Principles of Information Filtering in Metric Spaces

Paolo Ciaccia and Marco Patella
DEIS, Università di Bologna – Italy

SISAP 2009 – August 29-30 2009, Prague

Information Filtering

- The IF problem:
  - Deliver to users only the information that is relevant to them, filtering out all irrelevant new data items
  - News, papers, ads, CfP, …
- Compared to IR:

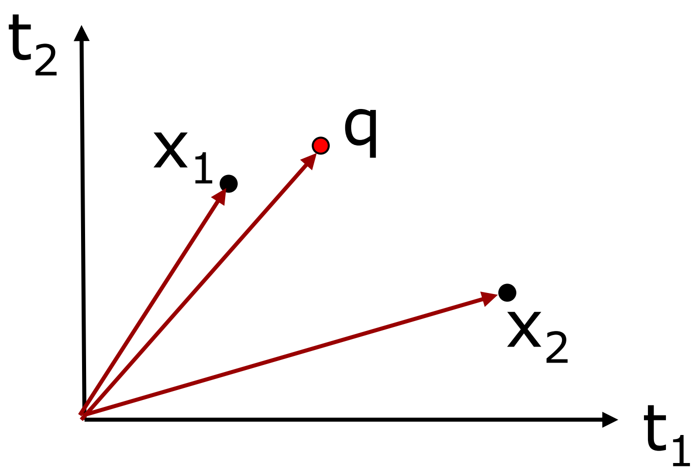
	IR	**IF**
Goal	Selecting relevant items for each query	Filtering out the many irrelevant data items
Type of use; Type of users	Ad-hoc use; one-time users	Repetitive use; **long-term users**
Representation of information needs	Queries	**User profiles**
Index	Items	**User profiles**

User Profiles

- Common (text-based) VSM approach:
  - Profile = vector in some appropriate space (terms, topics,…)
  - Built using e.g., TF-IDF text analysis

$$\boxed{x_i = ((t_{i,1}, w_{i,1}), \ldots, (t_{i,n}, w_{i,n}))}$$

  - Matching profiles with a new data item q: Cosine similarity

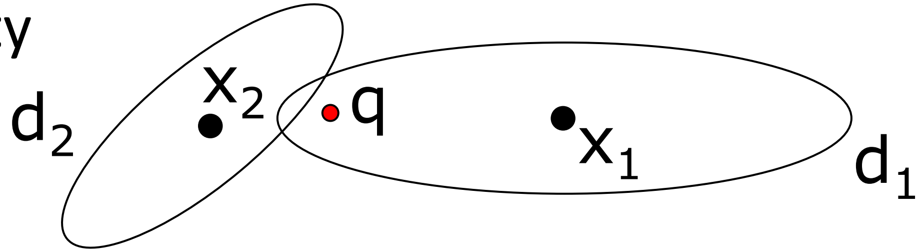


Limitations

- Suitable only for text
  - No analogous of content-based MM search
- VSM profiles capture only the "position" of users
- They do not model the (subjective) notion of similarity

OBJECTIVE:

- Extend the IF model to metric spaces (MIF), thus allowing also distance to depend on user preferences
  - This widens IF applicability



Preferences change the distance

- My preferences:
  - Highways
- Marco's preferences (driving his bike):
  - Scenery roads
- According to ViaMichelin:

$$\boxed{\begin{aligned} d_{Paolo}(\text{Bologna},\text{Prague}) &= 948 \text{ km} \\ d_{Marco}(\text{Bologna},\text{Prague}) &= 873 \text{ km} \end{aligned}}$$

- Other examples: RF for MM information retrieval

The Metric Information Filtering problem

Given a set X of user profiles $u_i = (x_i , d_i)$, where
x_i is the profile centroid and d_i is the user-specific distance, and a new data item q

Determine the profiles for which q is relevant

- Relevance of q to user u_i measured as $d_i(x_i,q)$
- Wlog we set a threshold/radius r_i to discriminate among relevant and irrelevant items

$$d_i(x_i,q) \leq r_i \Rightarrow q \text{ is relevant to } u_i$$

Metric Search vs Metric Filtering

- Both can use a user-specified distance d_i, but:

  Metric search: one d_i at a time

  MIF: N users = N distances at the same time!

