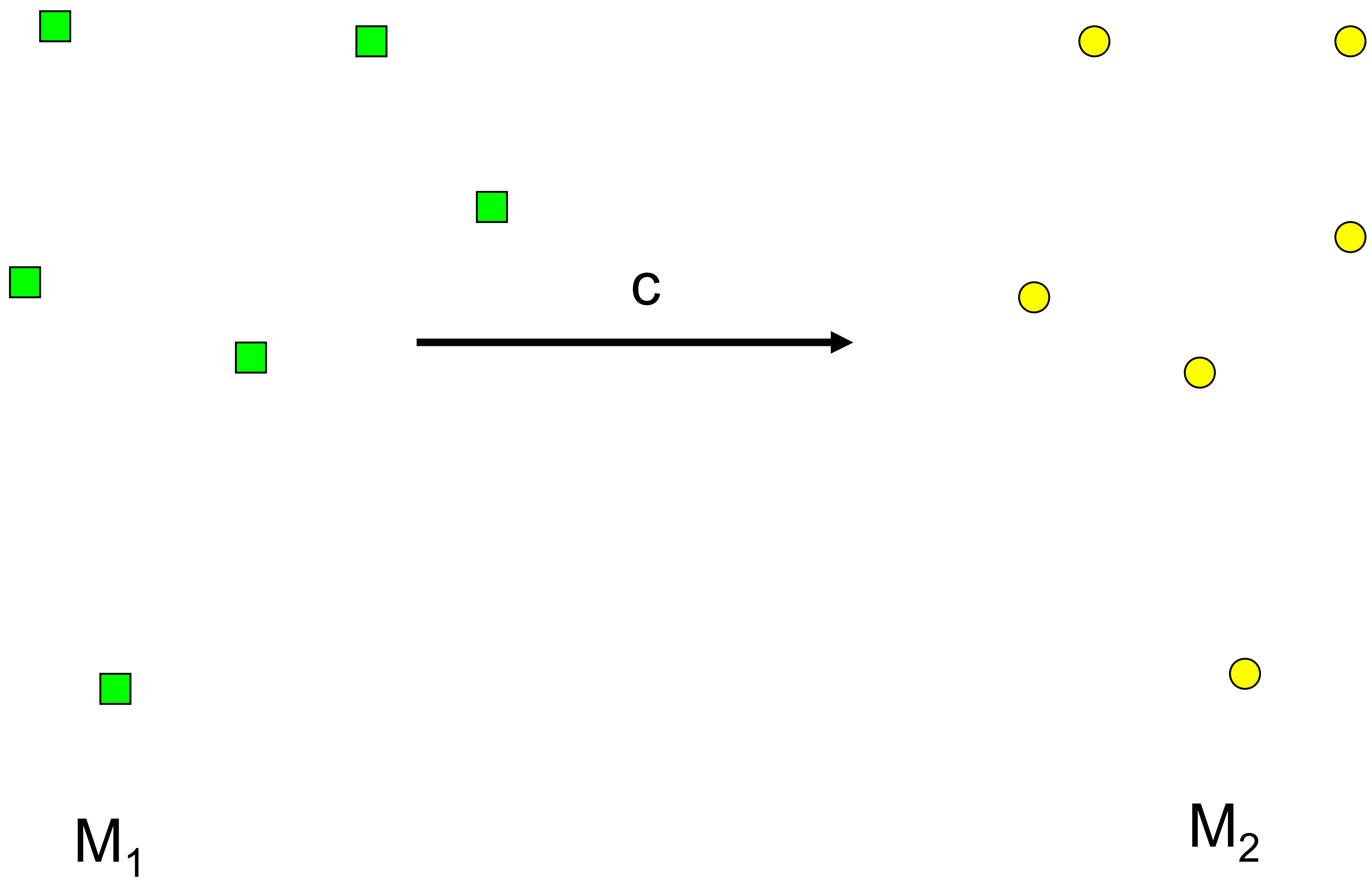
Algorithmic Applications of Low-Distortion Embeddings

Piotr Indyk
MIT

Embeddings



Embeddings

- Given $M_1=(X_1,D_1)$, $M_2=(X_2,D_2)$
- A mapping $f: X_1 \rightarrow X_2$, such that $\forall p,q \in X_1$:

$$D_1(p,q) \leq D_2(f(p),f(q)) \leq c^*D_1(p,q)$$

is called a c-embedding of M_1 into M_2

- The c-embedding definition composes:
  If M_1 c_1-embeds into M_2, and
  M_2 c_2-embeds into M_3, , then
  M_1 c_1c_2-embeds into M_3

