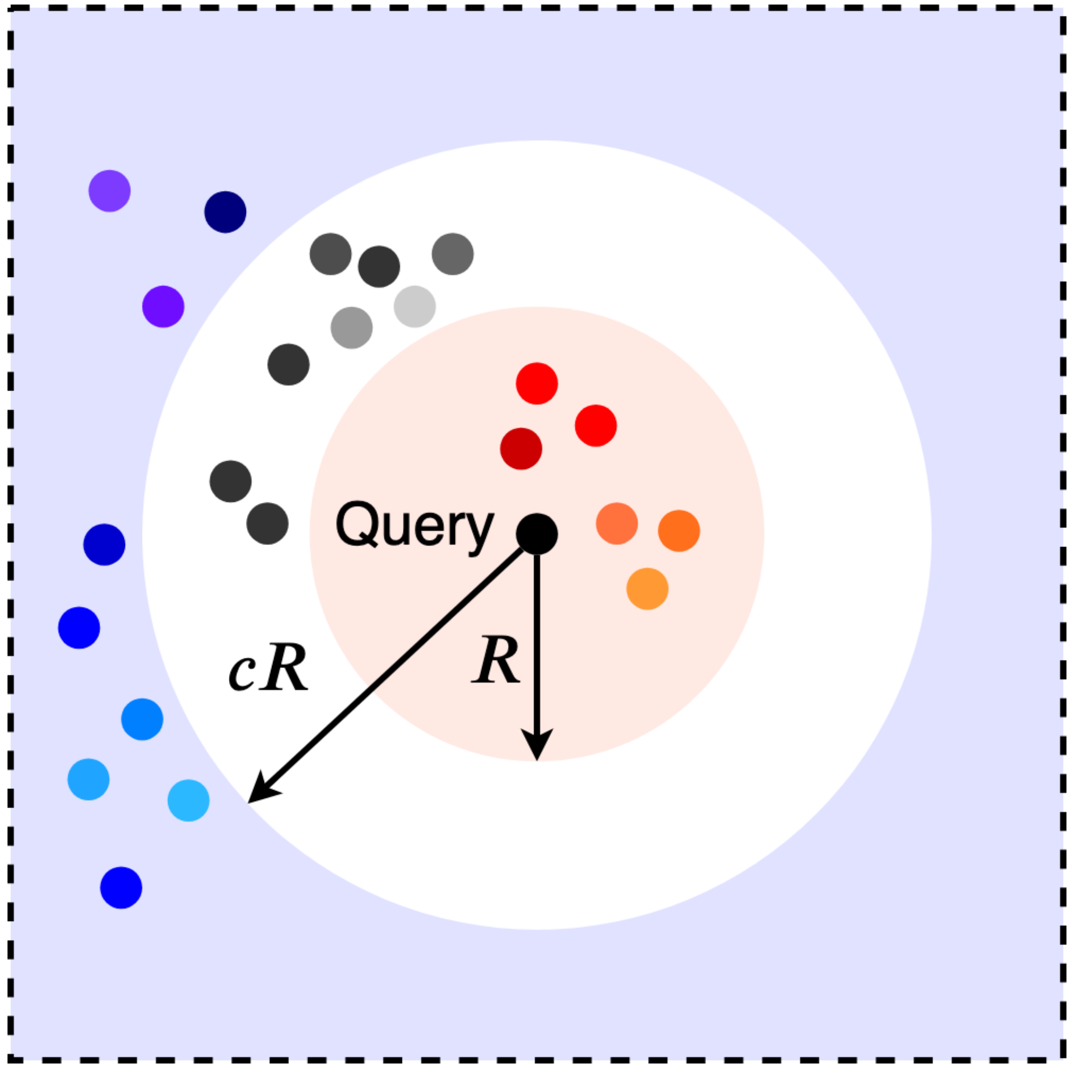
Locality-sensitive *hashing (LSH)*



Locality-sensitive *hashing (LSH)*

LSH *family* $\mathcal{F}$ es (R, cR, p_1, p_2)-sensitive con respecto a la distancia $d(x, y)$ si para algún $h \in \mathcal{H}$ tenemos que:

- Si $d(x, y) \leq R$ entonces $P_{\mathcal{H}}[h(x) = h(y)] \geq p_1$
- Si $d(x, y) \geq cR$ entonces $P_{\mathcal{H}}[h(x) = h(y)] \leq p_2$

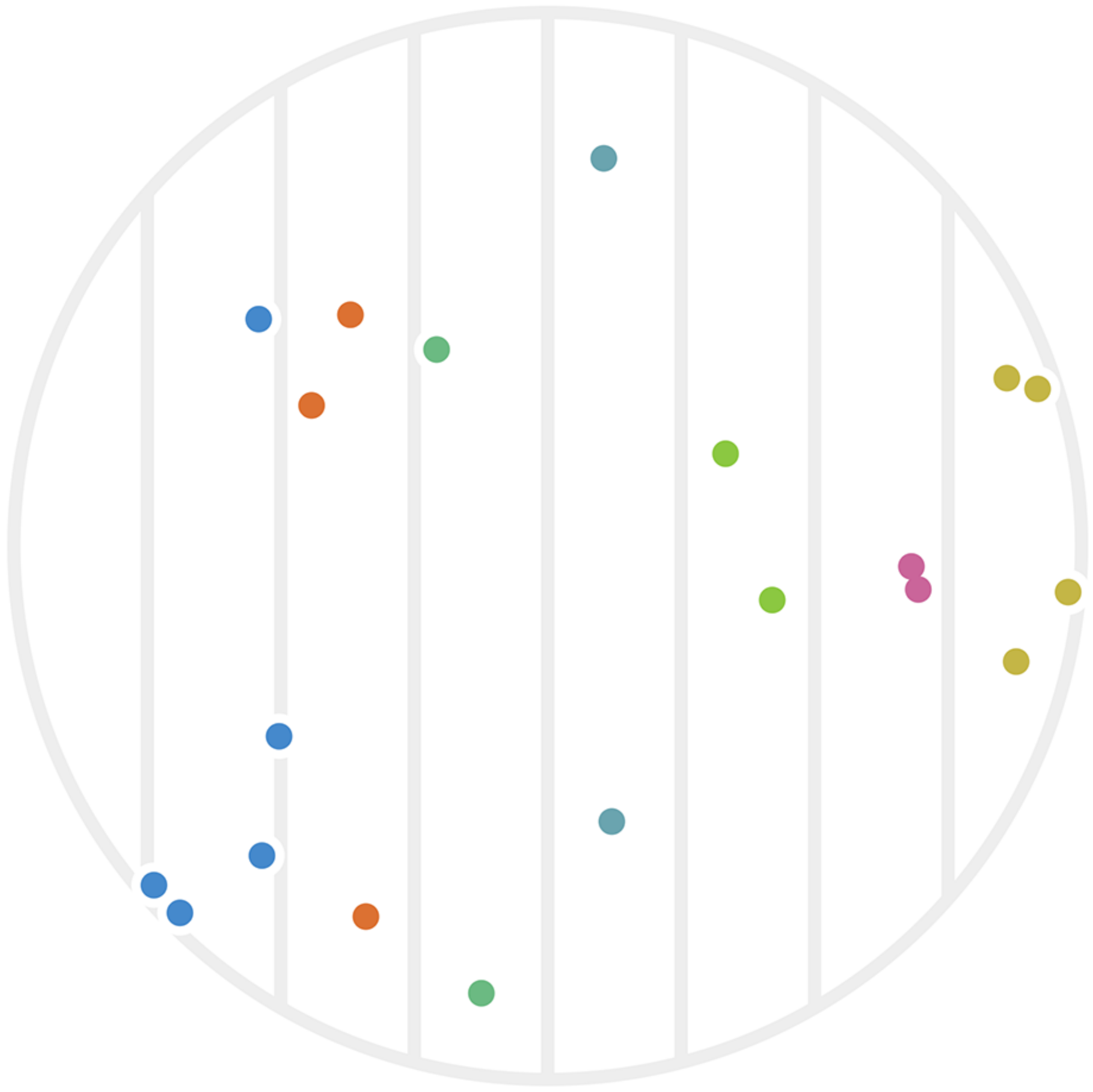


Hashing points *with projections*

Ejemplo:

$h_1 : \mathbb{R}^2 \longrightarrow \mathbb{Z}$ $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$

$$h_1(\boldsymbol{x}) = \lfloor x_1 \rfloor$$



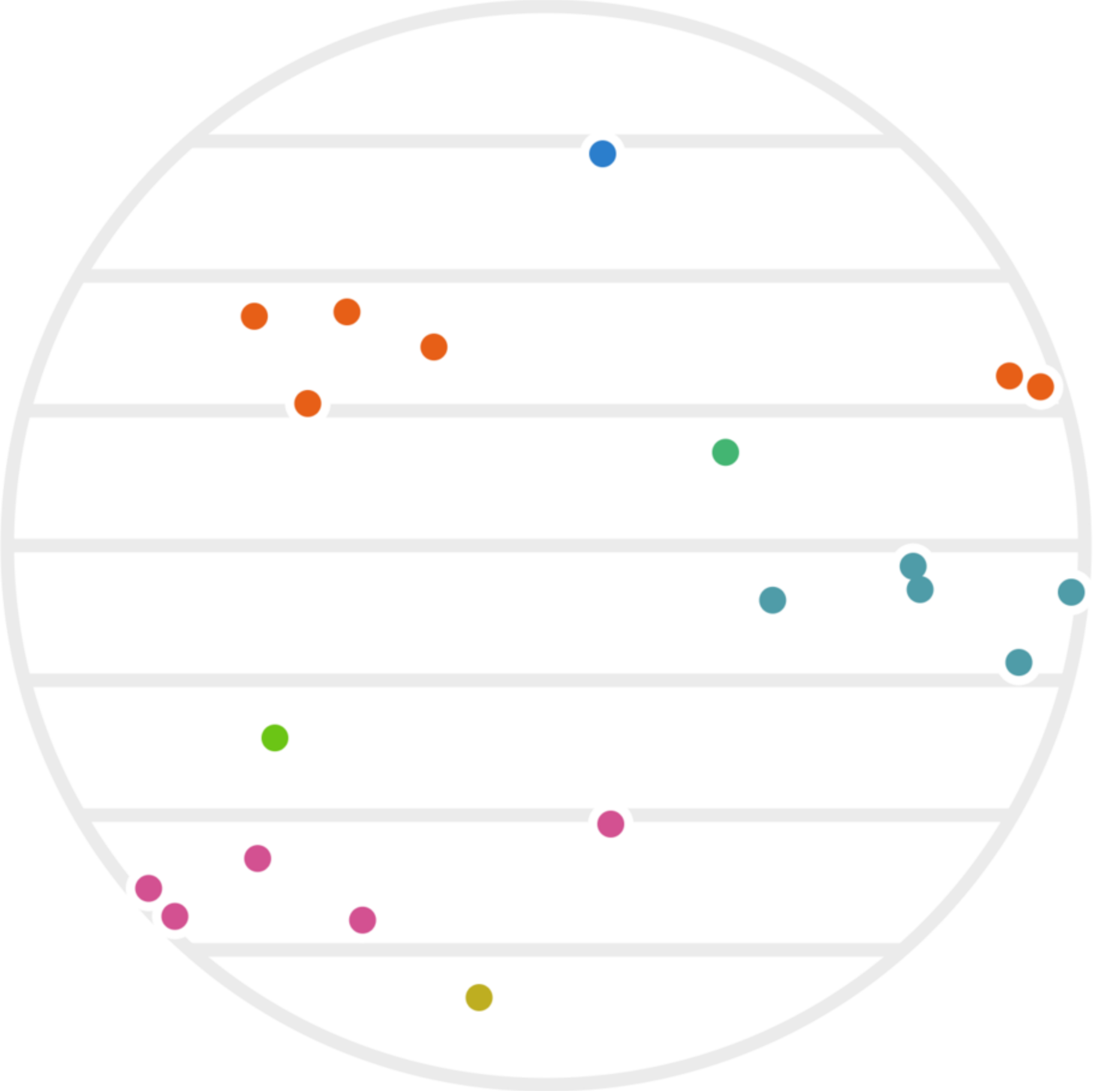
$a \sim b \Leftrightarrow h_1(a) = h_1(b)$

Hashing points *with projections*

Ejemplo:

$h_2 : \mathbb{R}^2 \longrightarrow \mathbb{Z}$ $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$

$$h_2(\boldsymbol{x}) = \lfloor x_2 \rfloor$$



$a \sim b \Leftrightarrow h_1(a) = h_1(b)$

Hashing points *with projections*

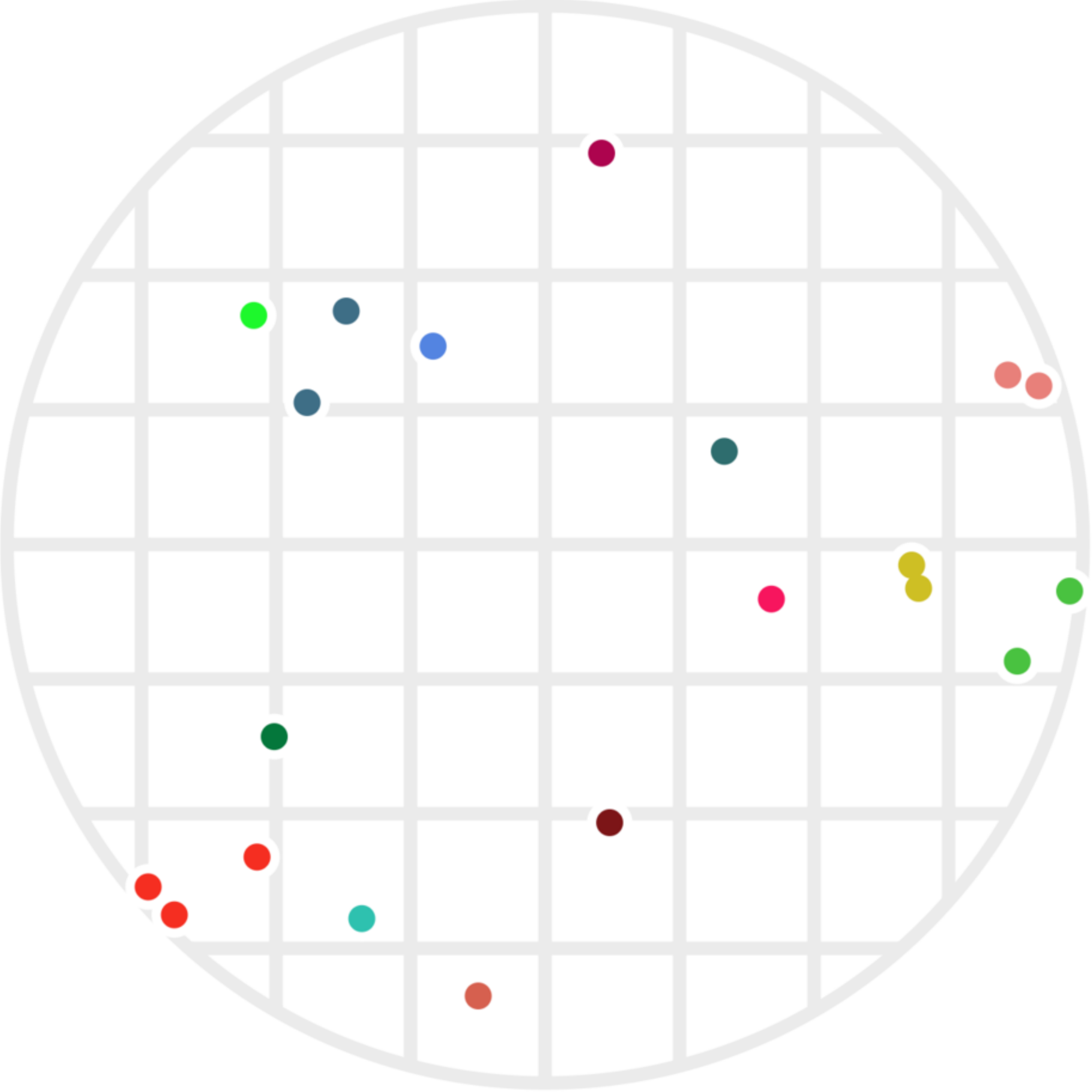
Ejemplo:

$$h_1 : \mathbb{R}^2 \longrightarrow \mathbb{Z} \qquad \boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$$

$$h_2 : \mathbb{R}^2 \longrightarrow \mathbb{Z}$$

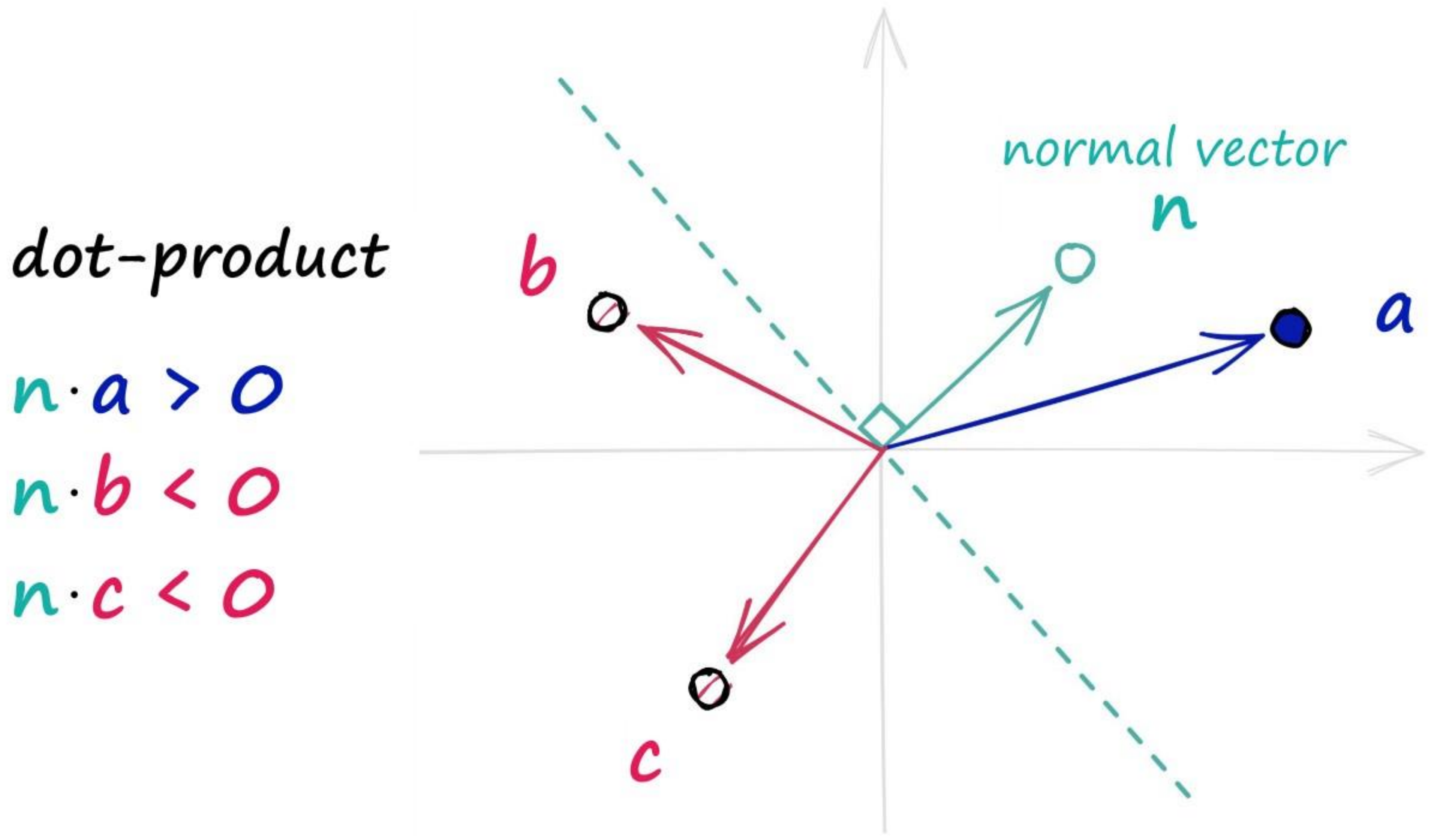
$$h_1(\boldsymbol{x}) = \lfloor x_1 \rfloor$$

$$h_2(\boldsymbol{x}) = \lfloor x_2 \rfloor$$

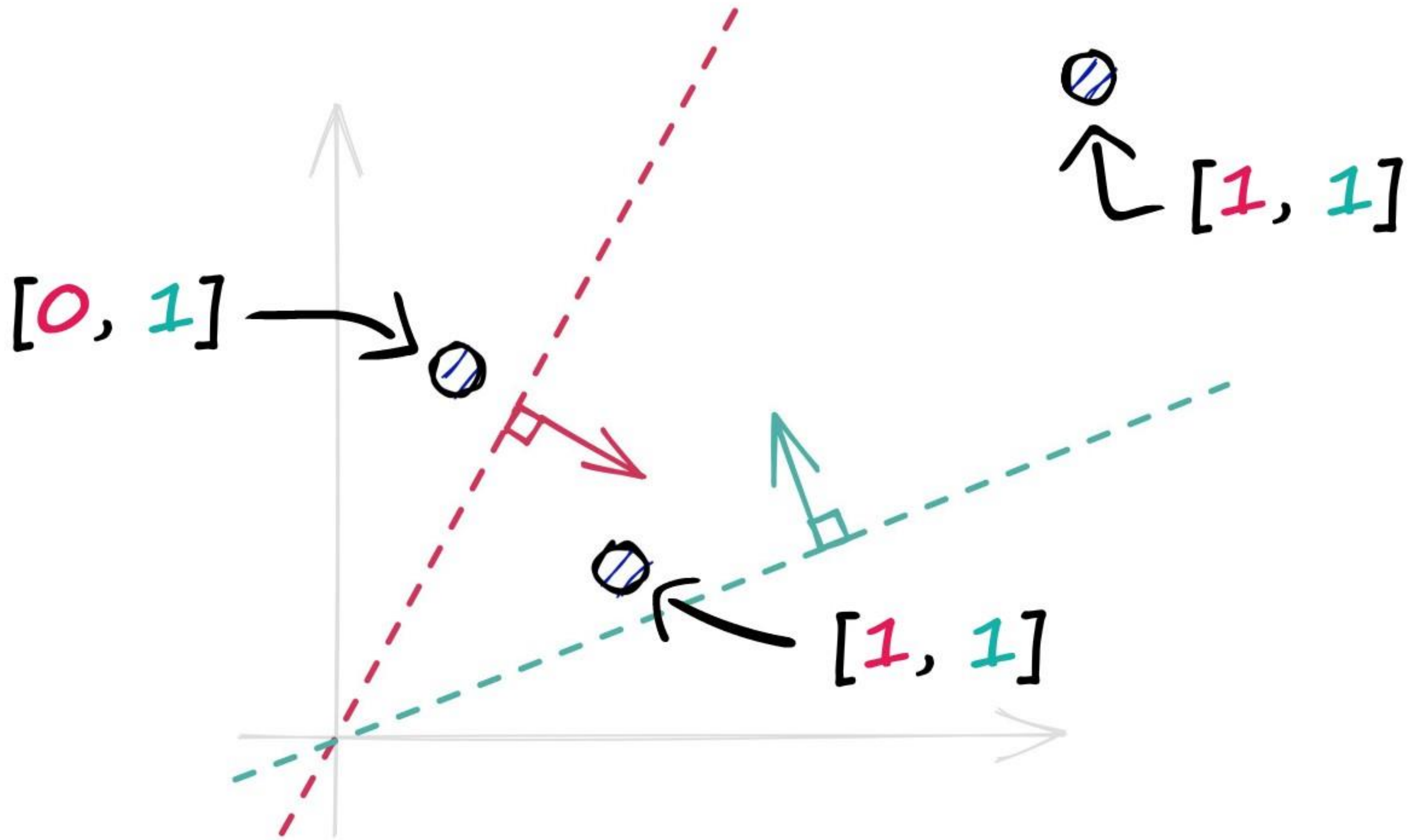


$$a \sim b \Leftrightarrow \begin{cases} h_1(a) = h_1(b) \\ h_2(a) = h_2(b) \end{cases}$$

Hiper*planos!*



Hiper*planos!*



Proyección *aleatoria*

Ejemplo:

$h_1 \colon \mathbb{R}^2 \longrightarrow \mathbb{Z}$ $\quad$ $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$

$$h_1(\boldsymbol{x}) = \lfloor U x_1 + b \rfloor$$

Proyección *aleatoria*

Ejemplo:

$h_i\colon \mathbb{R}^2 \longrightarrow \mathbb{Z}$ $\qquad\qquad$ $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$

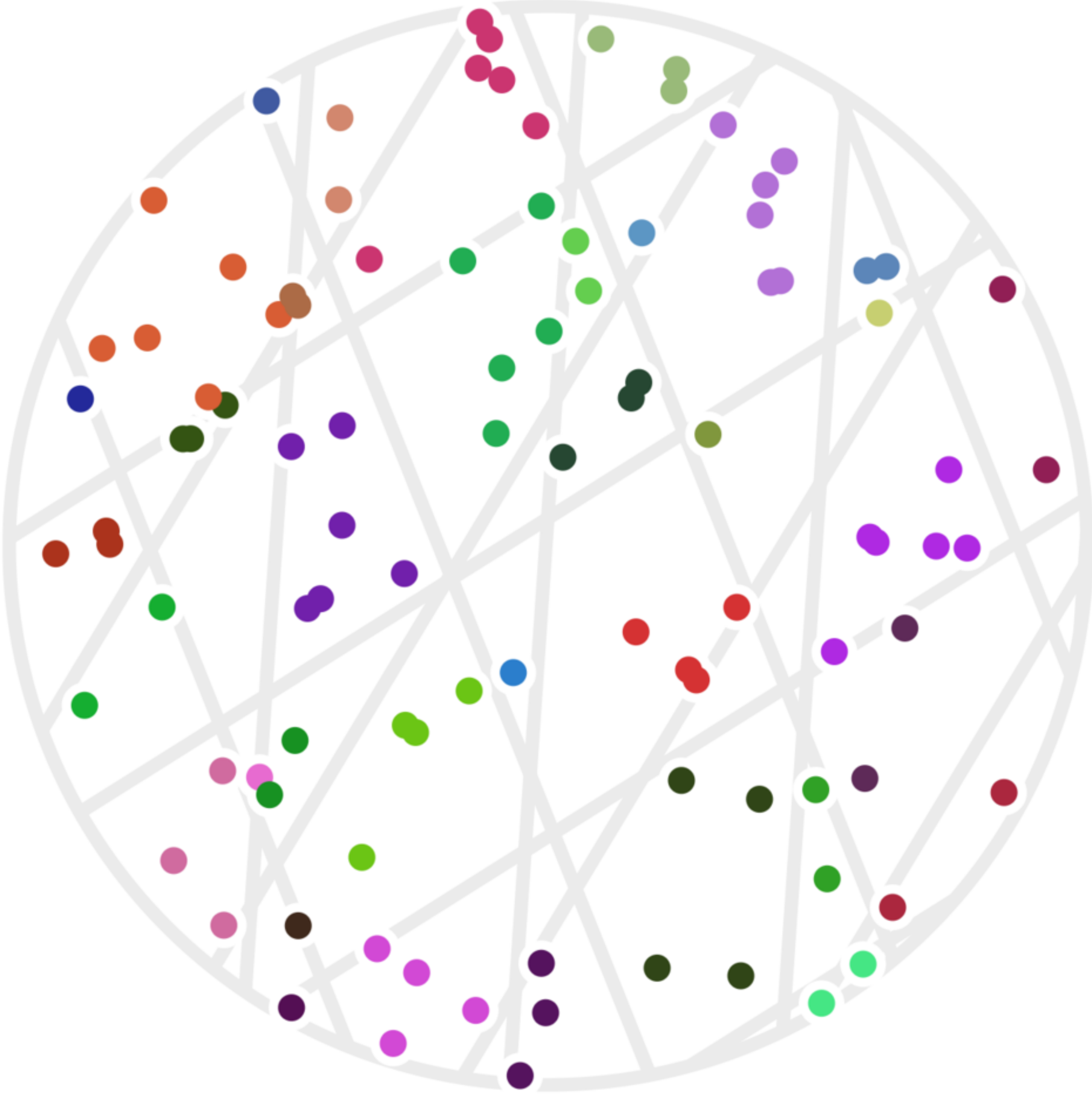
$$h_i(\boldsymbol{x}) = \lfloor U x_i + b \rfloor$$

Proyección *aleatoria*

Ejemplo:

$h_i : \mathbb{R}^2 \longrightarrow \mathbb{Z}$ $\qquad$ $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$

$$h_i(\boldsymbol{x}) = \lfloor U x_i + b \rfloor$$

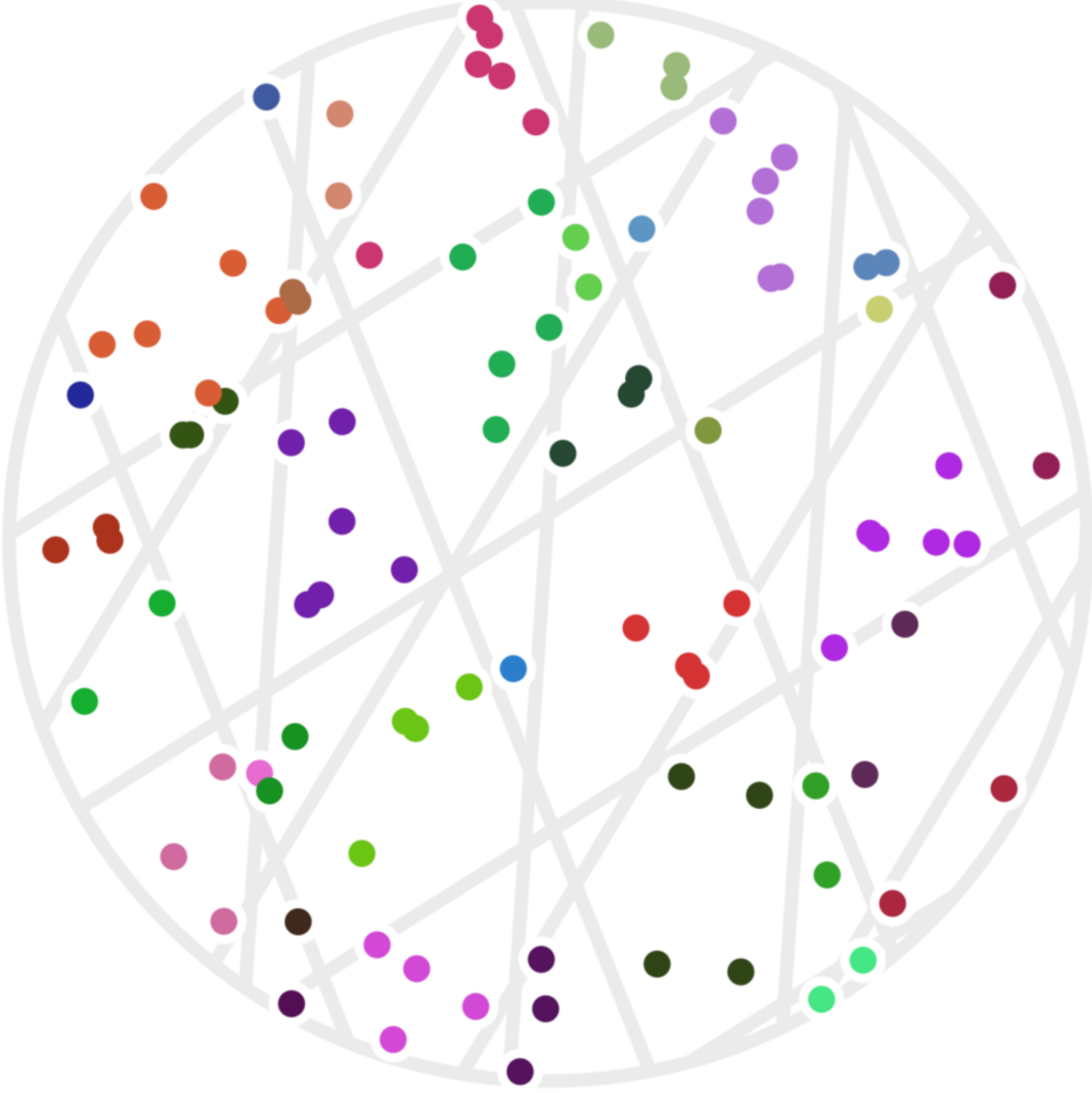


Proyección *aleatoria*

Ejemplo:

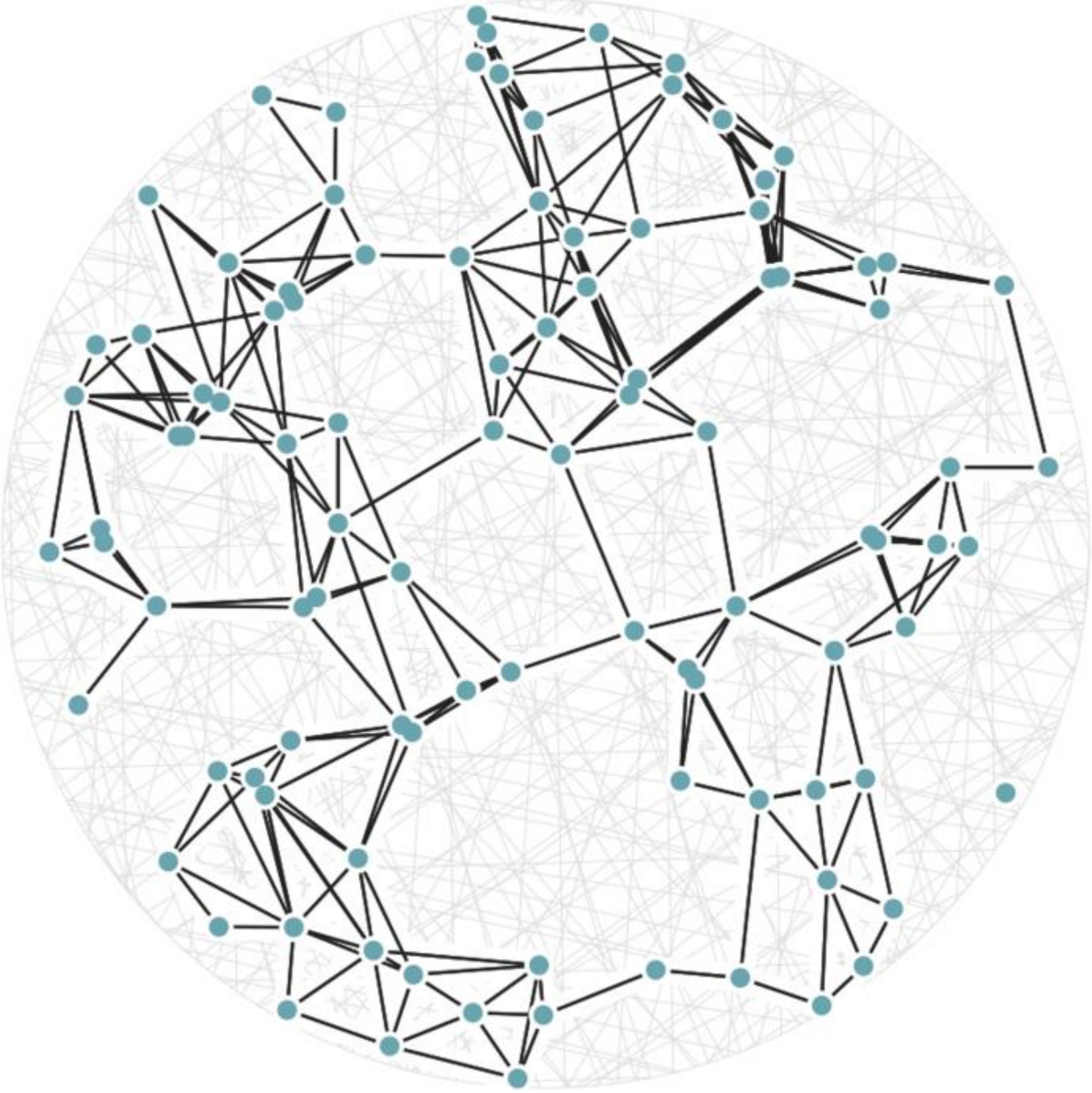
$h_i : \mathbb{R}^2 \longrightarrow \mathbb{Z}$ $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$

$$h_i(\boldsymbol{x}) = \lfloor U x_i + b \rfloor$$



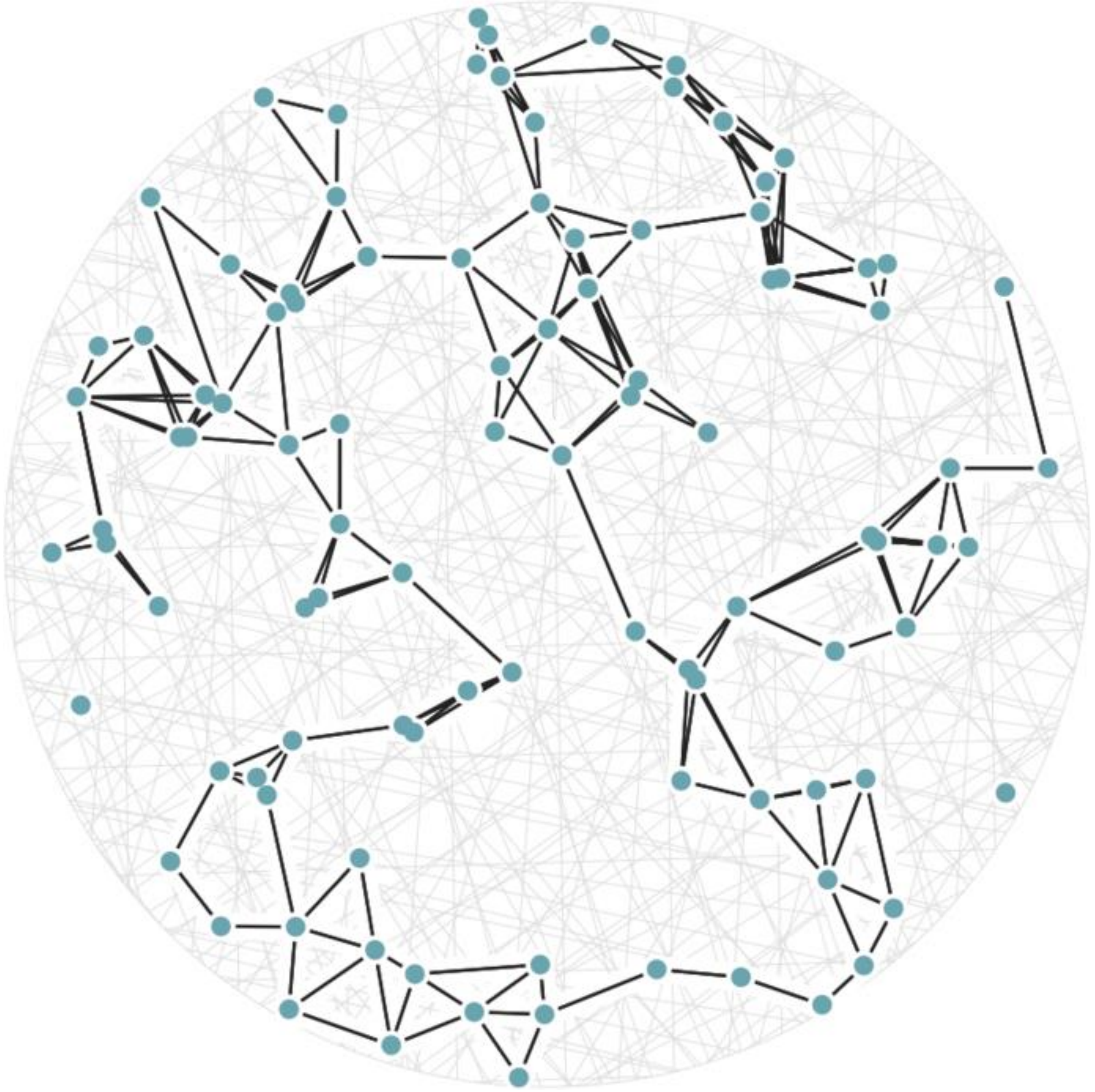
$a{\sim}b \Leftrightarrow \#\{i : h_i(a) = h_i(b)\} \geq j$

Proyección *aleatoria*



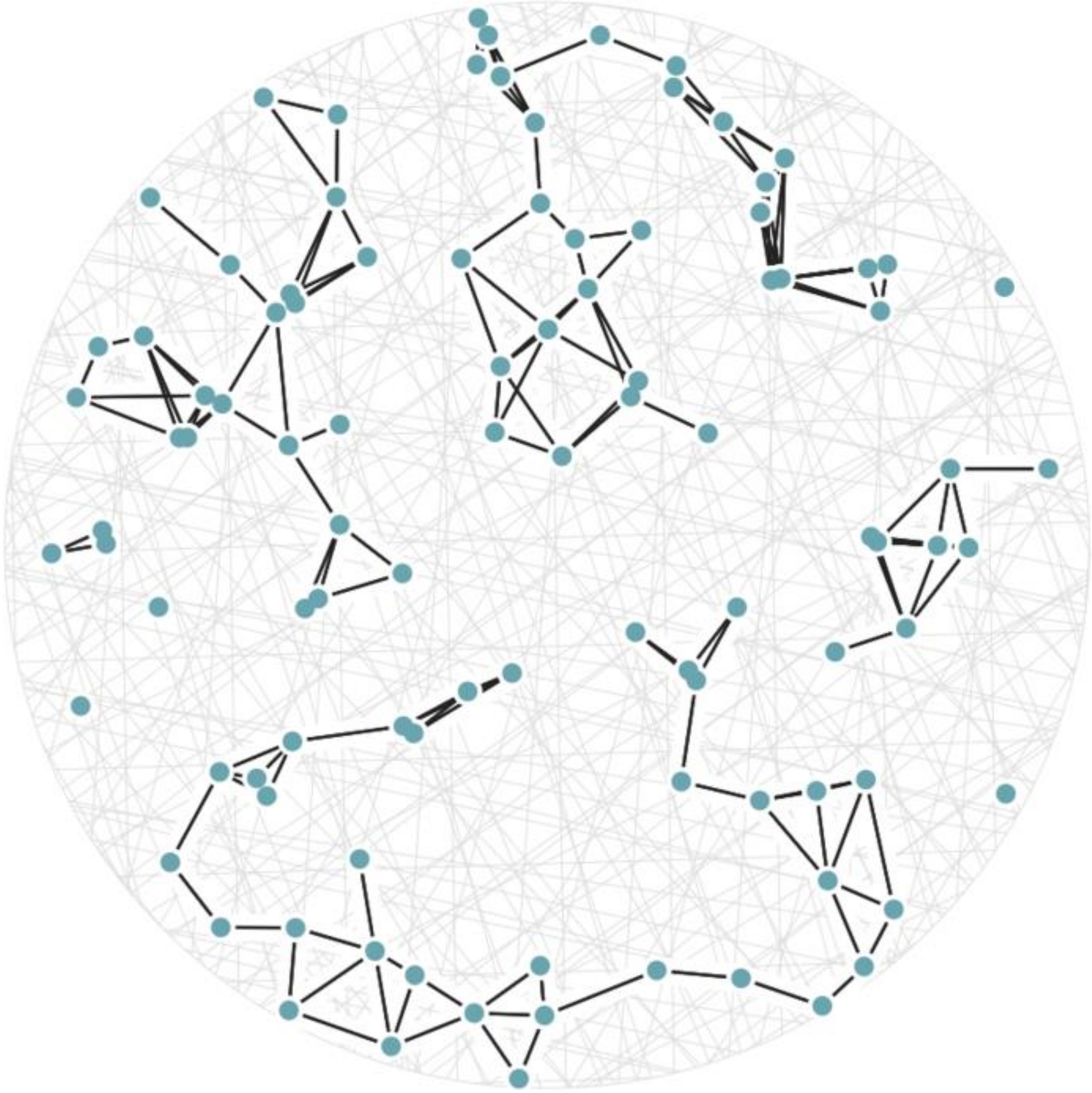
$$a \sim b \iff \#\{i : h_i(a) = h_i(b)\} \geq 6$$