- Lesson learned from metric search [Ciaccia, Patella; TODS 2002]:

If objects are indexed by a metric index using a distance δ and $\exists$ a finite $s_{\delta,d}$ s.t. $\delta(x,q) \leq s_{\delta,d}\, d(x,q)$ holds $\forall x,q$

Then the index can also process queries based on d

- The minimum of such $s_{\delta,d}$ is called the (optimal) scaling factor of d wrt δ

Examples of scaling factors

- Weighted Lp norms: $d_i(a,b) = \left(\sum_k w_i[k] \left| a[k] - b[k] \right|^p\right)^{1/p}$

$$\boxed{d_i(a,b) \le \max_k\{(w_i[k]/w_j[k])^{1/p}\}\ d_j(a,b)}$$

- Sum of metrics:

Weights	Marco	Paolo
Km	1	2
Time	2	5
Cost	3	1

$$\begin{aligned} d_i(a,b) = \ & w_i[km]d[km](a,b) + \\ & w_i[time]d[time](a,b) + \\ & w_i[cost]d[cost](a,b) \end{aligned}$$

$$d_{Marco}(a,b) \le 3/1\ d_{Paolo}(a,b)$$
$$d_{Paolo}(a,b) \le 5/2\ d_{Marco}(a,b)$$

Pivot-based methods for MIF

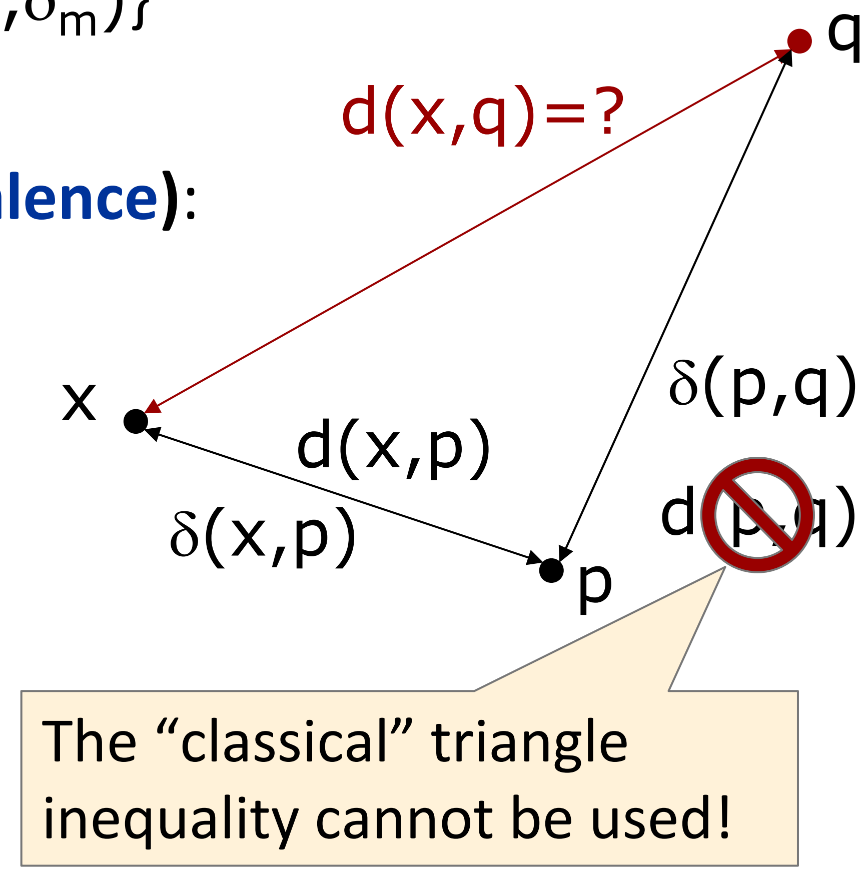
- Profiles $X = \{(x_1,d_1),\ldots,(x_n,d_n)\}$
- Pivots $P = \{(p_1,\delta_1),\ldots,(p_m,\delta_m)\}$