Metrics/Norms 101

- Metric $M=(X,D)$:
  - Reflexive: $D(p,q)=0$ iff $p=q$
  - Symmetric: $D(p,q)=D(q,p)$
  - Triangle ineq.: $D(p,q) \leq D(p,t) + D(t,q)$
- Norms over R^d:
  - L_s norm: $||x||_s = (\Sigma_i |x_i|^s)^{1/s}$
  - L_∞ norm: $||x||_\infty = \max_i |x_i|$
- Norm induces a metric: $D(p,q)=||p-q||_s$
- Use l_s^d to denote (R^d,l_s)

Outline

- Brief history of embeddings
  - Major results
  - Impact on TCS
- Dimensionality reduction: Johnson-Lindenstrauss Theorem
  - Theorem + construction
  - Inspirations: Locally-Sensitive Hashing for Approx Near Neighbor
- Metrics for computer vision: Earth-Mover Distance
- Conclusions and Resources

Very Brief History of Embeddings

- [Frechet, 1909]:
  Any metric (X,D), $|X|=n$, is 1-embeddable into l_∞^n

- Proof:
  Let $X=\{p_1,\ldots,p_n\}$. Define the mapping f as:

$$f(p)=[\ D(p,p_1),\ D(p,p_2),\ \ldots\ ,D(p,p_n)\ ]$$

- Then $\|f(p)-f(q)\|_\infty = \max_i |D(p,p_i)-D(q,p_i)|$
  – Non-expansion: $\leq D(p,q)$
  – Non-contraction: $\geq |D(p,p) - D(q,p)| = D(q,p)$

Brief History ctd.

[Bourgain'85]:

Any (X,D) is O(log n)-embeddable into l_2^k

- The dimension k can be made O(log n) (next slide)
- Technique: generalization of Frechet
- Proof gives a randomized $O(n^2 \log^2 n)$ algorithm [Linial-London-Rabinovich'95]

Brief History ctd.

[Johnson-Lindenstrauss'84]:

For any $X \subseteq l_2^d$, there is a $(1+\varepsilon)$-embedding of (X,l_2) into $l_2^{d'}$, where $d'=O(\log n/\varepsilon^2)$

