Mimicking Data By Learning Patterns on Data Constraints

By
Kapil Khurana & Vishal Goel

Motivation

- Database vendors need to test their engine on real world databases.

Motivation

- Database vendors need to test their engine on real world databases.
- But database clients cannot send their data because of privacy issues and huge transfer cost.

Motivation

- Database vendors need to test their engine on real world databases.
- But database clients cannot send their data because of privacy issues and huge transfer cost.
- Thus, database vendors have to create their own synthetic database that resembles the client's database, qualitatively and quantitatively.

Motivation

- Database vendors need to test their engine on real world databases.
- But database clients cannot send their data because of privacy issues and huge transfer cost.
- Thus, database vendors have to create their own synthetic database that resembles the client's database, qualitatively and quantitatively.

- But how?

CLIENT SIDE

VENDOR SIDE

CLIENT SIDE

Table Employee T

Age	Rating	Salary
25	5.0	25,000
33	8.0	40,000
51	9.0	70,000

VENDOR SIDE

CLIENT SIDE

Table Employee T

Age	Rating	Salary
25	5.0	25,000
33	8.0	40,000
51	9.0	70,000

Run a set of queries like
*Select * from T where Age>25 and Rating<8.5*

VENDOR SIDE

CLIENT SIDE

Table Employee T

Age	Rating	Salary
25	5.0	25,000
33	8.0	40,000
51	9.0	70,000

→ Run a set of queries like

*Select * from T where Age>25 and Rating<8.5*

→ Training data :

A set of pairs (q_i, c_i)

$c_i \in R$

VENDOR SIDE

CLIENT SIDE

Table Employee T

Age	Rating	Salary
25	5.0	25,000
33	8.0	40,000
51	9.0	70,000

Run a set of queries like
*Select * from T where Age>25 and Rating<8.5*

Training data :
A set of pairs (q_i, c_i)
$c_i \in R$

Anonymize

Build a cardinality estimation model CEM
that learns distribution of T

Goal : Given a query q , return cardinality.

VENDOR SIDE

CLIENT SIDE

Table Employee T

Age	Rating	Salary
25	5.0	25,000
33	8.0	40,000
51	9.0	70,000

Run a set of queries like
*Select * from T where Age>25 and Rating<8.5*

Training data :
A set of pairs (q_i, c_i)
$c_i \in R$

Anonymize

Build a cardinality estimation model CEM
that learns distribution of T

Use CEM to generate T'

Goal : Given a query q , return cardinality.

VENDOR SIDE

CLIENT SIDE

Table T

Age	Rating	Salary
25	5.0	25,000
33	8.0	40,000
51	9.0	70,000

Run a set of queries like
*Select * from T where Age>25 and Rating<8.5*

Training data :
A set of pairs (q_i,c_i)

Anonymize

Build a cardinality estimation model CEM
that learns distribution of T

Goal : Given a query q , return cardinality.

Use CEM to generate T'

Table T'

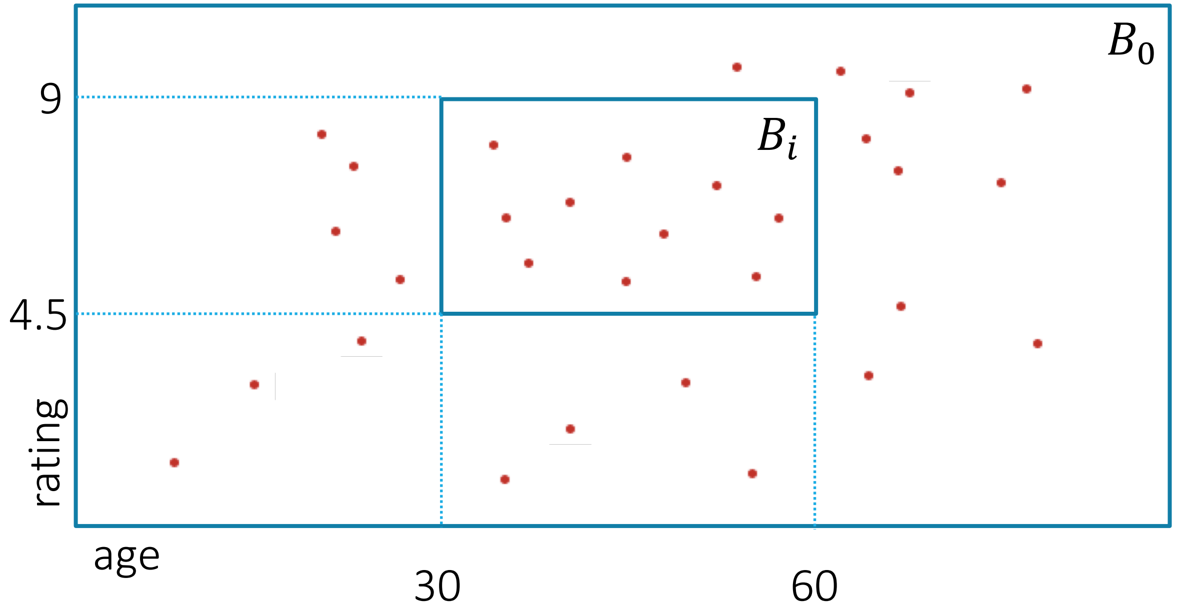
Age	Rating	Salary
23	4.5	23,000
30	7.8	44,000
55	8.7	67,000

$q_i(T') = c_i$
$q'(T') \approx q'(T)$

VENDOR SIDE

Notations

- q_i : Select * from T where 30 ≤ age ≤ 60 and 4.5 ≤ rating ≤ 9.
- P_i : 30 ≤ age ≤ 60 and 4.5 ≤ rating ≤ 9
- c_i = 10/|T|



Problem Statement

- Consider a set of n observed queries $(P_1, c_1), \dots, (P_n c_n)$ for T and let $f(x)$ denote pdf of T.