Assumption (Lipschitz equivalence):

$\forall d,\delta\ \exists\ s_{d,\delta}$ and $s_{\delta,d}$:

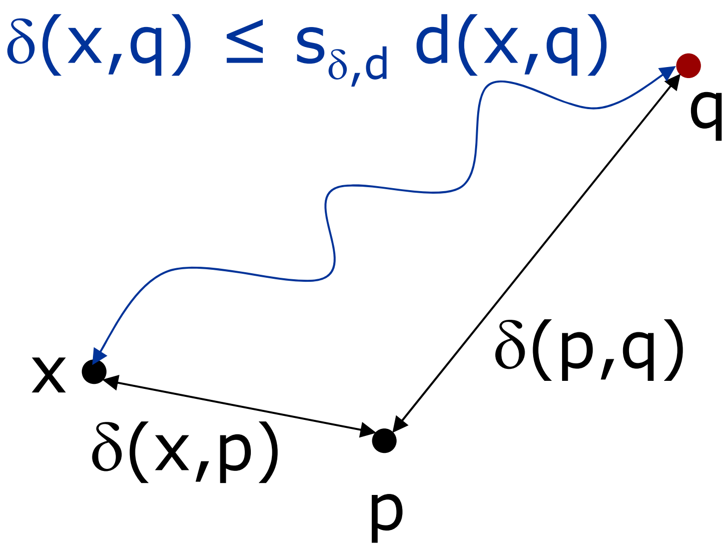
$$d(a,b) \le s_{d,\delta}\,\delta(a,b)$$
$$\delta(a,b) \le s_{\delta,d}\,d(a,b)$$

Goal: to provide a (tight) lower bound to $d(x,q)$



Pivot-space

- The index stores $\delta(x,p)$



$$d(x,q) \ge \delta(x,q)/s_{\delta,d}$$
$$\ge [\delta(p,q)-\delta(x,p)]/s_{\delta,d} \quad (7)$$

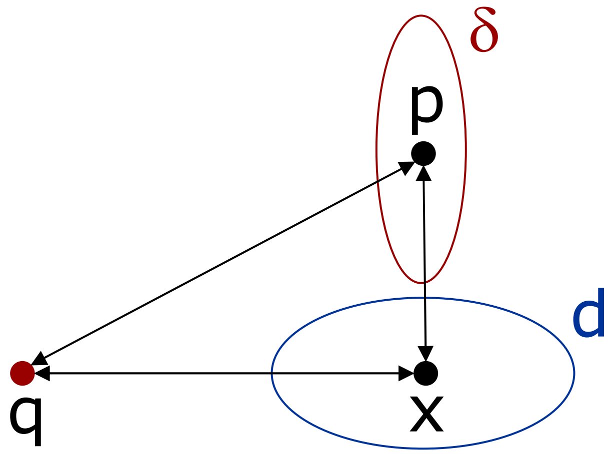
$$d(x,q) \ge [\delta(x,p)-\delta(p,q)]/s_{\delta,d} \quad (9)$$

- By using both scaling factors two other LB's can be obtained, but they are always looser