Proyección *aleatoria*



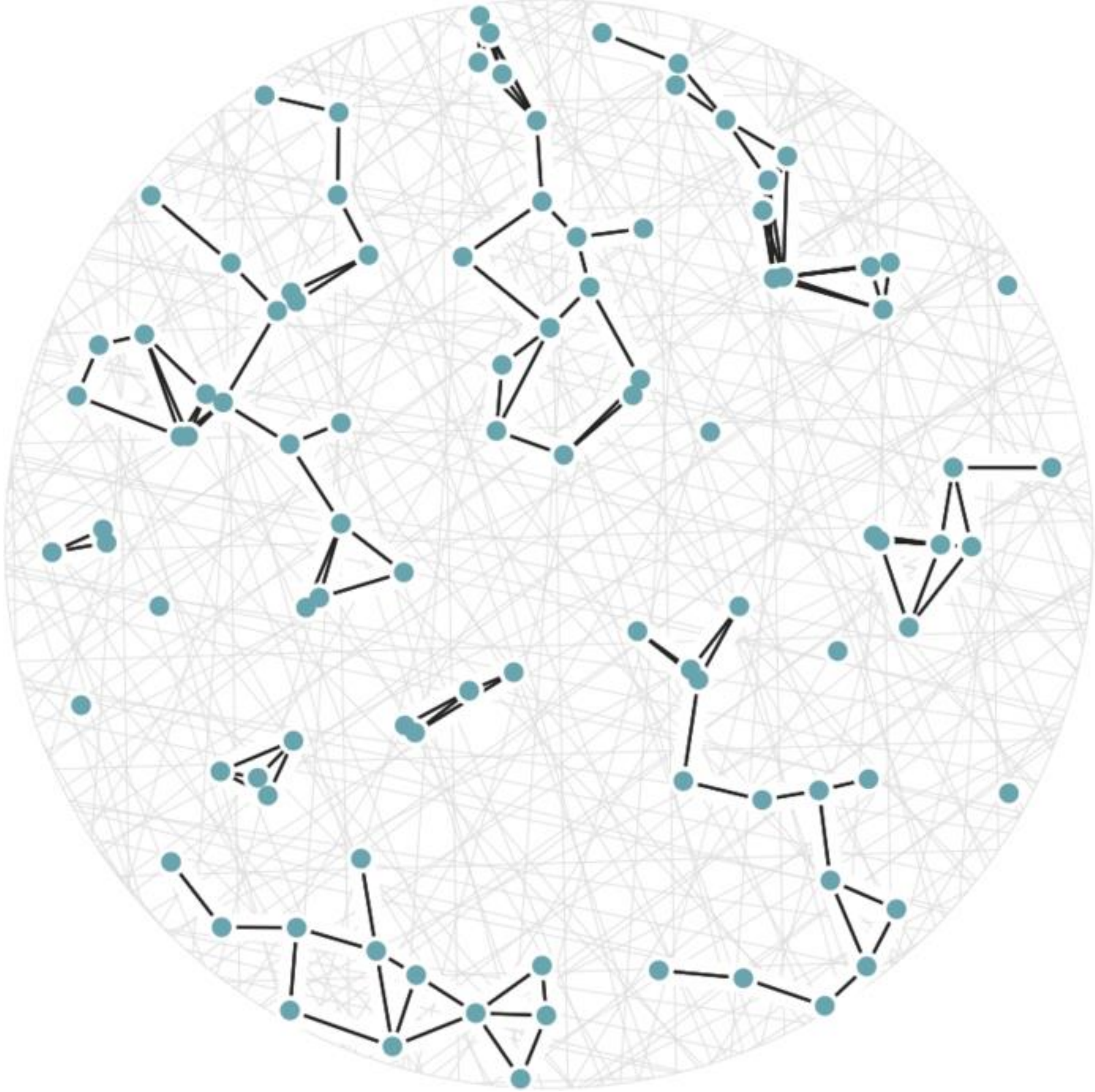
$a \sim b \iff \#\{i: h_i(a) = h_i(b)\} \geq 7$

Proyección *aleatoria*



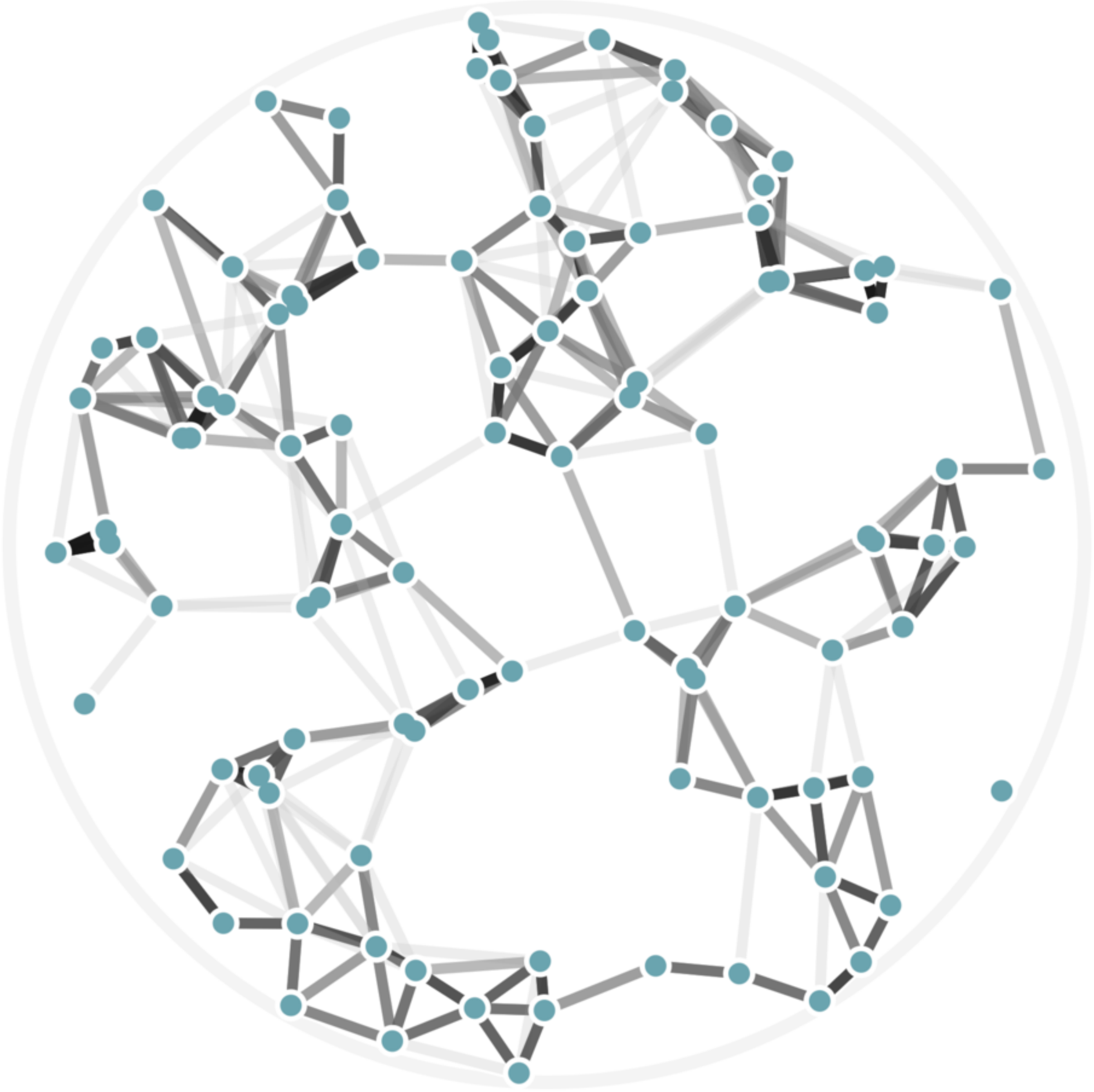
$$a \sim b \iff \#\{i : h_i(a) = h_i(b)\} \geq 8$$

Proyección *aleatoria*

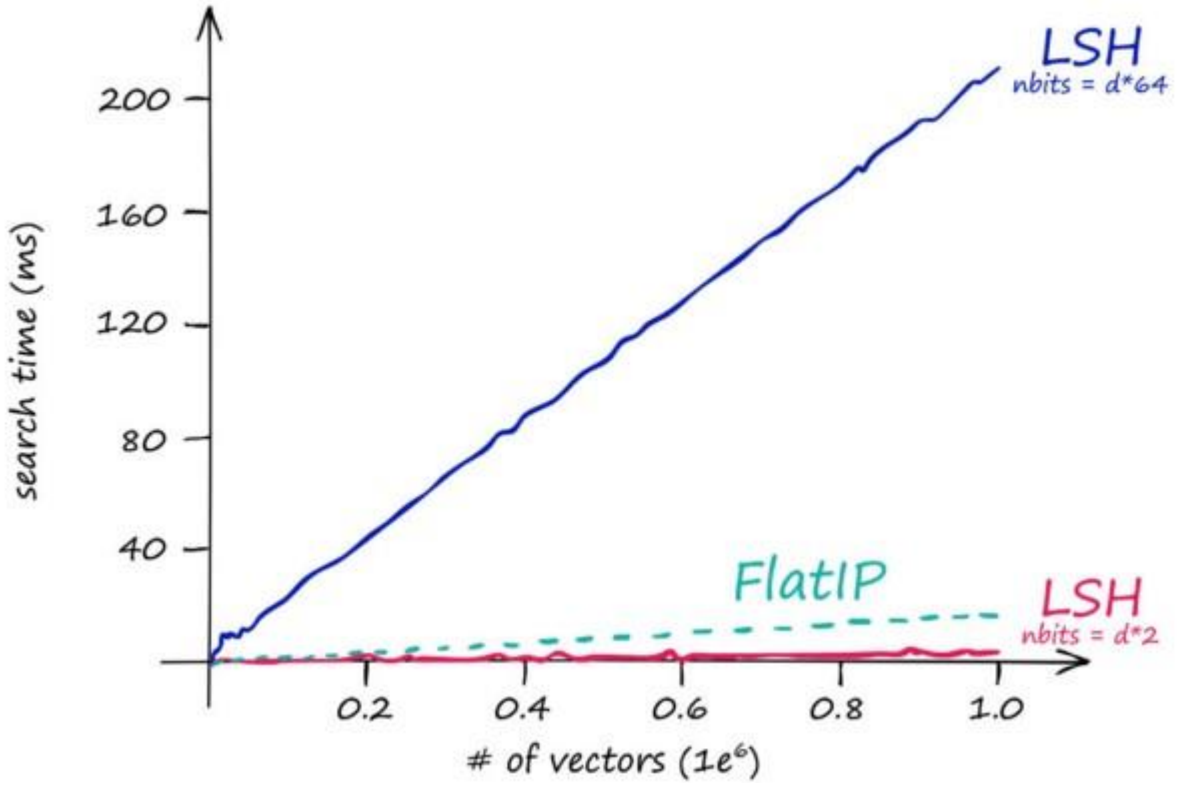


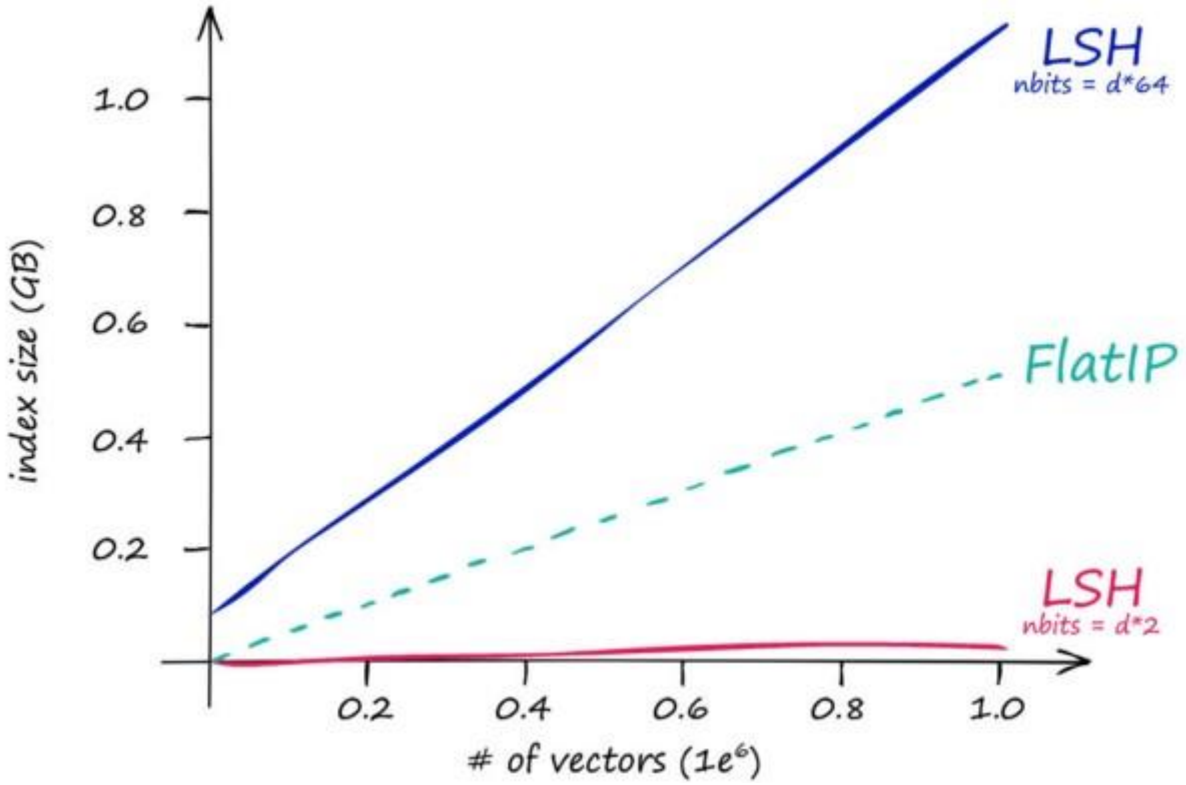
$$a \sim b \iff \#\{i : h_i(a) = h_i(b)\} \geq 9$$

