Reminder: project proposals due Monday

Last time:

Lossy compression:
- Bloom filters
- Heavy hitters & count-min sketch

Key idea:

use magic of hashing and sacrifice a little bit of correctness ⟹ significant space savings

Today

- short reviews from probability
  - variance & tail bounds
  - Gaussians & CLT
- Distinct elts
- Similarity search & dimension reduction

Distinct Elements

$a_1, a_2, a_3, \ldots\ldots$ each $a_i \in U$

goal: maintain approx count of number of distinct elts seen.

n_t : # distinct elts seen in $a_1, a_2, \ldots, a_t$

do this approximately.

$h : U \longrightarrow [0,1]$

$\tilde{h} : U \to \{0, 1, \ldots, m-1\}$

$\tilde{h}(x) = i$

set $h(x) = \frac{i}{m}$

track $Y = \min_{1 \le i \le t} h(a_i)$

	a_1	a_2	a_3	. . .	–	–	–	–
	32	5	17	32	14	5	17	5 . . .
Y	0.43	0.43	0.19	0.19	0.19	0.19	0.19	0.19

$h(32) = 0.43$

$h(5) = 0.61$

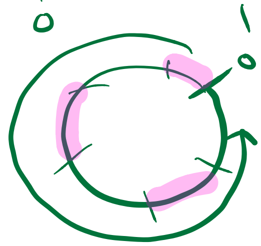
$h(17) = 0.19$

$h(14) = 0.85$

suppose that at time t, seen n distinct elts.

$$\boxed{E(Y)} = E[\min \text{ of } n \text{ hash values}]$$

$$= \frac{1}{n+1}$$

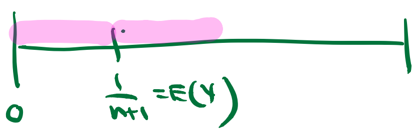


if magically

$Y = \frac{1}{n+1}$ not true.

$n+1 = \frac{1}{Y}$

estimate $n = \frac{1}{Y} - 1$



$$\mathrm{Var}(Y) \approx \frac{1}{(n+1)^2}$$

$$\sigma(Y) = \frac{1}{n+1}$$

	a_1	a_2	a_3	$\cdots$					
	32	5	17	32	14	5	17	5	$\cdots$
Y_1	0.43	0.43	0.19	0.19	0.19	0.19	0.19	0.19	h_1
Y_2									h_2
$\vdots$									
Y_k									h_k

$$Y_j = \min_{1 \le i \le t} h_j(a_i) \qquad E[Y_j] = \frac{1}{n+1}$$

$$Var(Y_j) = \frac{1}{(n+1)^2}$$

$$\bar{Y} = \frac{1}{k} \sum_{j=1}^{k} Y_j$$

$$E(\bar{Y}) = \frac{1}{n+1} \qquad Var(\bar{Y}) = \frac{1}{k} \frac{1}{(n+1)^2}$$

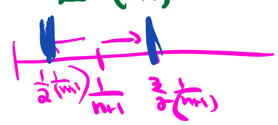
$$Var(\bar{Y}) = \frac{1}{k^2}\left(k \; Var(Y_1)\right)$$

$$= \frac{1}{k} \frac{1}{(n+1)^2}$$

$$k = 16 \qquad \sigma^2 = \frac{1}{16} \cdot \frac{1}{(n+1)^2}$$

$$\sigma = \frac{1}{4} \cdot \frac{1}{(n+1)}$$

$$2\sigma = \frac{1}{2} \frac{1}{(n+1)}$$



$$Pr\left(\left|\bar{Y} - \frac{1}{n+1}\right| > 2\sigma\right) \le \frac{1}{4} \leftarrow$$

μ

w.p. $\ge \frac{3}{4}$

$$\frac{1}{2}\frac{1}{(n+1)} \le \bar{Y} \le \frac{3}{2}\frac{1}{(n+1)}$$

$$\frac{1}{2\bar{Y}} \le n+1 \le \frac{3}{2}\frac{1}{\bar{Y}}$$

$$Pr(|Y - \mu| \ge c\sigma) \le \frac{1}{c^2}$$

$$\frac{1}{\bar{Y}} \le 2(n+1) \qquad \frac{2}{3}(n+1) \le \boxed{\frac{1}{\bar{Y}}}$$

Similarity search:

dataset, notion of similarity between items in set.

Items	Goal
documents	plagiarism detection
	similar topic
web pages	mirror sites
DNA sequences	find similar genes
movie tastes	similar
ML-classification	

What is an item (data pt)?

vector in high-dimensional space $\mathbb{R}^k$

k very large.

$(w_1, w_2, \ldots, w_N)$ bag of words

vector representing pixel values image

Similarity

1. ℓ_2 distance $\quad d(x,y) = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2}$

$x, y \in \mathbb{R}^k \quad x = (x_1, x_2, \ldots, x_k)$

n vectors, $\in \mathbb{R}^k$

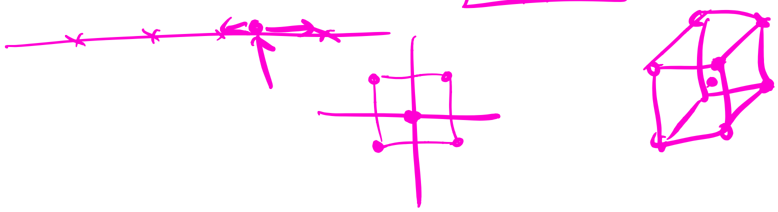
compute all distances: $O(n^2 \cdot k)$

Preprocess a data set consisting of n vectors & then be able to quickly answer queries
What's the closest pt to $q \in \mathbb{R}^k$

preprocessing k-d trees.

curse of dimensionality.

log n



Dimension reduction:

$x_1, \dots, x_n \in \mathbb{R}^k$ $\quad d \ll k$

$$f: \mathbb{R}^k \longrightarrow \mathbb{R}^d$$

so that distances between pts are almost exactly preserved.