Approximation can help

- Consider (7): $d(x,q) \geq [\delta(p,q)-\delta(x,p)]/s_{\delta,d}$

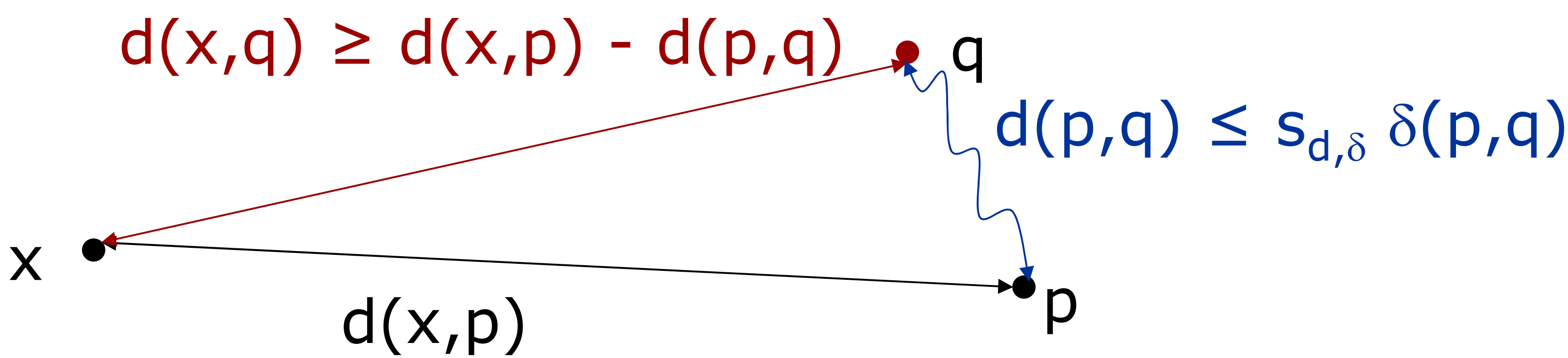
  and the classical inequality: $d(x,q) \geq d(p,q)-d(x,p)$
- It can well be $[\delta(p,q)-\delta(x,p)]/s_{\delta,d} \geq d(p,q)-d(x,p)$, thus working in pivot-space can be even better!



$d(p,q)$	high
$d(x,p)$	medium
$\delta(p,q)/s_{\delta,d}$	medium
$\delta(x,p)/s_{\delta,d}$	very low

Point / profile-space (1)

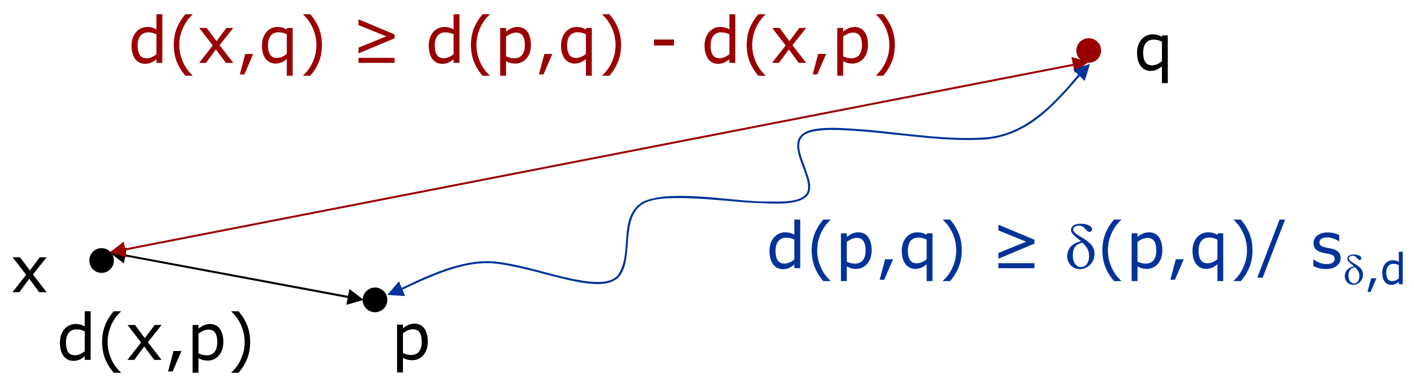
- The index stores d(x,p)
- "Large" pivot-point distance



$$d(x,q) \geq d(x,p) - s_{d,\delta}\, \delta(p,q) \quad (10)$$

Point-space (2)