Proyección *aleatoria*

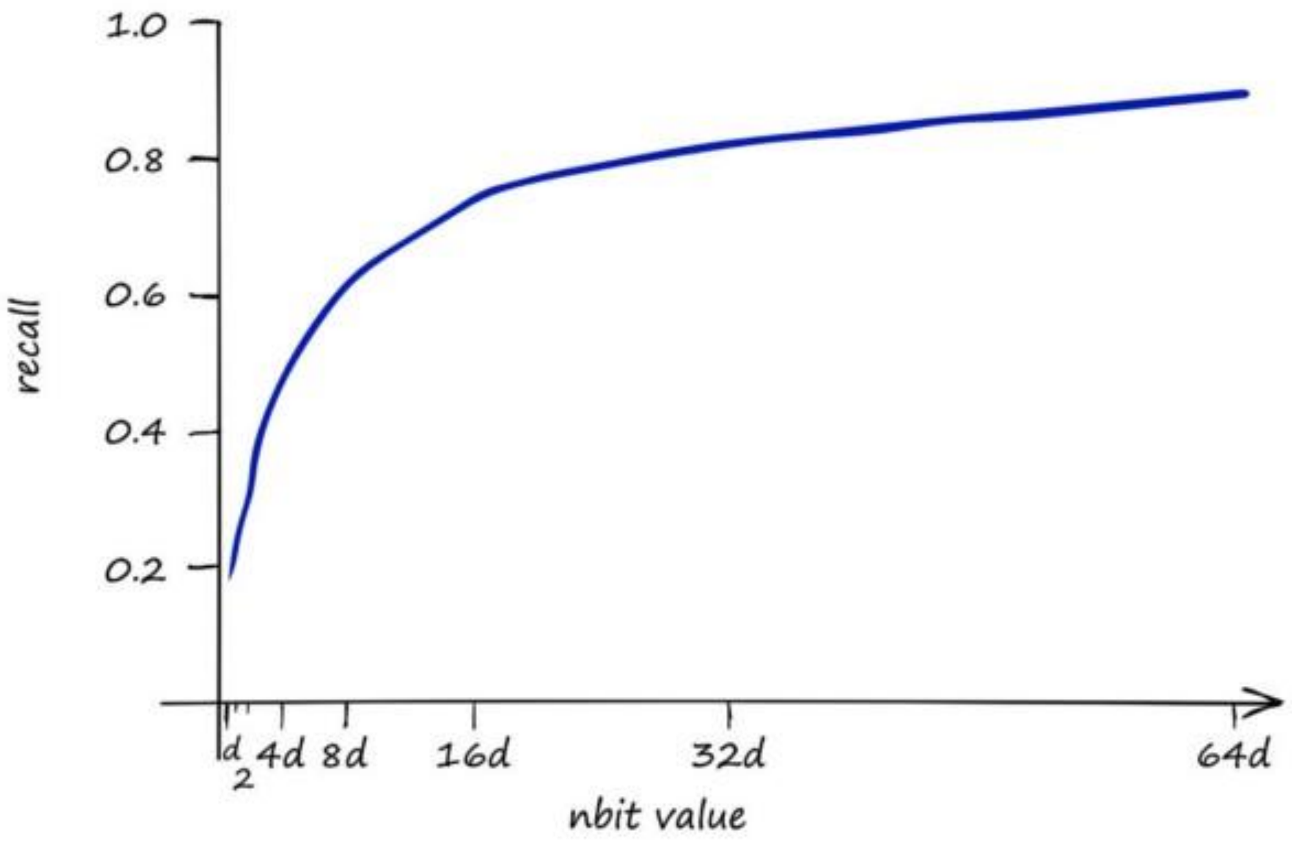


Proyección *aleatoria*

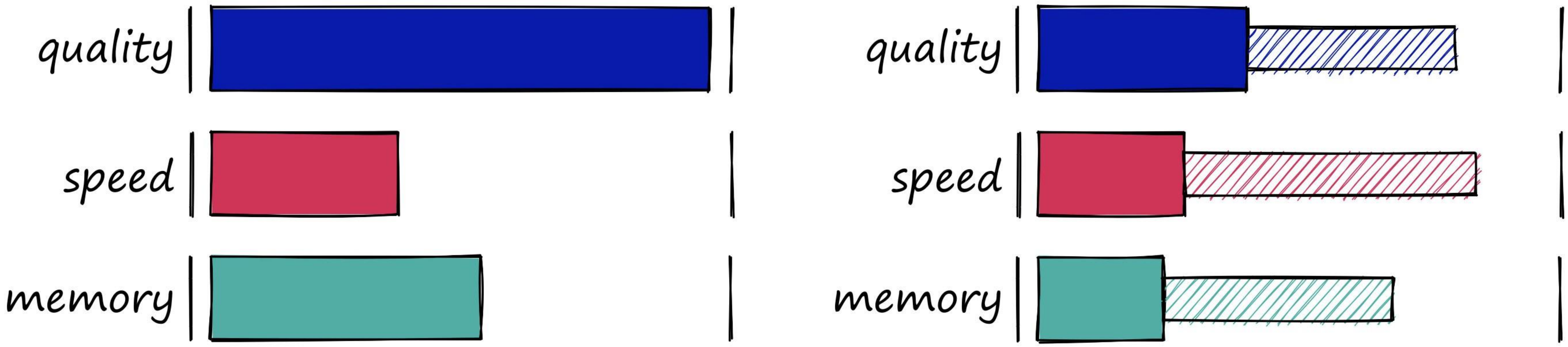




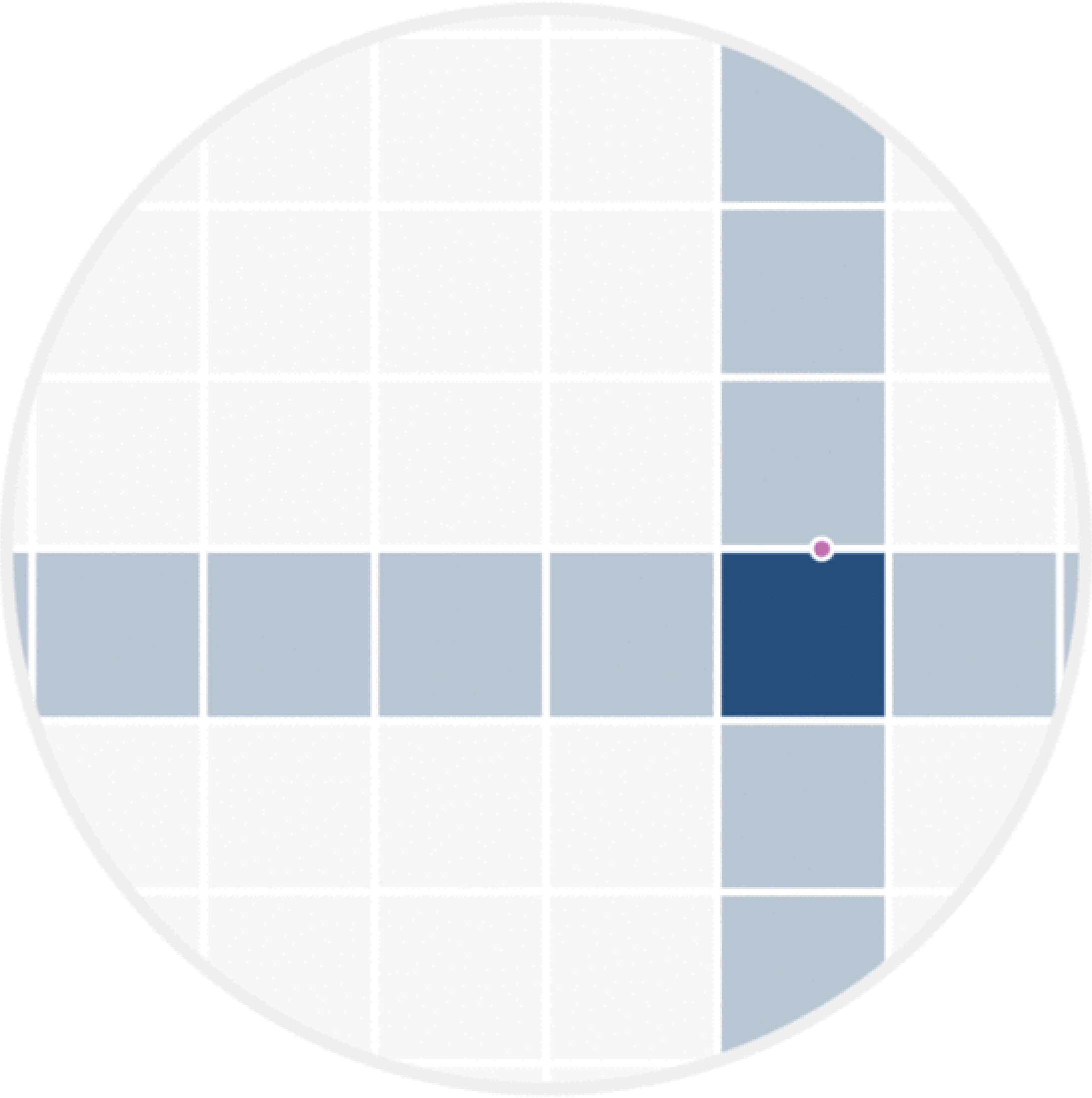
Proyección *aleatoria*



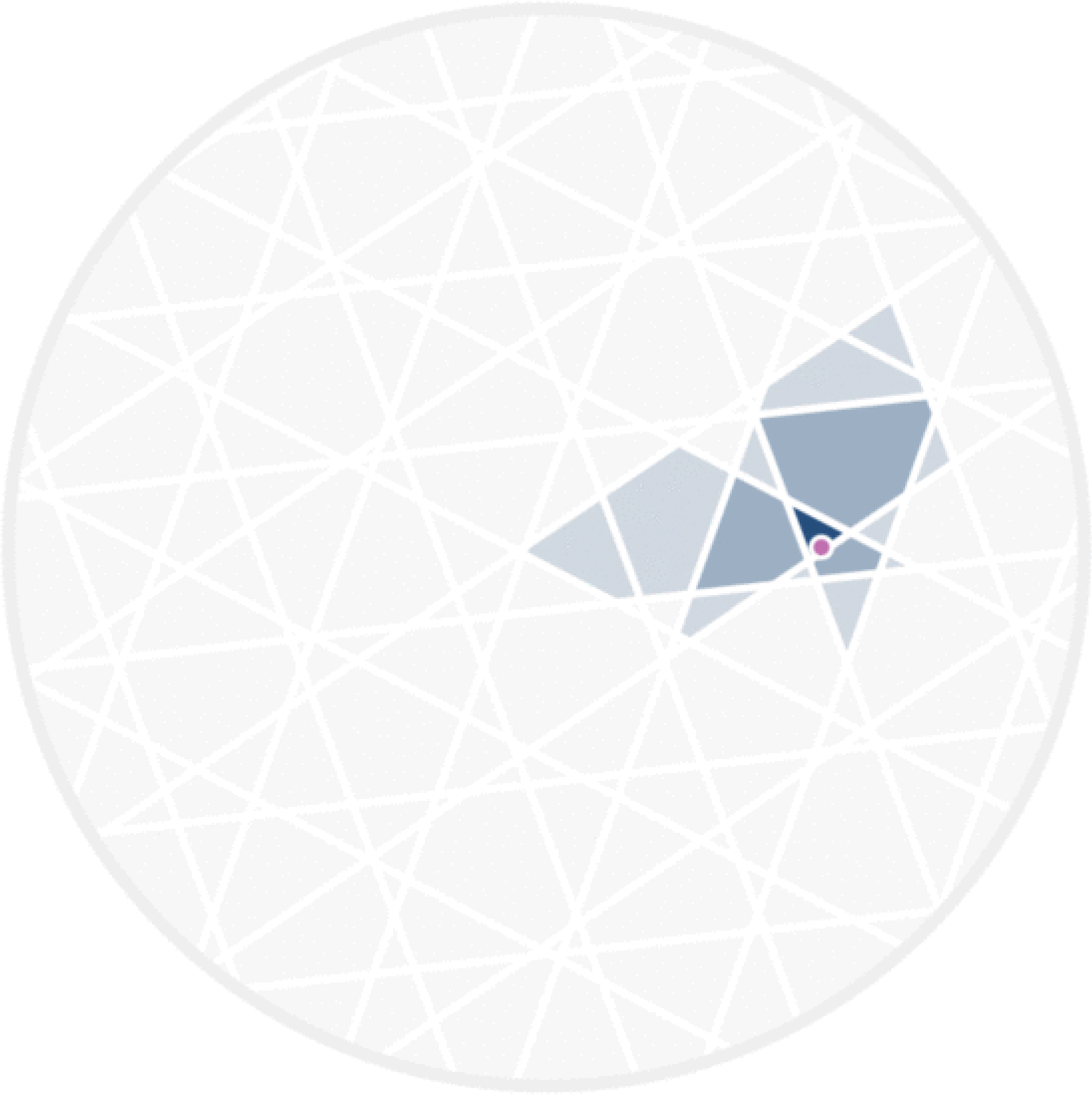
Proyección *aleatoria*



Proyección *aleatoria*

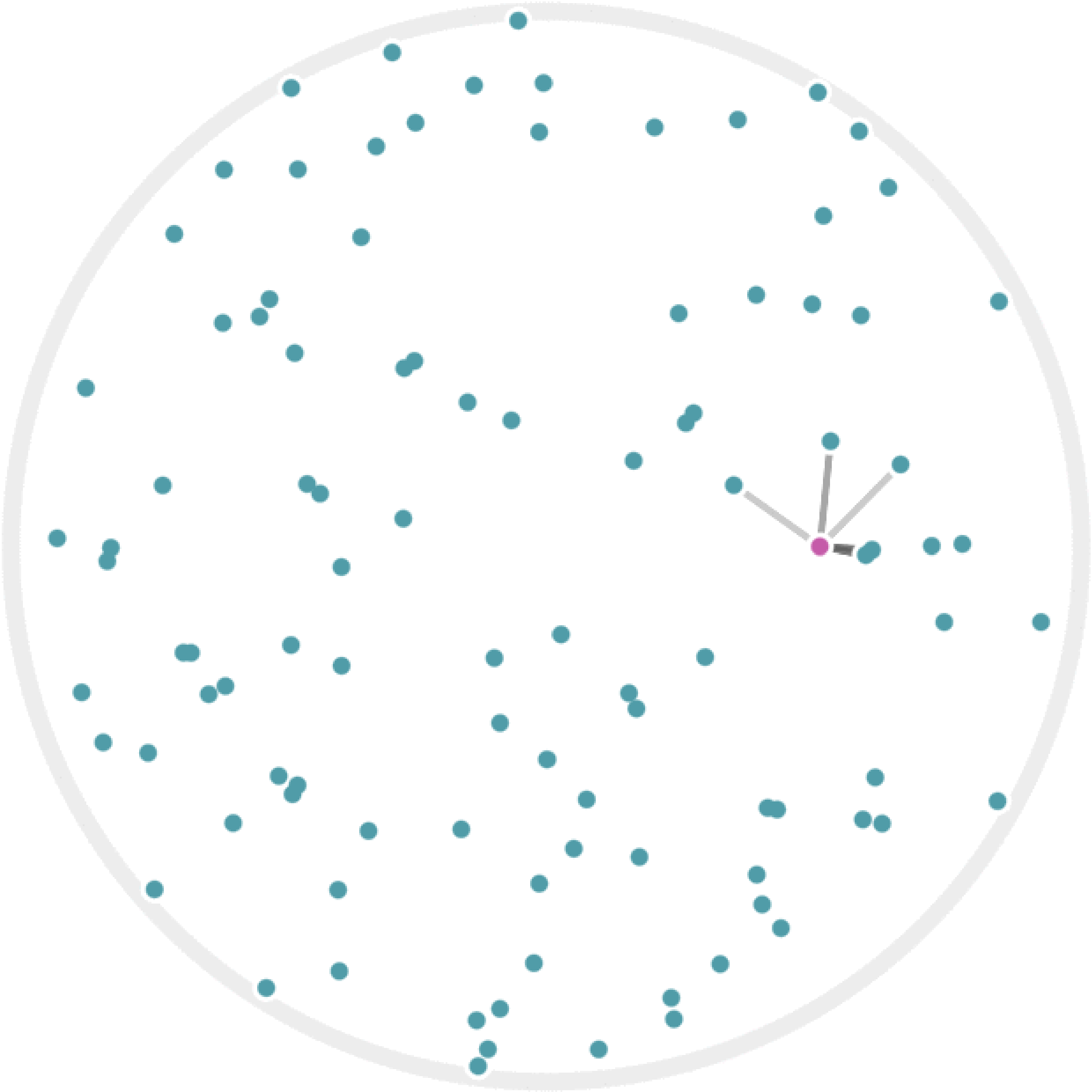


$k = 2$



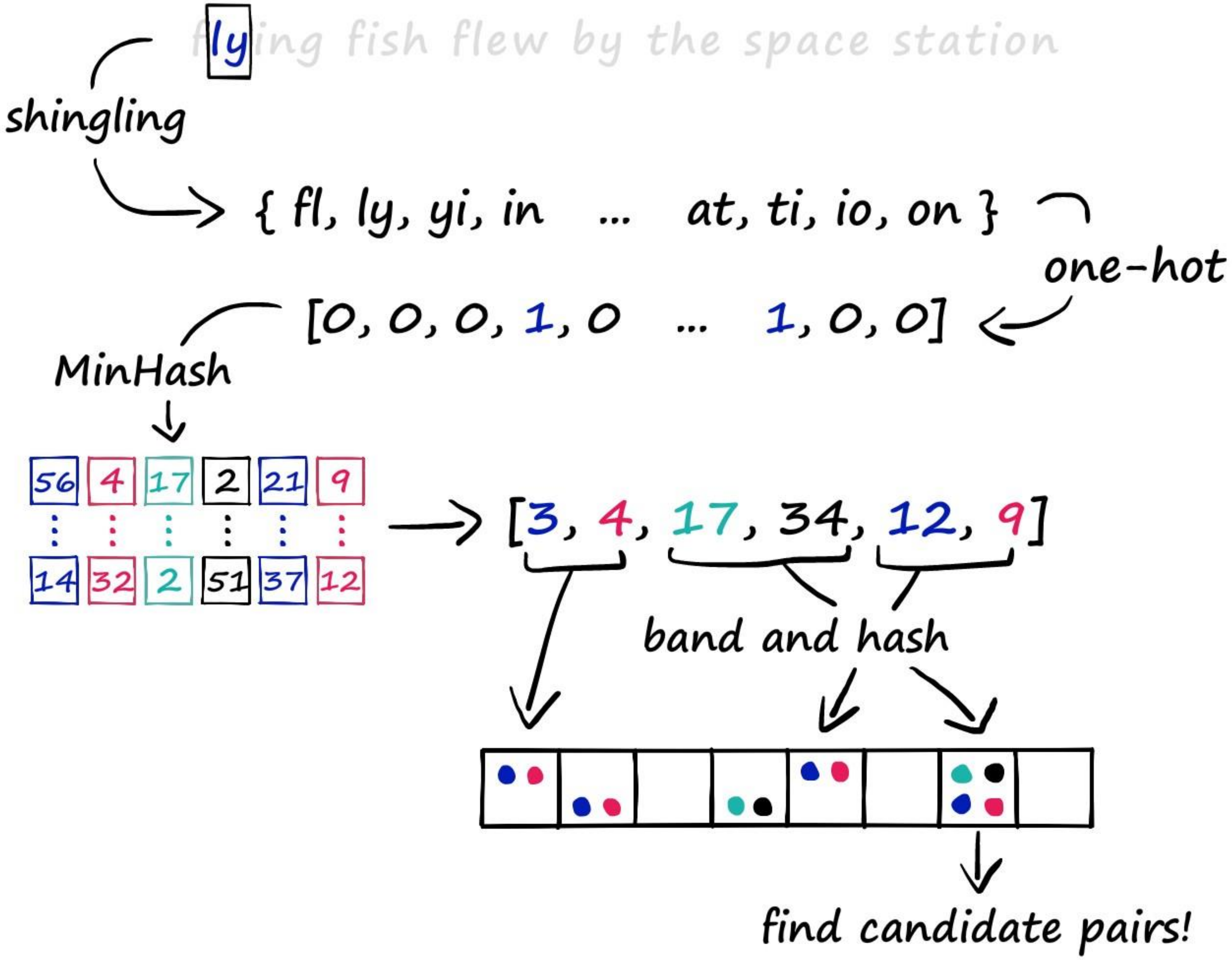
$k = 5$

Proyección *aleatoria*



2. MinHashing

Buscar frases *similares*

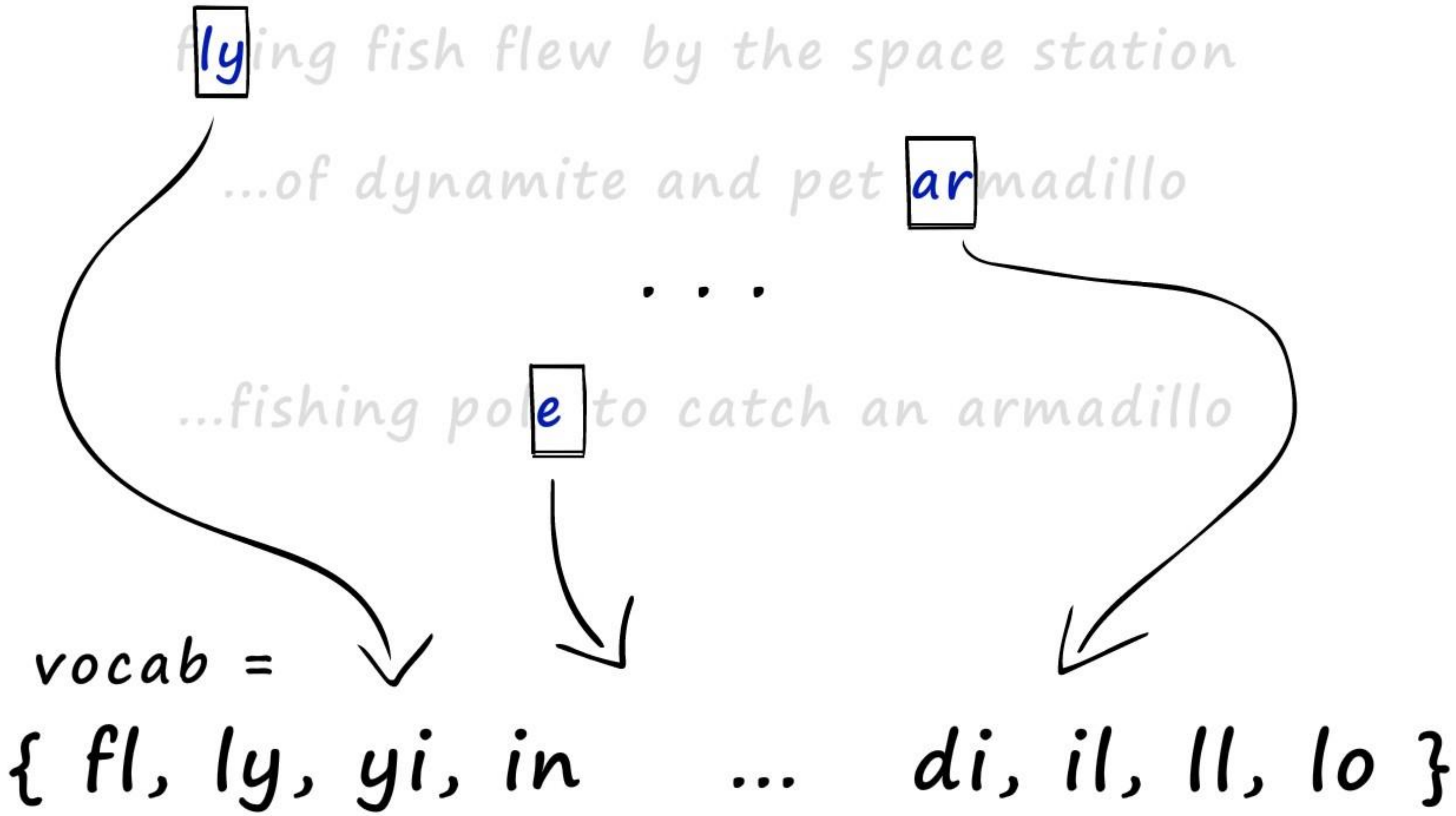


Gabriela Hristescu and Martin Farach-Colton (1999) "Cluster-preserving embedding of proteins".