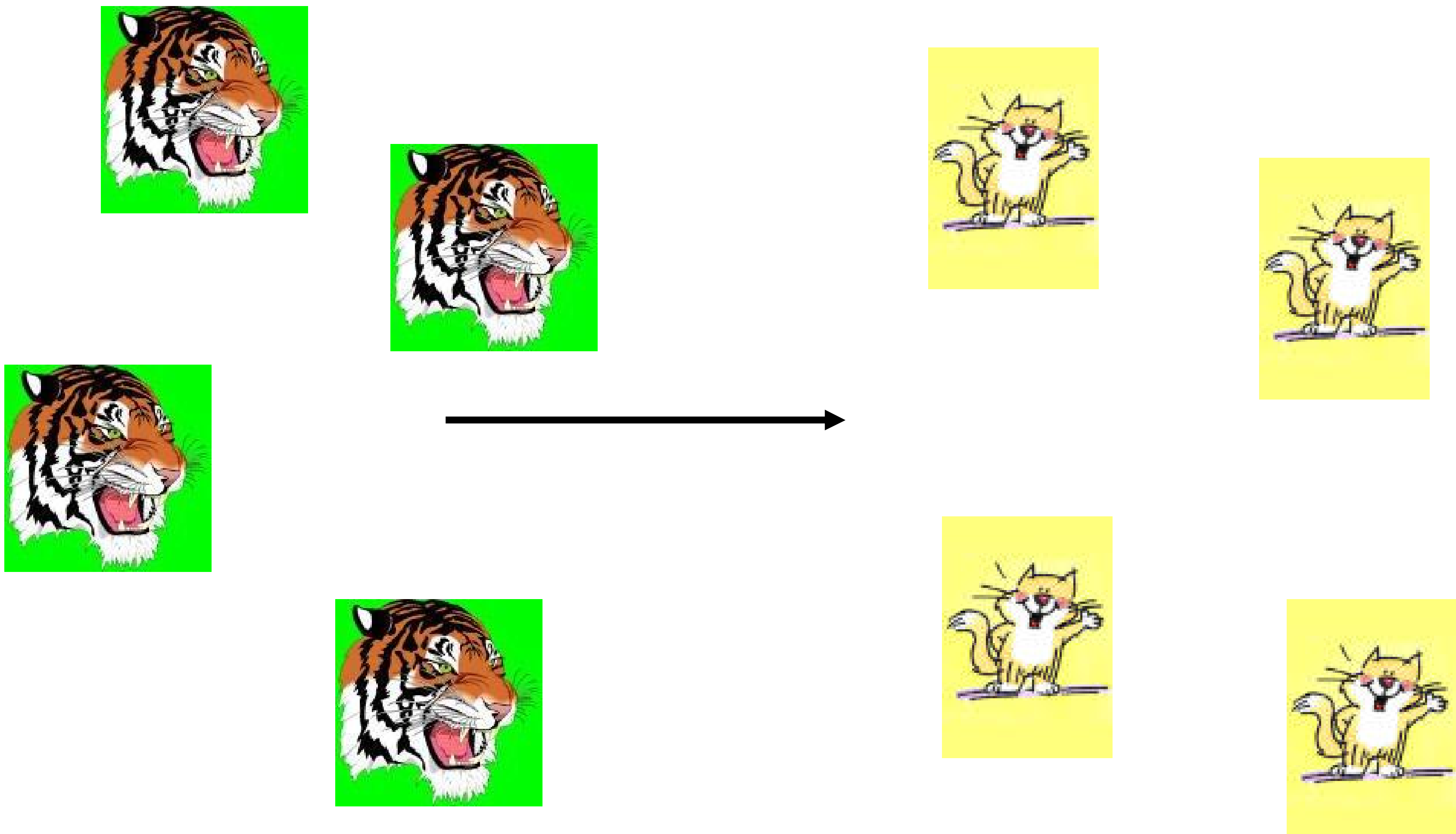
Brief History - Algorithms

- [Linial-London-Rabinovich’95]:
  - Used Bourgain’s theorem to get an approximation algorithm for the sparsest cut problem
  - Introduced the notion of embeddings to CS community

Brief History – Algorithms ctd.

- Probabilistic embeddings of general metrics into trees [Alon-Karp-Peleg-West'91, Bartal'96 '98, Fakcharoenphol-Rao-Talwar'03]
  - Applications to combinatorial optimization problems
- Dimensionality reduction:
  - Approximate nearest neighbor algorithms with polynomial space [Kleinberg'97, Kushilevitz-Ostrovski-Rabani'98, Indyk-Motwani'98, Indyk'00, Datar-Immorlica-Indyk-Mirrokni'04]
  - Algorithms for streaming data [Alon-Matias-Szegedy'96, Indyk'00, GGIKMS'02, Indyk'04]
- ...
- Machine learning: PCA, MDS [Kruskal], LLE [Roweis-Saul'00], Isomap [Tenenbaum-da Silva-Langford'00]

Embeddings for Algorithms



In This Talk

- Dimensionality reduction: techniques and inspirations
- Earth-Mover Distance (EMD) into l_1

Dimensionality Reduction

Randomized Dim Reduction

JL Theorem: For any $X \subseteq l_2^d$, there is a $(1+\varepsilon)$-embedding of (X, l_2) into $l_2^{d'}$, where $d' = A \ln n/\varepsilon^2$ $(A=4)$

Proof: For a linear mapping $f(p)=Ap$, where A is a $d' \times d$ "random" matrix, we have for any p,q in X

$$\Pr[\,|\,\|Ap-Aq\|_2 - \|p-q\|_2\,| > \varepsilon\|p-q\|_2\,] \le e^{-\Omega(d'/\varepsilon^2)}$$

- Choices of A:
  - Rows: random orthogonal unit vectors [JL'84]
  - Rows: random unit vectors
  - Entries: independently chosen from $N(0,1)$
  - Entries: independently chosen from $\{-1,1\}$ [Achlioptas'00]
  - ….

