1. Feature Hashing [Weinberger et al. ’09]

Setting: Define $A \in \mathbb{R}^{m \times n}$ to be a random matrix s.t. every column has one randomly chosen non-zero entry, which is randomly set to 1 or $-1$.

Goal: For every $x \in \mathbb{R}^n$, $Ax$ approximates $|x|$.

More precisely, given $\epsilon \in (0,1)$ we want that for every $x \in \mathbb{R}^n$.

$||Ax||_2 \leq \epsilon \cdot |x|$, \quad \forall x \in \mathbb{R}^n\setminus \{0\}$

Observation 1 [Weinberger et al. ’09]: $\frac{\text{Pr}[|Ax| \leq (1 \pm \epsilon)|x|]}{|x|} \geq 1 - \delta.$

2. Motivation

Example Problem: How do we find the $k$ closest movies?

Goal: Reduce Storage and Running Time. Maintain Approximate Distance

2.1. Recommendation Systems

<table>
<thead>
<tr>
<th>1 0 1 1 0 0 1 1 0 . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1 0 1 0 1 0 1 0 1 . . . )</td>
</tr>
<tr>
<td>Storing a corpus of $M$ movies requires $\Omega(M)$ memory</td>
</tr>
<tr>
<td>Comparing two vectors takes $\Omega(m)$ operations</td>
</tr>
</tbody>
</table>

Edge labeled graph $\tilde{G}$ over $[k]$ with $r$ edges.

3. Main Results

3.1. Formalizing Observation 1

Define $v(m, \epsilon, \delta)$ as the max $v$ such that whenever $||Ax||_2 \leq \epsilon |x|$, then (1) holds.

Theorem 1: $v(m, \epsilon, \delta) \leq \left(\frac{\log m}{\log 1/\delta}\right)^{2\epsilon \cdot \log 1/\delta}.$

4. Proof Technique

4.1. Main Technical Result

For $r = 1, 1, \ldots, 1, 0, \ldots, 0$, theorem holds a $2\epsilon \cdot \log 1/\delta$ factor.

4.2. Independent

5. Counting Eulerian Graphs

Notation 1: Let $\mathcal{G}_{\alpha, \beta}$ denote the set of edge-labeled Eulerian multigraphs $G$ with $r$ edges over vertex set $[\alpha]$ with $\beta$ non-degenerate connected components and $\alpha - \beta$ isolated vertices.

Theorem 2: Given bounds on the size of $\mathcal{G}_{\alpha, \beta}$, in terms of $\alpha$ and $\beta$.

Formally, $|\mathcal{G}_{\alpha, \beta}| = \frac{\alpha!}{\beta!} \cdot (\alpha - \beta)!^2 \cdot (\alpha - 2\beta)!^2 \cdot (\alpha - 4\beta)!^2 \cdots$.

6. Proof of Upper Bound

6.1. Observation 1: Encode the list of non-isolated nodes.

6.2. Observation 2: Encode a tree in every connected component.

6.3. Observation 3: Encode one extra edge in every connected component.

6.4. Observation 4: Encode the rest of the edges.

6.5. Total of $\Theta(r) \cdot 2^{\Omega(r)}$ bits.

7. Proof of Lower Bound

7.1. Observation 3: Select the $r$ non-isolated nodes.

7.2. Observation 4: Select $\beta \leq r$ unordered pairs.

7.3. Observation 5: Select a cycle in the large connected component.

7.4. Observation 6: Select the rest of the $r - \beta$ edges.

For $r = 1, 1, \ldots, 1, 0, \ldots, 0$, $\frac{\left(\frac{\log m}{\log 1/\delta}\right)^{2\epsilon \cdot \log 1/\delta}}{\frac{\alpha!}{\beta!} \cdot (\alpha - \beta)!^2 \cdot (\alpha - 4\beta)!^2 \cdots}$.

8. Synthetic Data

For $m$ and $r$ values exponentially placed in $[2^1, 2^2]$, and $[2^5, 2^6]$, we tested $2^{r+1}$ generated bit vectors with $v = \frac{\epsilon}{\log 1/\delta}$.