- “Small” pivot-point distance



$$d(x,q) \geq \delta(p,q)/ s_{\delta,d} - d(x,p) \quad (11)$$

- (11) is always dominated by (7):

$$\delta(p,q)/s_{\delta,d} - \delta(x,p)/s_{\delta,d} \geq \delta(p,q)/ s_{\delta,d} - d(x,p)$$

Symmetric Scaling Factors

- Define the Symmetric Scaling Factor of d and δ as:

$$SSF(d,\delta) = s_{d,\delta} * s_{\delta,d}$$

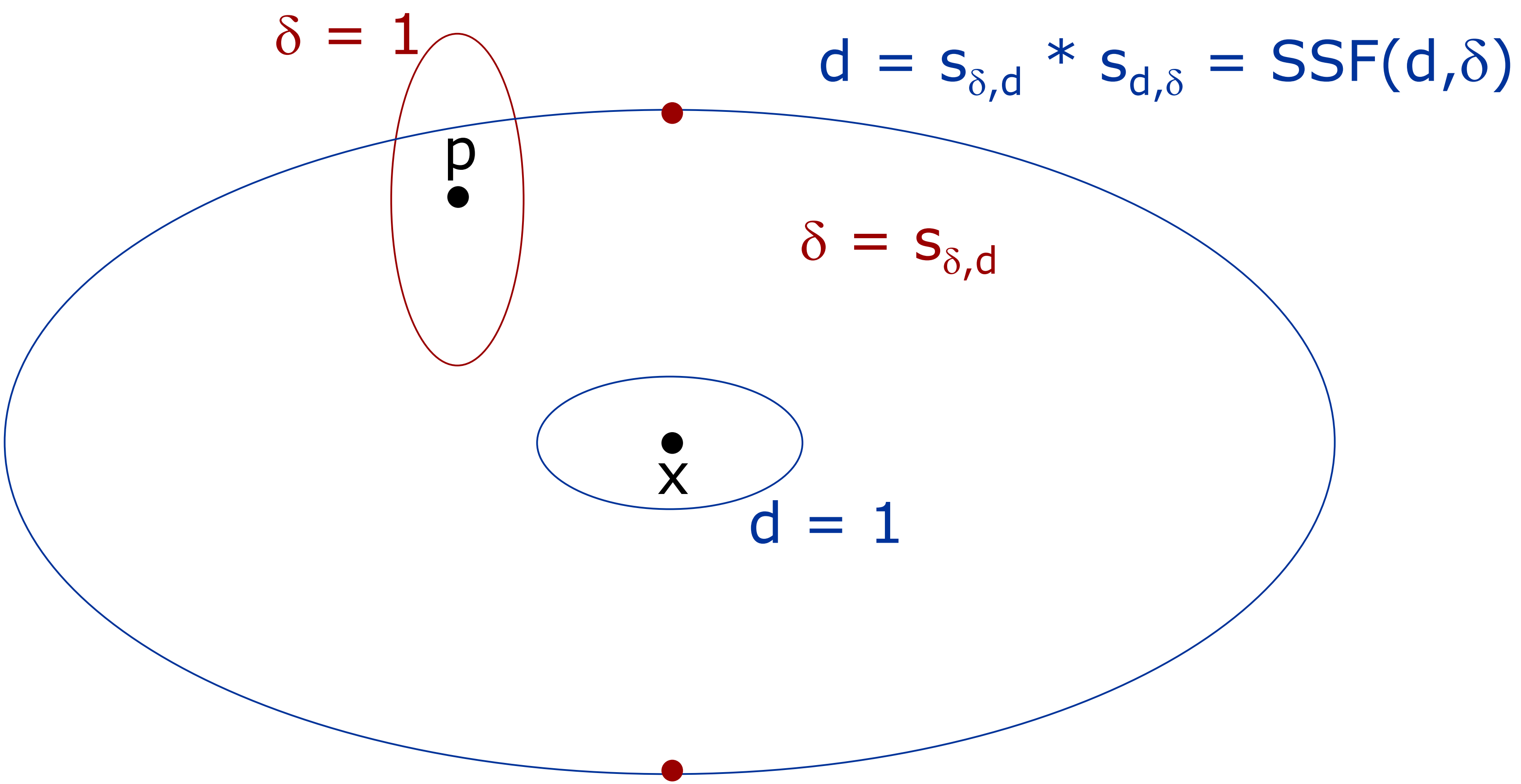
SSF Properties

- $SSF(d,\delta) = SSF(\delta,d)$