Proof

- We map $f(u)=Au=[a^1*u,\ldots,a^{d'}*u]$, where each entry of A has normal distribution
- Need to show that there exists scaling factor S such that, with probability at least ½, for each pair p,q in X, we have $||f(p)-f(q)|| \approx S\,||p-q||$
- Sufficient to show that for a *fixed* $u=p-q$, where p,q in X, we have $||Au|| \approx S||u||$ with probability at least $1-1/n^2$
- In fact, by linearity of A we can assume $||u||=1$, so we just need to show $||Au|| \approx S$

Normal Distribution

- Normal distribution:
  - Range: $(-\infty, \infty)$
  - Density: $f(x)=e^{-x^2/2} / (2\pi)^{1/2}$
  - Mean=0, Variance=1
  - If X and Y independent r.v. with normal distribution, then X+Y has normal distribution
- Basic facts:
  - $Var(cX)=c^2 \, Var(X)$
  - If X,Y independent, then $Var(X+Y)=Var(X)+Var(Y)$

Back to embedding

- Consider $Z=a^i*u = a*u=\sum_i a_i u_i$
- Each term $a_i u_i$
  - Has normal distribution
  - With variance u_i^2
- Thus, Z has normal distribution with variance $\sum_i u_i^2 = 1$
- This holds for each a^j

What is $||Au||_2$

- $||Au||^2 = (a^1 * u)^2 + \ldots + (a^{d'} * u)^2 = Z_1^2 + \ldots + Z_{d'}^2$
  where:
  - All Z_i's are independent
  - Each has normal distribution with variance=1
- Therefore, $E[\ ||Au||^2\ ] = d' * E[Z_1^2] = d'$
- By Chernoff-like bound

$$\Pr[\ |\ ||Au||^2 - d'| > \varepsilon d'] < e^{-B\ d' \varepsilon^2} < 1/n^2$$

  for some constant B
- So, $||Au||_2 \approx (d')^{1/2}$ with probability $1-1/n^2$

Implications

- Replace d by $O(\ln(n)/\varepsilon^2)$ in the running time
- Works (w.h.p.) even if not all points known in advance. E.g., query point in nearest neighbor
- Mapping is linear

Experiments I

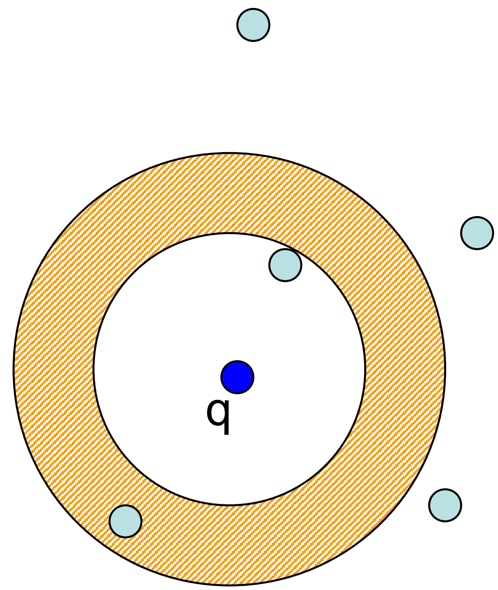
- [Dasgupta, UAI'00]: Compared JL with PCA in the context of supervised learning using EM (on OCR data set):
  - Reduce dimension
  - Run EM to fit a Gaussian mixture
  - Use it as a classifier
- Conclusions:
  - Reduction from 256 to 40 dim improved the accuracy (of both PCA and JL)

Experiments II

- [Fradkin-Madigan, KDD'03]: Compared JL with PCA in the context of supervised learning
  - Reduce the dimension
  - Apply C4.5, 1NN, 5NN or SVM
  - Measure the classification error
- Conclusions:
  - To reach optimal error, JL needs dimension that is {1, 10, 50} times larger than PCA
  - However:
    - JL needs no additional space (matrix A can be pseudo-generated), and has lower pre-computation time
    - JL needs no updating when new data points are added

Inspiration

- c-Approximate Near Neighbor:
  - Given: set P of points in l_2^d, $r>0$
  - Goal: build data structure which, for any query q, if there is a point $p \in P, ||q-p||_2 \leq r$, it returns $p' \in P$, $||q-p'||_2 \leq cr$