Technical Report 99-50, Computer Science Department, Rutgers University

k-*shingling*

flying fish flew by the space station

Vocab*ulario*



flying fish flew by the space station

...of dynamite and pet armadillo

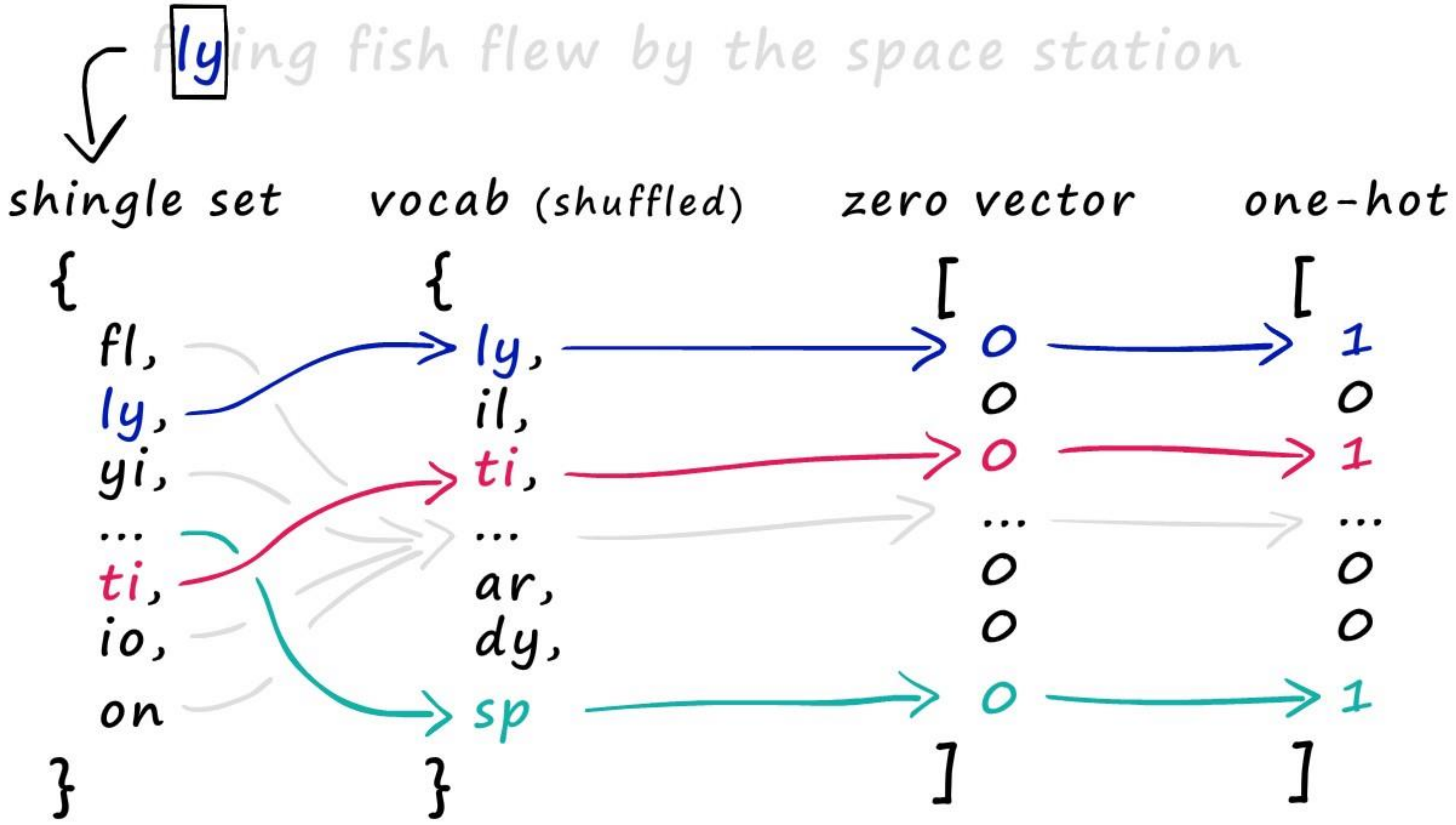
. . .

...fishing pole to catch an armadillo

vocab =

{ fl, ly, yi, in ... di, il, ll, lo }

Vocab*ulario*


flying fish flew by the space station
ly
shingle set
vocab (shuffled)
zero vector
one-hot
{ fl, ly, yi, ..., ti, io, on }
{ ly, il, ti, ..., ar, dy, sp }
[0 0 0 ... 0 0 0]
[1 0 1 ... 0 0 1]

Min*Hashing*

1

0

0

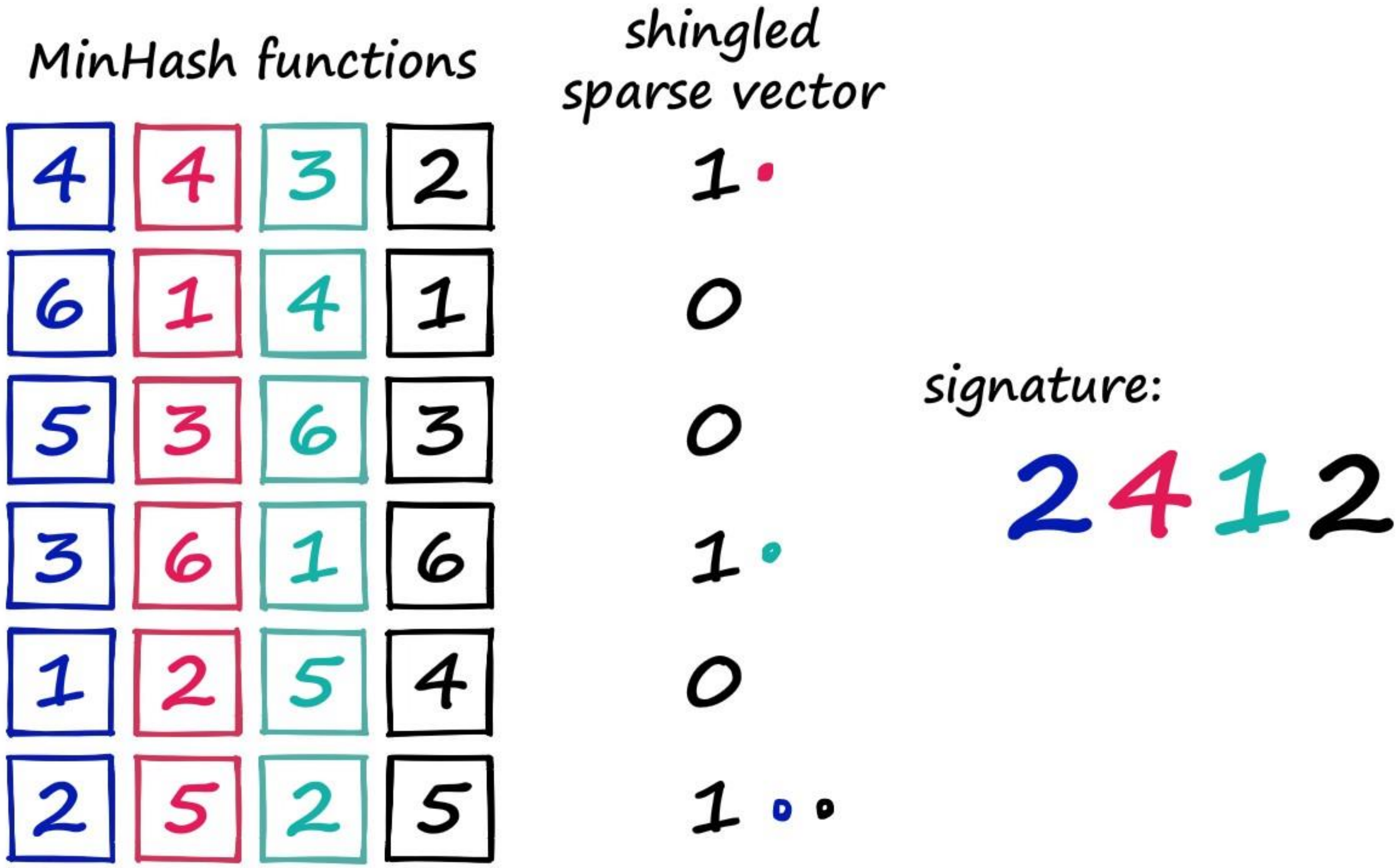
1

0

1

signature:

Min*Hashing*

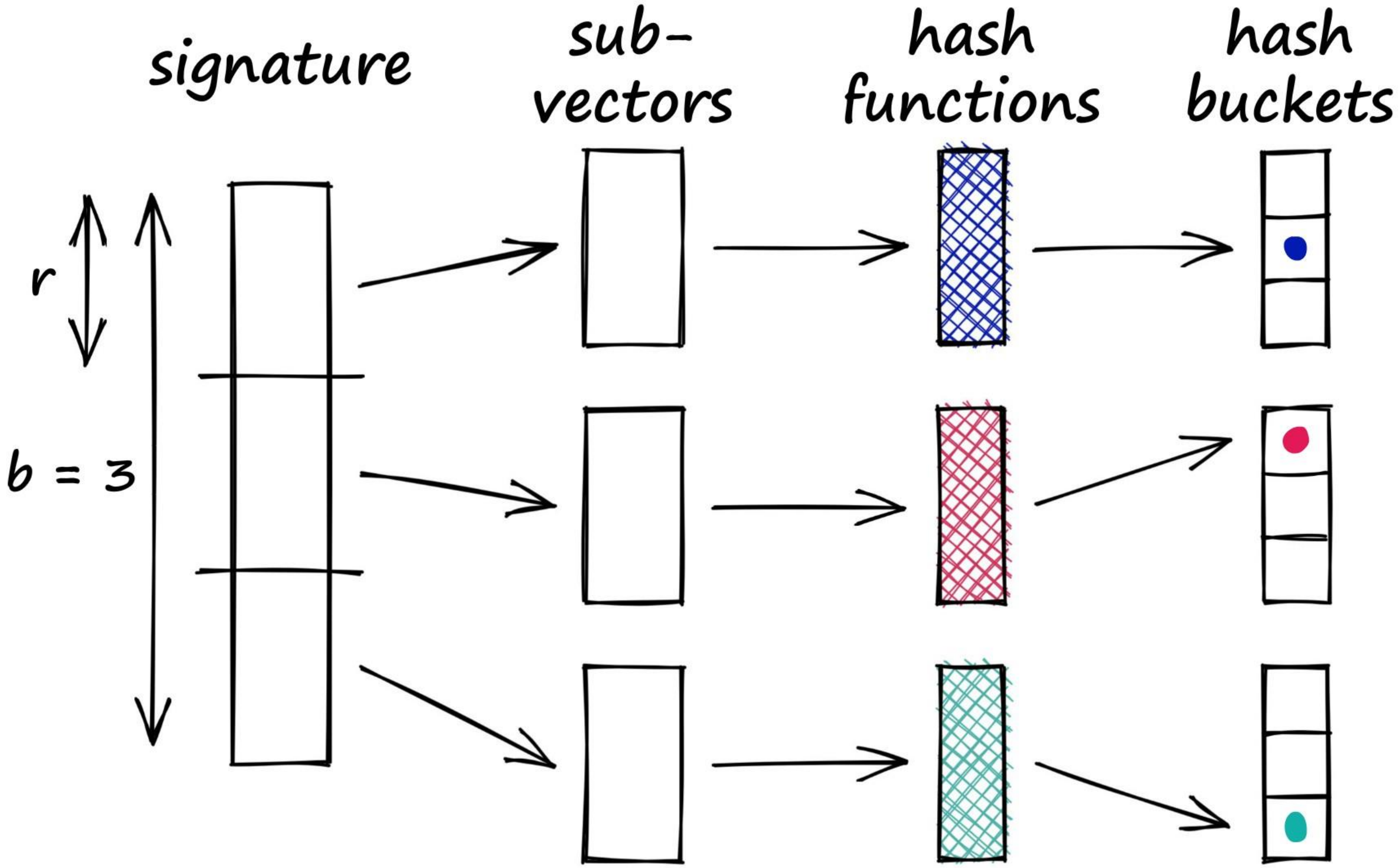


MinHash functions				shingled sparse vector
4	4	3	2	1
6	1	4	1	0
5	3	6	3	0
3	6	1	6	1
1	2	5	4	0
2	5	2	5	1

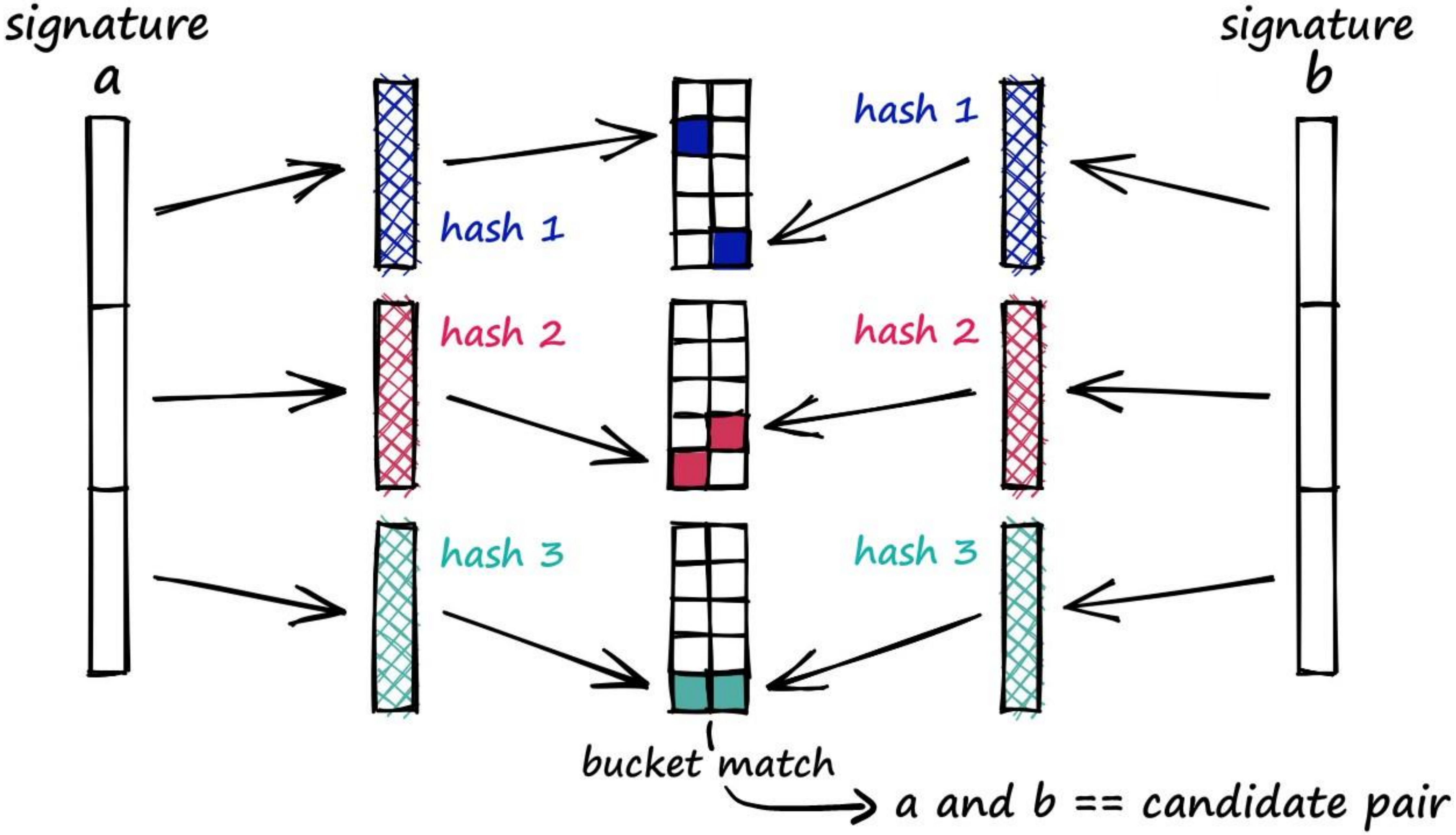
signature:

2412

Band *method*

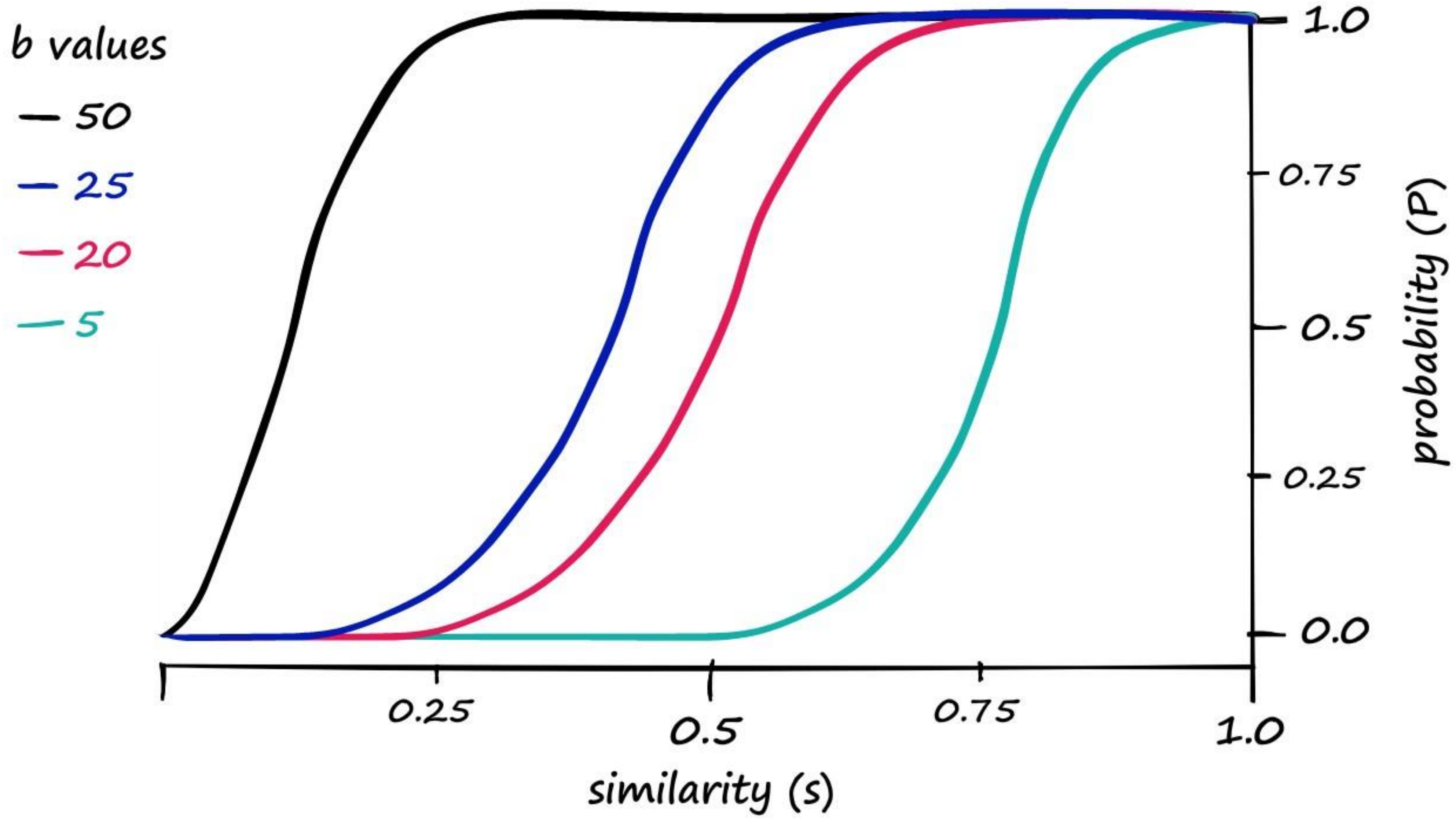


Band *method*


signature a
hash 1
hash 2
hash 3
hash 1
hash 2
hash 3
signature b
bucket match
a and b == candidate pair