- $SSF(d,\delta) \geq 1$ ($= 1$ iff d is a scaled version of δ)
- $SSF(d,\delta) \leq SSF(d,d') * SSF(d',\delta) \quad \forall d'$

log SSF is a pseudo-metric on every space of Lipschitz-equivalent metrics

- SSF can be used to measure how well δ approximates d
  - Also known as the "distortion" of the two metrics

Q: What does SSF measure?



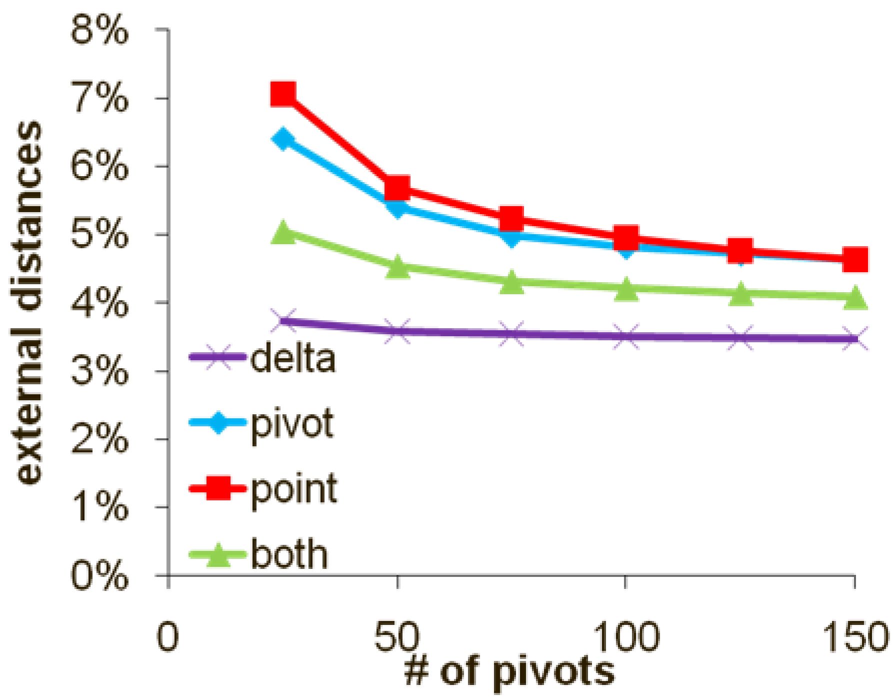
A: How much, in the worst-case (red points), we relax d by approximating it with δ (and vice versa)