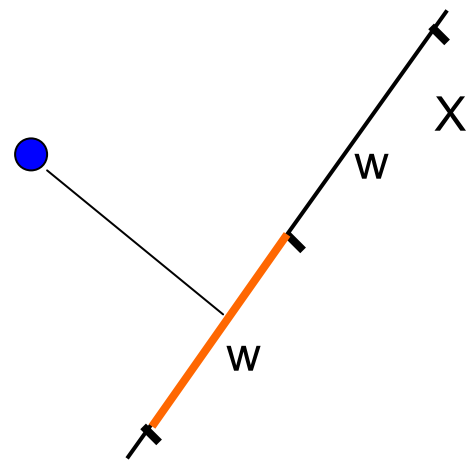
LSH

- A family H of functions $h: l_s^d \rightarrow U$ is called (P_1,P_2,r,cr)-sensitive [IM'98], if for any p,q:
  - if $||p-q||_s < r$ then $Pr[\, h(p)=h(q)\,] > P_1$
  - if $||p-q||_s > cr$ then $Pr[\, h(p)=h(q)\,] < P_2$
- Given H, we can solve a c-approximate NN with:
  - Query time: $O(d\, n^{\rho} \log n)$, $\rho = \log_{1/P2}(1/P_1)$
  - Space: $O(n^{\rho+1} + dn)$

LSH [DIIM'04]

Define $h_X(p)=\lfloor p*X/w \rfloor$, where:

- $w \approx r$
- $X=(X_1 \ldots X_d)$, where X_i is chosen from "stable" distribution
- I.e., $p*X$ has same distribution as $||p||\ Z$, where Z is "stable"
- For l_2, Gaussian distribution is stable



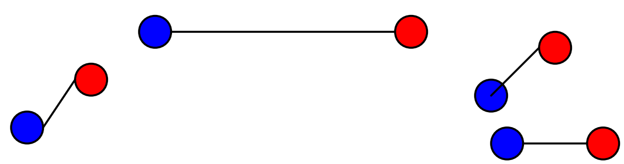
LSH [DIIM'04]

- Recall the query time is $O(dn^{\rho})$
- Bounds on ρ :
  - $\rho < 1/c$ for l_2 (improves on [IM'98])
  - $\rho \approx 1/c$ for l_1
- Works directly in l_s spaces (unlike [IM'98])

Earth Mover Distance

Earth-Mover Distance

- Given: two (multi)sets $P,Q \subseteq R^2$, $|P|=|Q|$
- $EMD(P,Q)$=min weight matching between P and Q



Applications

- A natural measure of dissimilarity between point-sets
- [Rubner-Tomasi-Guibas'98] used it for comparing
  - color histograms of images
  - texture information of images
  - …
- Experimentally works well

Issues

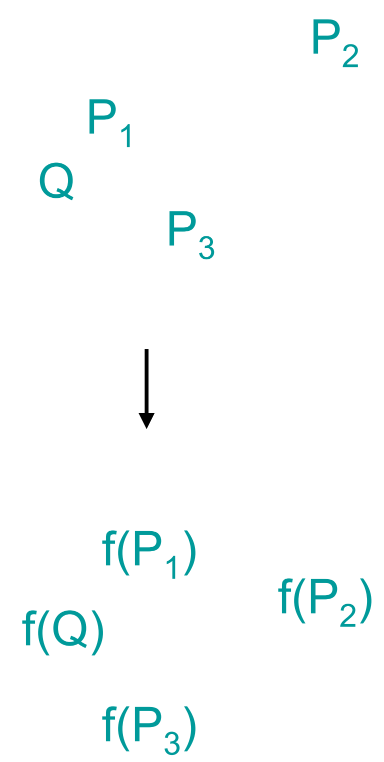
- EMD(P,Q) takes a super-linear (in |P|) time to compute
- Typically, one wants to find a NN of Q with respect to EMD
- How to do this faster than linear scan ?