- compression of data
- visualization

$$\underset{\in \mathbb{R}^k}{x_1, \dots, x_n} \longrightarrow \underset{\in \mathbb{R}^d}{f(x_1) \dots f(x_n)} \longleftarrow d \approx \log n$$

$\forall i,j \quad \mathrm{dist}(x_i, x_j) \approx \mathrm{dist}(f(x_i), f(x_j))$ w.h.p.

r at random

$$\begin{array}{ccc} \mathbb{R}^k & \longrightarrow & \mathbb{R} \\ x & \longrightarrow & r\cdot x \\ y & \longrightarrow & r\cdot y \end{array}$$

$\to f_r(x) = r\cdot x = \sum_{i=1}^{k} r_i x_i$

$\to r = (r_1, \dots, r_k)$ $\quad x = (x_1, \dots, x_k)$

$r_i \sim N(0,1)$ indep.

$$f_r(x) - f_r(y) = r\cdot x - r\cdot y = r\cdot(x-y) = \boxed{\sum_{i=1}^{k} \overset{N(0,1)}{r_i}\, \underbrace{(x_i - y_i)}_{a_i}}$$

$\|x-y\| = \sqrt{\sum (x_i - y_i)^2}$

$$N\left(0, \sum_{i=1}^{k} (x_i - y_i)^2\right)$$

$$= N\left(0, \|x-y\|^2\right)$$

$Z = \sum_{i=1}^{k} a_i Z_i$

$Z_i \sim N(\mu_i, \sigma_i^2)$

$Z \sim N\left(\sum a_i \mu_i, \sum a_i^2 \sigma_i^2\right)$

$$E\left[\boxed{(r\cdot x - r\cdot y)^2}\right] = \|x-y\|^2$$

$$E\left[(f_r(x) - f_r(y))^2\right]$$
$$= \mathrm{Var}\left(f_r(x) - f_r(y)\right)$$
$$= \|x-y\|^2$$

$$\mathrm{Var}(W) = E\left[(W - \overset{0}{\mu})^2\right] = E\left[W^2\right]$$

$(r\cdot x - r\cdot y)^2$ is an unbiased estimator for $\|x-y\|^2$

$r^{(1)}, r^{(2)}, \dots, r^{(d)}$ $\qquad r^{(i)} \in \mathbb{R}^k$

$r^{(i)}_j \sim N(0,1)$

$f_r(x) = r\cdot x$

$f_{r^{(i)}}(x) = c\, r^{(i)}\cdot x$

$$x \longrightarrow f(x) = \begin{pmatrix} c\, r^{(1)}\cdot x \\ c\, r^{(2)}\cdot x \\ \vdots \\ c\, r^{(d)}\cdot x \end{pmatrix}$$

$$f: \mathbb{R}^k \longrightarrow \mathbb{R}^d$$

$$\|f(x) - f(y)\|^2 = \sum_{i=1}^{d} \left(c\, r^{(i)}\cdot x - c\, r^{(i)}\cdot y\right)^2 = c^2 \sum_{i=1}^{d} \underbrace{\boxed{\left(r^{(i)}\cdot(x-y)\right)^2}}_{\left(f_{r^{(i)}}(x) - f_{r^{(i)}}(y)\right)^2}$$

$$f(y) = \begin{pmatrix} c r^{(1)} \cdot y \\ c r^{(2)} \cdot y \\ \vdots \\ c r^{(d)} \cdot y \end{pmatrix}$$

$$f(x) = \underset{\uparrow}{A}\, x$$

$$\frac{1}{\sqrt{d}} \begin{bmatrix} \text{---} r^{(1)} \text{---} \\ \text{---} r^{(2)} \text{---} \\ \vdots \\ \text{---} r^{(d)} \text{---} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_k \end{bmatrix} = \frac{1}{\sqrt{d}} \begin{pmatrix} r^{(1)} \cdot x \\ r^{(2)} \cdot x \\ \vdots \\ r^{(d)} \cdot x \end{pmatrix}$$

$\underline{d} \times k$ matrix

$$E\left(\|f(x) - f(y)\|^2 \right) = \underbrace{c^2 \sum_{i=1}^{d} \underbrace{E\left[\left(r^{(i)} \cdot (x-y) \right)^2 \right]}_{\|x-y\|^2}}$$

$$= c^2 \cdot d \cdot \|x-y\|^2$$

want this $= \|x-y\|^2$

so should set $\boxed{c = \frac{1}{\sqrt{d}}}$

If you choose $d \approx O\left(\frac{\log n}{\varepsilon^2} \right)$

then $\forall$ 2 pts in dataset of size n.
$\underline{x, y}$

$\underline{\|f(x) - f(y)\| \in (1 \pm \varepsilon) \|x-y\|}$

w.h.p.