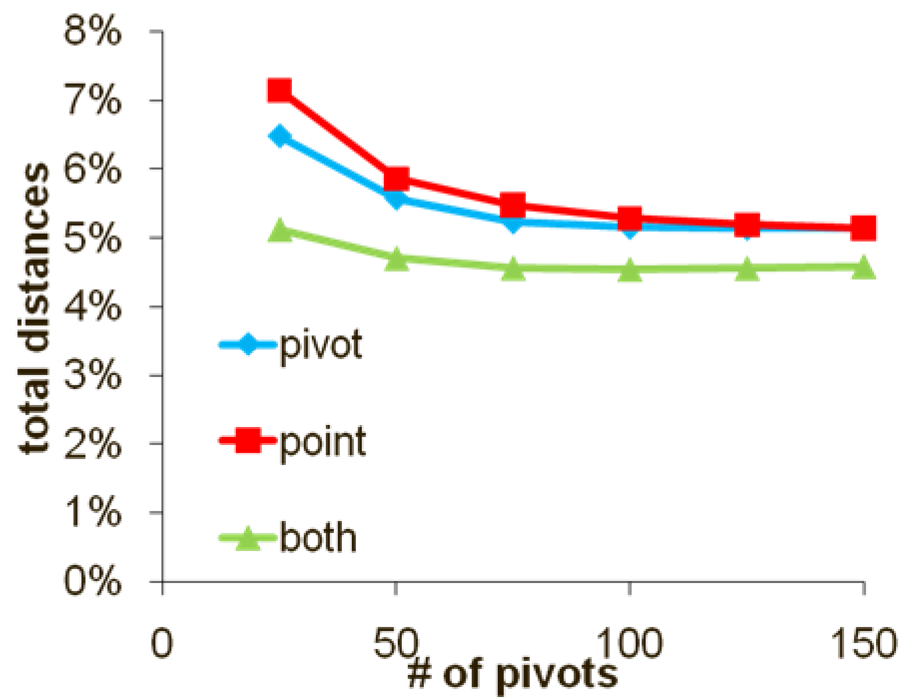
Experimental settings

- 3D synthetic datasets w weighted Euclidean distance:
  - uniform
  - clustered (5 Gaussian clusters)
  - **random walk** (points/weights obtained by slightly perturbing the previous point/weight)
  - radii = about 3% of data items are relevant for each profile
- Strategies:
  - Δ (classical triangle inequality – only for reference purpose)
  - Δ-pivot (pivot-space: (7)+(9))
  - Δ-point (point-space: (10)+(11))
  - Δ-both (pivot- **and** point-space: (7)+(9)+(10))

Experiment I: the best strategy

30K data points

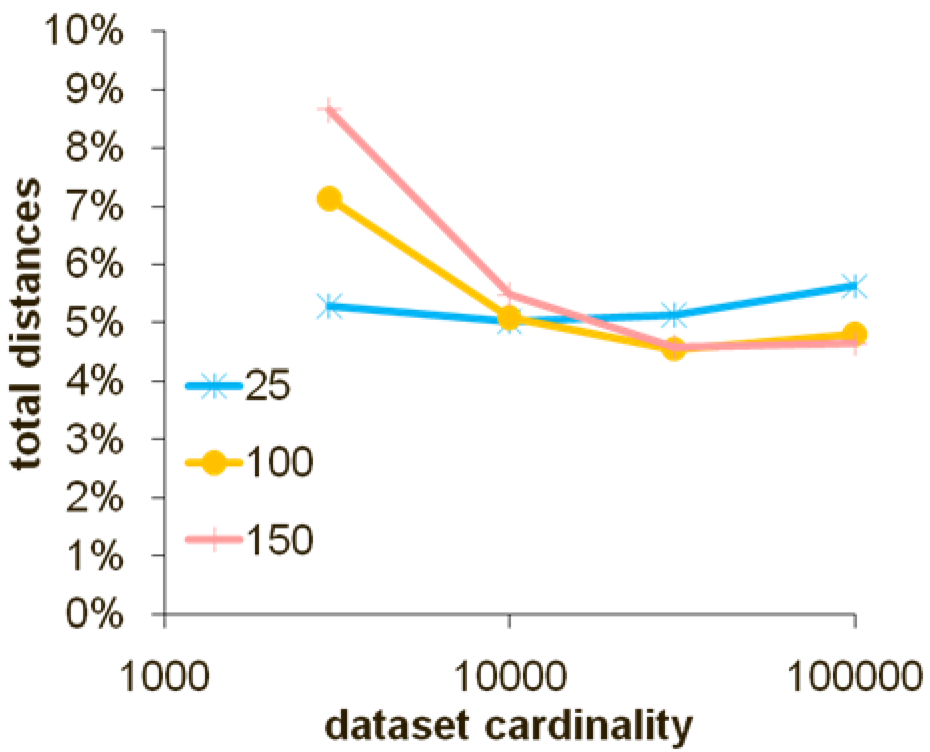




- external distances: distances between q and profiles
- total distances: external distances + distances between q and pivots