Q

P_1
P_2
⋮
P_n

Approximate NN via Embeddings

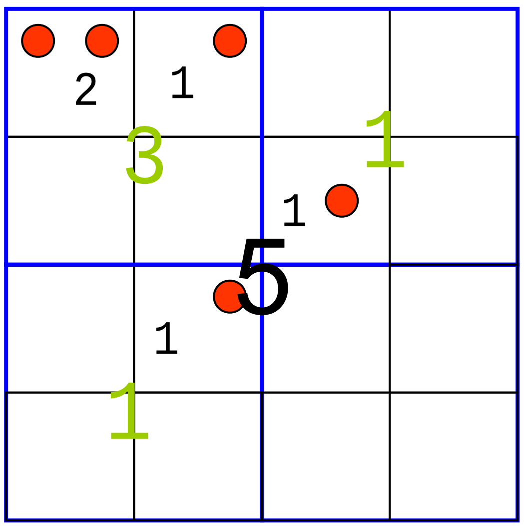
- Approach:
  - Embed EMD into l_1^d (with distortion c)
  - Use c'-approximate NN for l_1^d
  - This gives cc' -approximate NN for EMD
- Used earlier in
  - [FarachColton-Indyk'99]: Hausdorff metric over l_p^d into low-dimensional l_∞
  - [Cormode-Paterson-Sahinalp-Vishkin'00, Muthukrishnan-Sahinalp'00, Cormode-Muthukrishnan'02]: Block-edit distance into l_1



EMD into l_1

- Assume $P \subseteq \{1,\ldots,\Delta\}^d$
- Impose square grids $G_{-1} \ldots G_k$, with side lengths $2^{-1}, 2^0, \ldots, 2^k = \Delta$, shifted at random.
- For each square cell c in G_i, let $n^i_P(c)$ be the number of points in $|c \cap P|$.
- Embedding: P is mapped to

$$f(P) = 2^{-1}n^{-1}_P, 2^0 n^0_P \ldots 2^k n^k_P$$

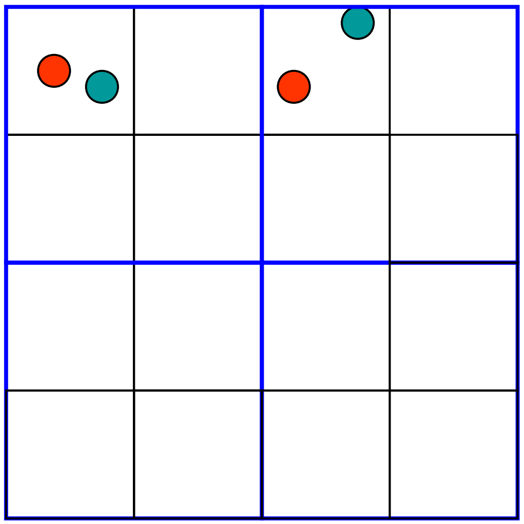


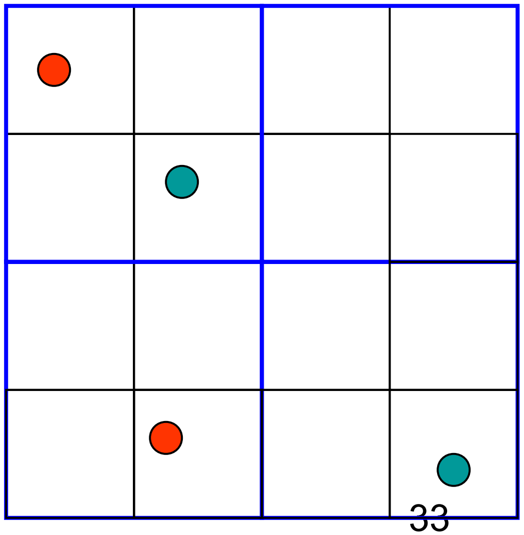
Guarantees

- Theorem:
  - $EMD(P,Q) < O(||f(P)-f(Q)||_1)$
  - $E[||f(P)-f(Q)||_1] = O(\log \Delta) EMD(P,Q)$
- Due to:
  - Charikar'02, Kleinberg-Tardos'99, Bartal'96, Peleg'97+Goel [personal communication]
  - Indyk-Thaper'02, Varadarajan'02

Proof intuition

- EMD(P,Q) small:
  - Most points in P are close to the corresponding points in Q
  - Corresponding points fall to the same cell
  - Counts cancel out: $||f(P)-f(Q)||_1$ small
- EMD(P,Q) large:
  - Many points in P are far from the points in Q
  - Corresponding points fall to different cells
  - Counts do not cancel out