Band *method*

$$P = 1 - (1 - s^r)^b$$

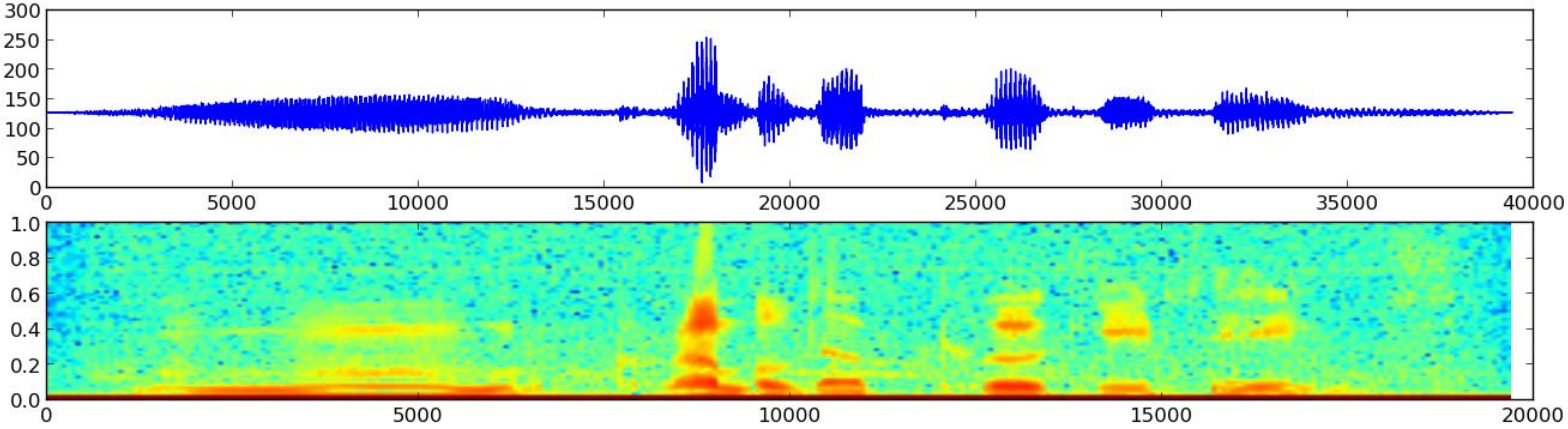


3. Shazam!

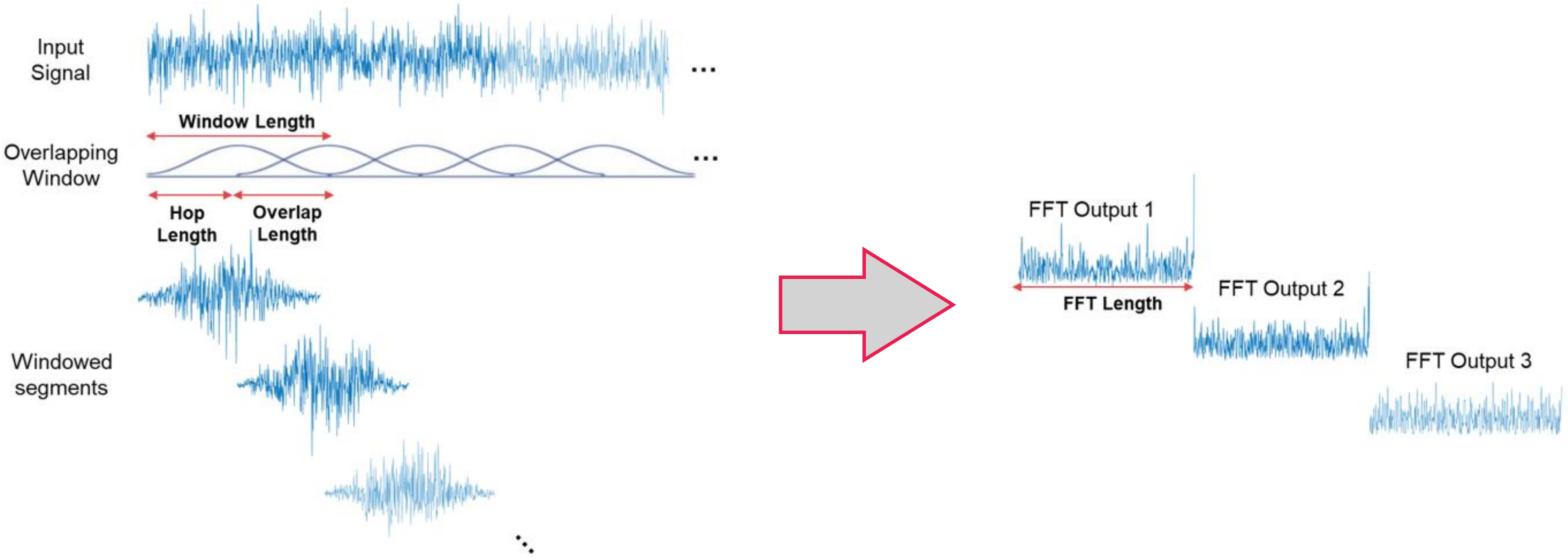
Sha*zam*

Avery Li-Chun Wang (2003) "An Industrial-Strength Audio Search Algorithm".
En Ismir. 2003. p. 7-13.

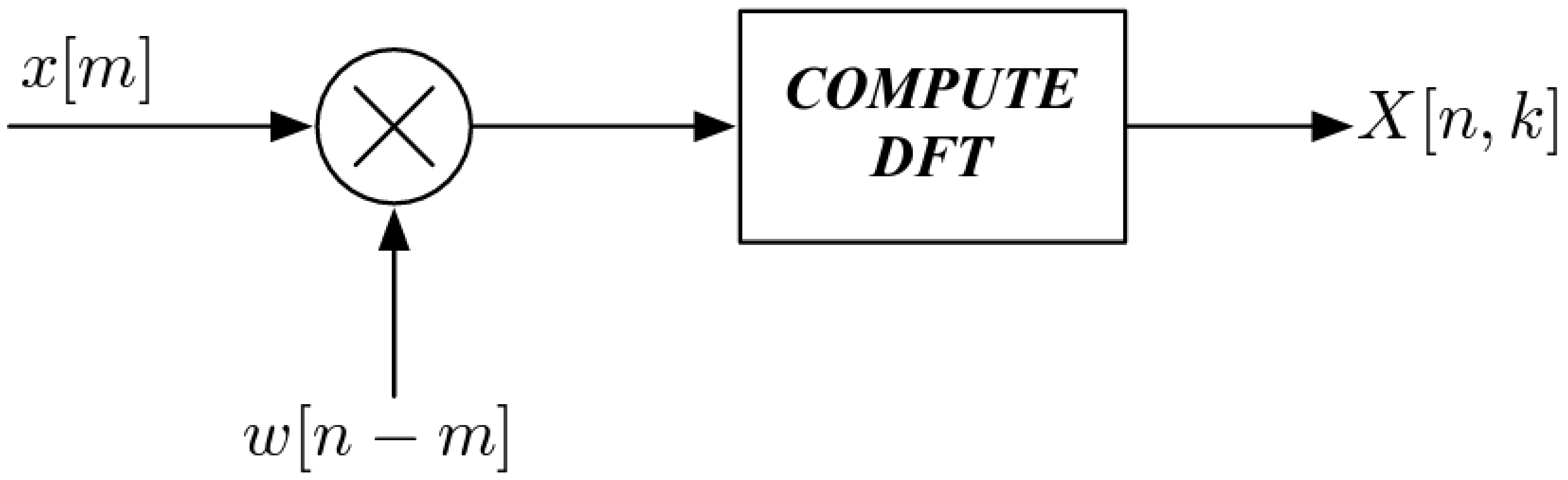
Dominio de *Frecuencia*



Short-Time *Fourier Transform*

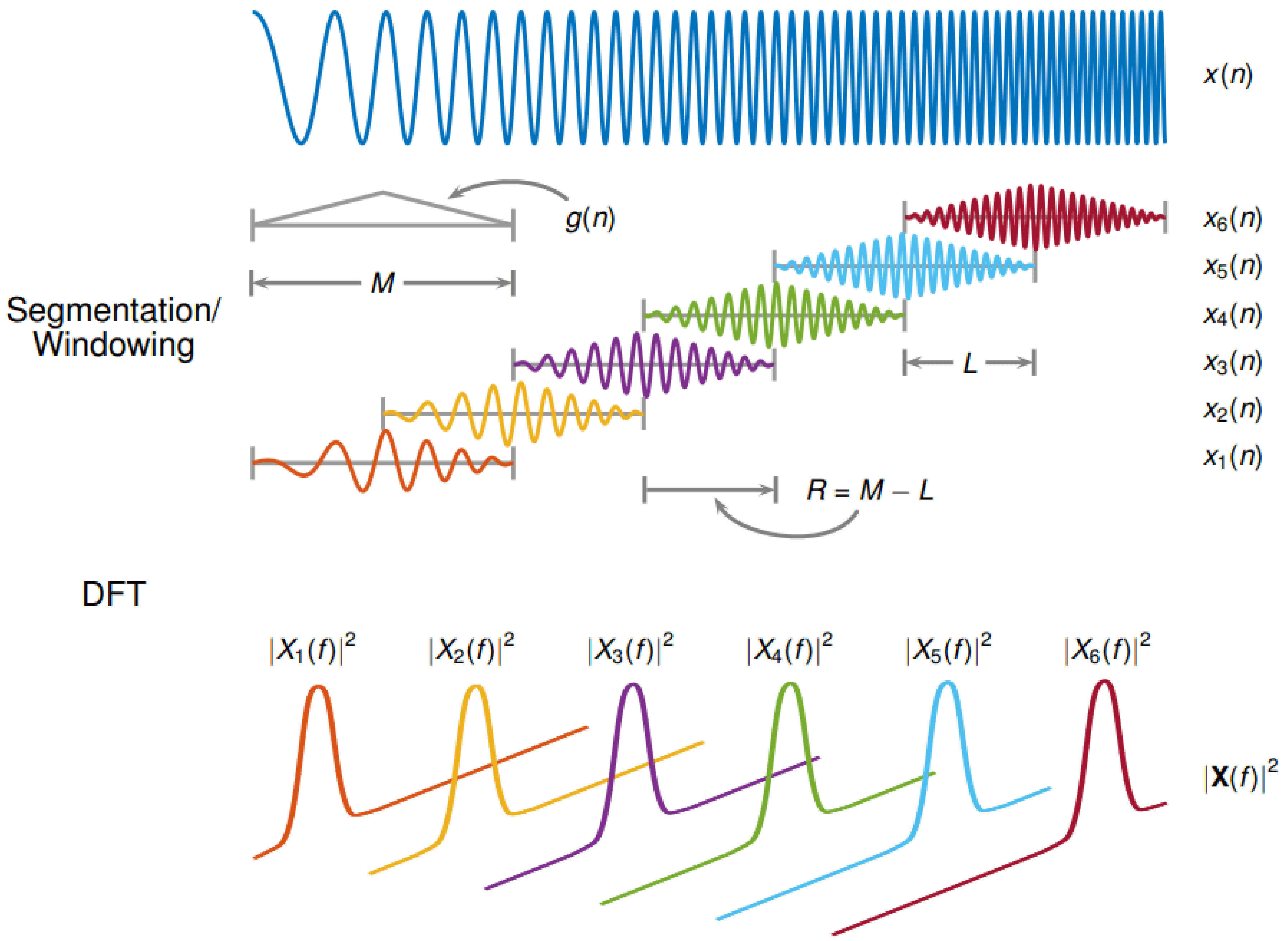


Short-Time *Fourier Transform*



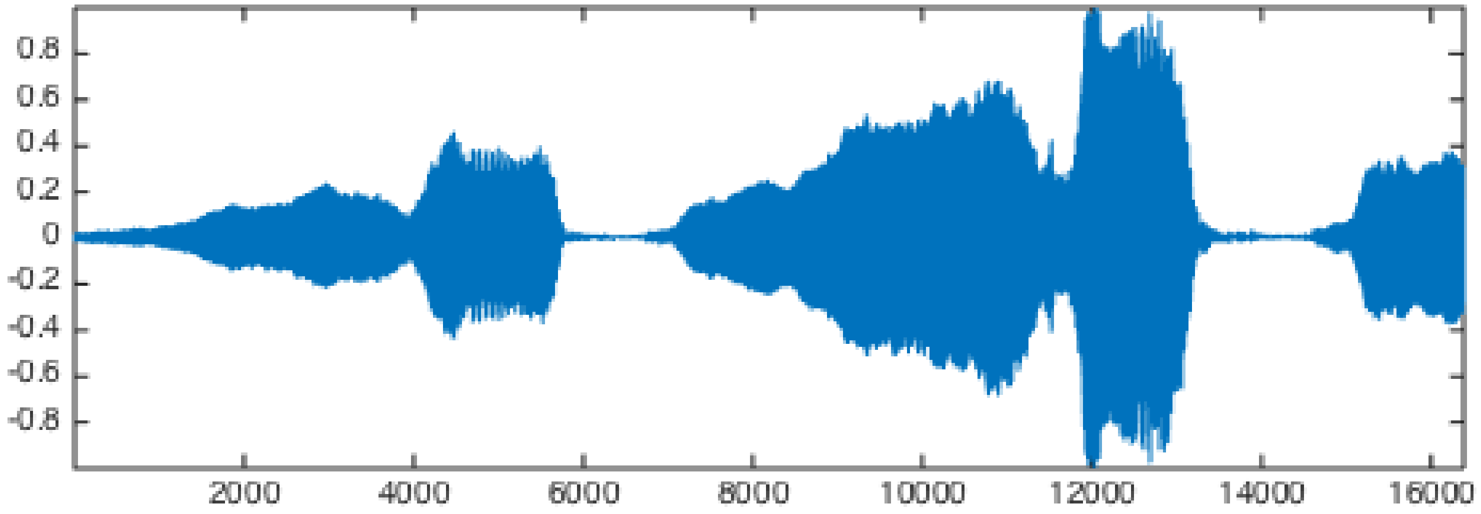
$$X[n,k] = \sum_{m=n-(N_w-1)}^{n} (x[m]w[n-m])e^{-j2\pi mk/N} = \sum_{m=n-(N_w-1)}^{n} (x[m]w[n-m])e^{-j\omega_k m}$$

Short-Time *Fourier Transform*

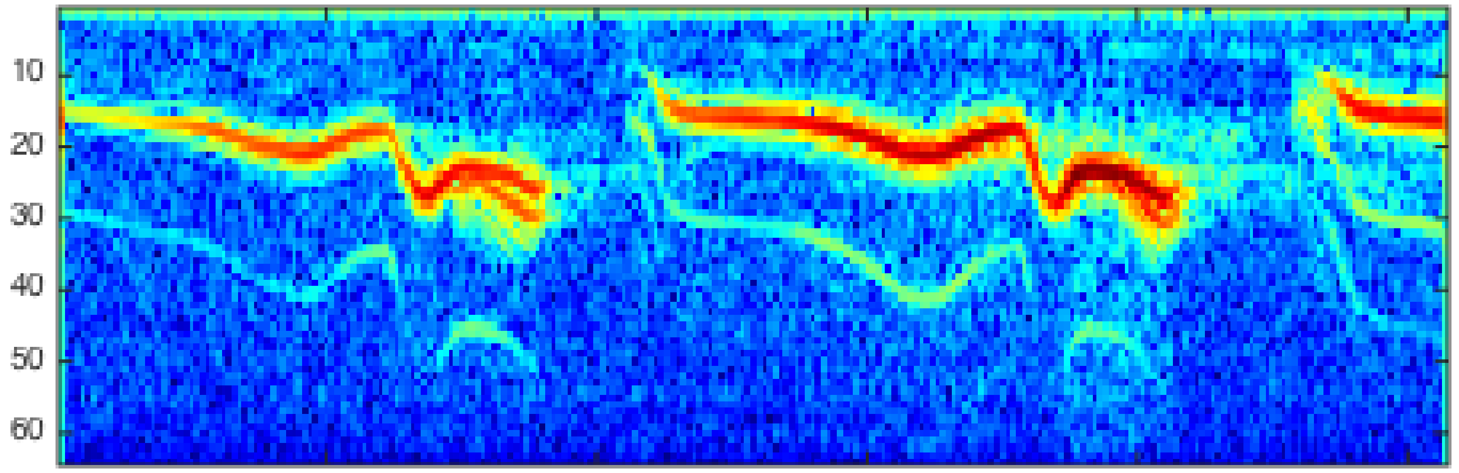


Short-Time *Fourier Transform*

Dominio de Tiempo



Dominio de Frecuencia



UTEC

UNIVERSIDAD DE INGENIERÍA
Y TECNOLOGÍA