Experiment II: optimal # of pivots

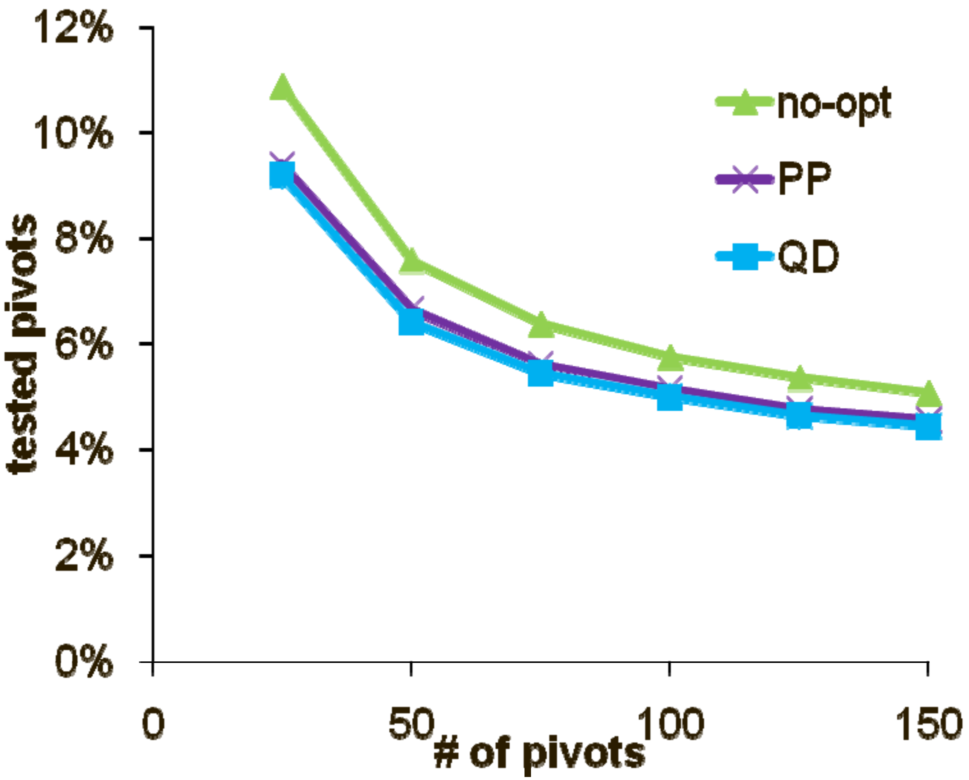
Δ-both strategy



Experiment III: sorting pivots

- Pivots are sorted so as to minimize the number of comparisons
- Strategies:
  - QD: increasing distance to q
  - PP: decreasing pruning power (computed using the distance distribution of each pivot)

Δ-both strategy, 30K points



Conclusions and open issues

- Introduced basic principles of Metric Information Filtering
  - Suitable for any family of Lipschitz-equivalent metrics
  - Not limited to pivot-based methods
  - Space-time tradeoff on what to index (pivot- vs point-space)

- Is MIF also suitable for collaborative filtering?
  - Relevance of a new item now depends on profiles' similarity
- Can MIF exploit batch arrivals of new items?
  - Need some "default" metric to compare items
- Can SSF be used for choosing pivots?
- What if a pivot does not use its own metric?
  - Can we decouple pivot position from pivot preferences?

Thanks for your attention !