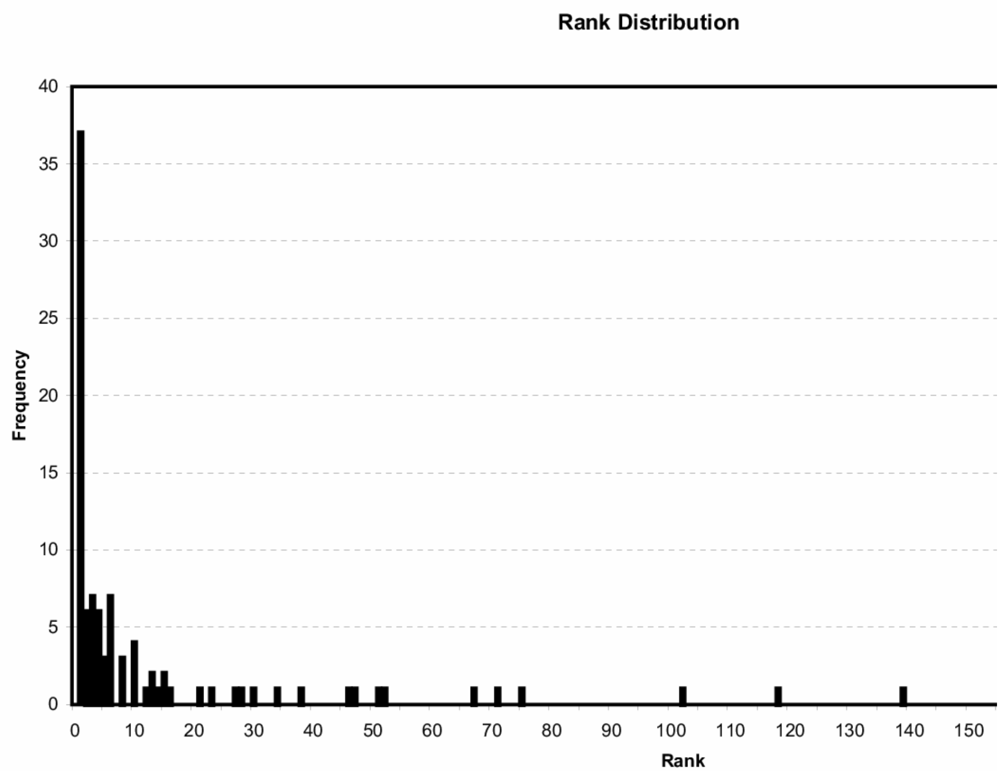




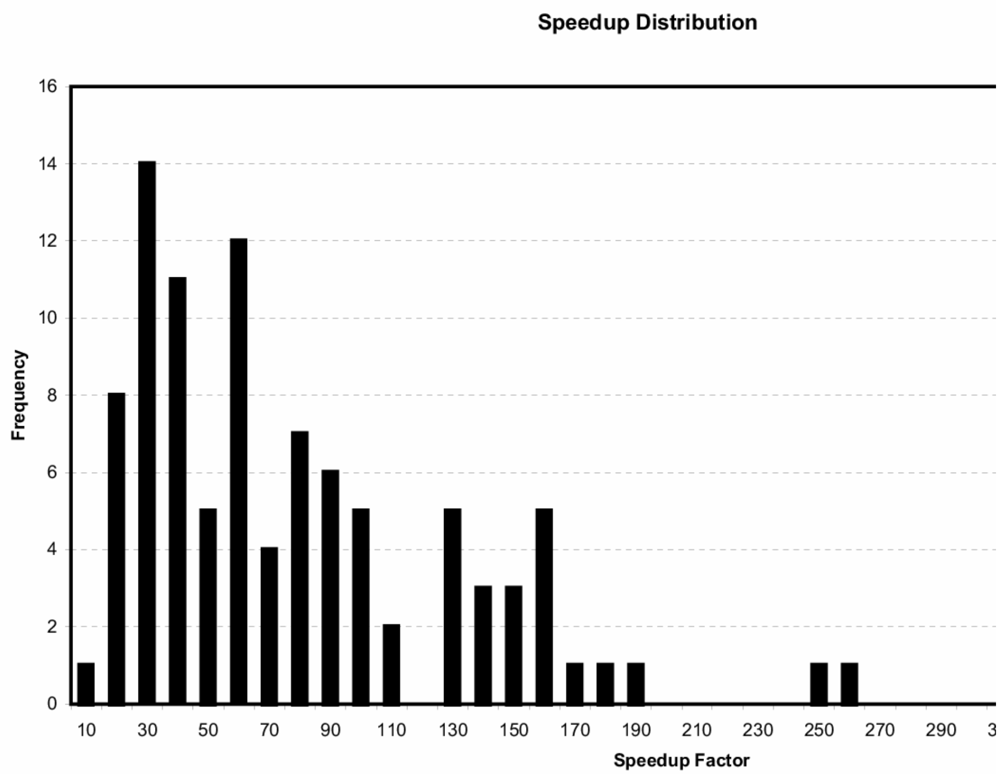
How Does This Work in Practice?

- Data: color histograms of 20,000 Corel-Draw images:
  - Each pixel in an image is a point in 3D color space
  - Image represented by a bag of pixels
- 100 queries
- Parameters:
  - Probability of failure set to 10%
  - Embedding done 5 times per query
  - Approximation factor c set by hand
- Compare our approximate NN to the exact NN (w.r.t. EMD)

NN Quality: Rank

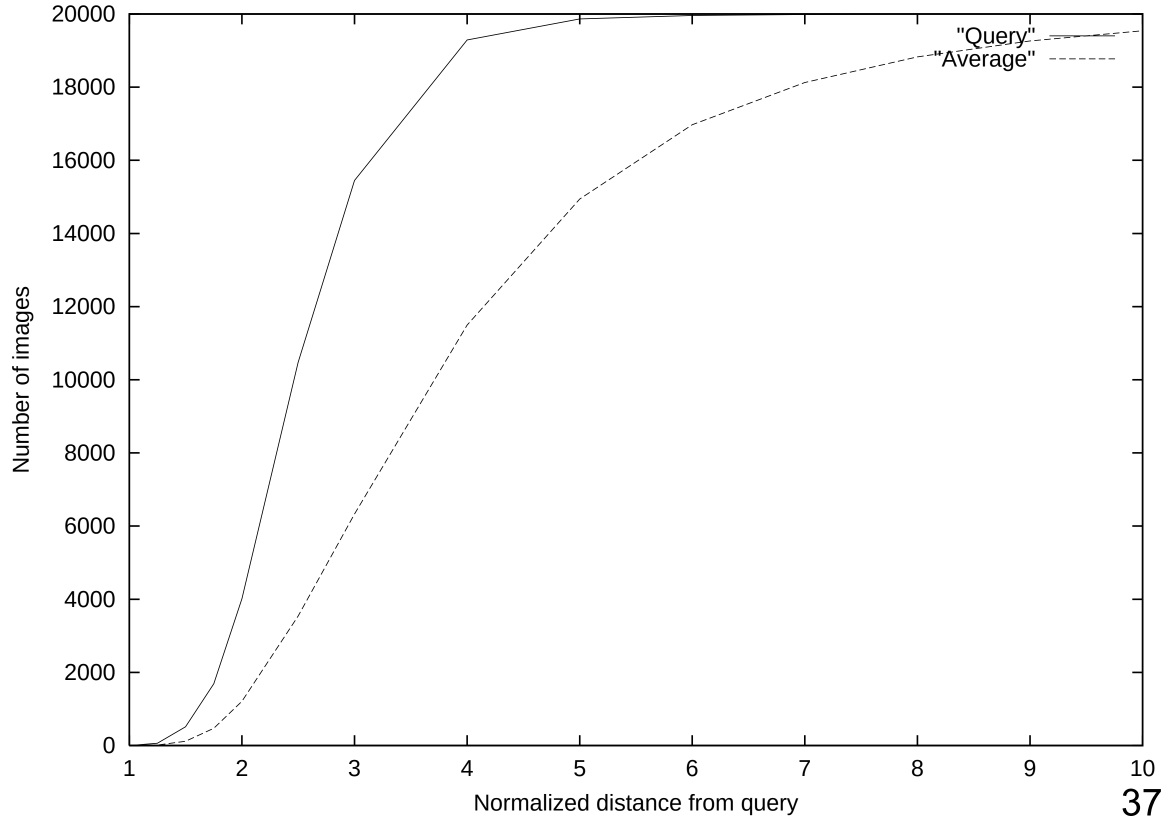


Speedup Over Linear Scan



Data profile

- Shows the number of c-approximate nearest neighbors as a function of c:
  - Bad case
  - Typical case



Conclusions for NN under EMD

- Efficient algorithm for NN under EMD via:
  - Embedding EMD into l_1^d
  - Fast NN in l_1^d
- $O(\log \Delta)$ pretty good in practice