Problem Statement

- Consider a set of n observed queries $(P_1, c_1), \dots, (P_n c_n)$ for T and let $f(x)$ denote pdf of T.
- By definition, we have the following for each $i = 1, \dots, n$

$$\int_{x \in B_i} f(x) = c_i$$

Problem Statement

- Consider a set of n observed queries $(P_1, c_1), \ldots, (P_n c_n)$ for T and let $f(x)$ denote pdf of T.
- By definition, we have the following for each $i = 1, \ldots, n$

$$\int_{x \in B_i} f(x) = c_i$$

- GOAL : To build CEM of $f(x)$ that satisfies the above n constraints and can estimate the cardinality c' of a new predicate P'.

Problem Statement

- Consider a set of n observed queries $(P_1, c_1), \dots, (P_n c_n)$ for T and let $f(x)$ denote pdf of T.
- By definition, we have the following for each $i = 1, \dots, n$

$$\int_{x \in B_i} f(x) = c_i$$

- GOAL : To build CEM of $f(x)$ that satisfies the above n constraints and can estimate the cardinality c' of a new predicate P'.

- Next Step : To generate a synthetic Table T' using CEM .

Approach

- **Uniform Mixture Model** : Represent the population distribution $f(x)$ as a weighted sum of multiple uniform distributions, $g_z(x)$ for $\mathrm{z} = 1, \ldots, m$. Specifically,

$$f(x) = \sum_{z=1}^{m} w_z \, g_z(x)$$

Reference : Yongjoo Park, Shucheng Zhong, Barzan Mozafari "QuickSel: Quick Selectivity Learning with Mixture Models", SIGMOD 2018

Approach

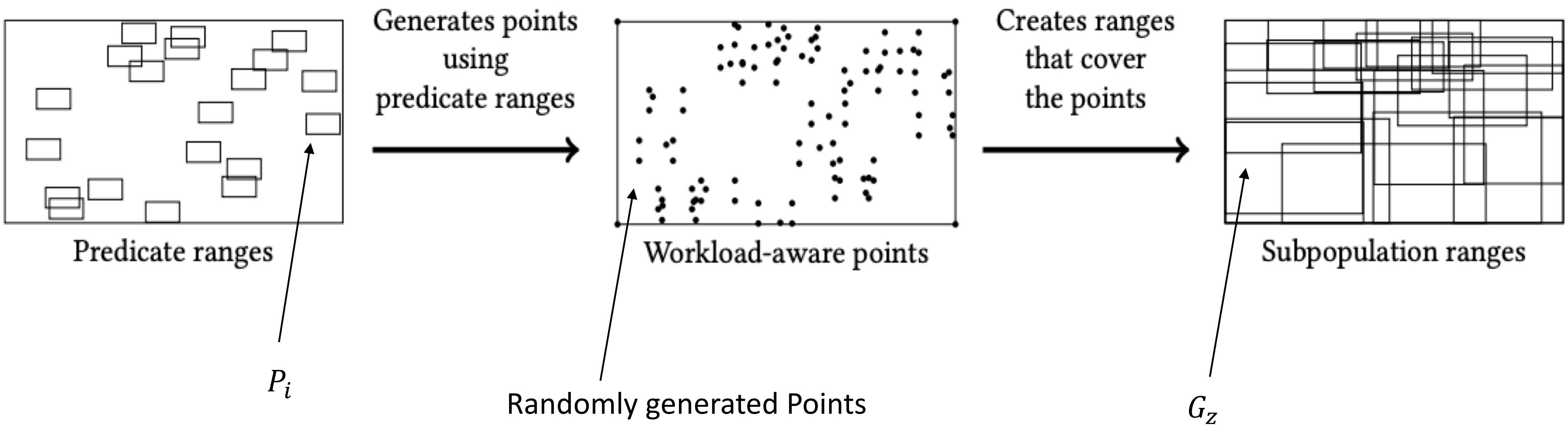
- **Uniform Mixture Model** : Represent the population distribution $f(x)$ as a weighted sum of multiple uniform distributions, $g_z(x)$ for $\mathrm{z} = 1, \ldots, m$. Specifically,

$$f(x) = \sum_{z=1}^{m} w_z \, g_z(x)$$

- $g_z(x)$ is the pdf (which is a uniform distribution) for the z^{th} subpopulation
- The support for $g_z(x)$ is represented by a hyper-rectangle G_z

Reference : Yongjoo Park, Shucheng Zhong, Barzan Mozafari "QuickSel: Quick Selectivity Learning with Mixture Models", SIGMOD 2018

Approach



HYPER-PARAMETERS : p, m, k

Approach

- The optimal parameter w for the model is obtained by solving

$$\underset{w}{\operatorname{argmin}} \quad \int_{x \in B_0} (f(x) - \frac{1}{|B_0|})^2 dx$$

$$such \quad that \int_{B_i} f(x)dx = c_i, \quad \forall i = 1, .., n$$

$$f(x) \geq 0$$

Approach

- The optimal parameter w for the model is obtained by solving

$$\underset{w}{\operatorname{argmin}} \quad \int_{x \in B_0} (f(x) - \frac{1}{|B_0|})^2 dx$$

$$such \quad that \int_{B_i} f(x)dx = c_i, \quad \forall i = 1, .., n$$

$$f(x) \geq 0$$

- The approximate solution of the above problem is given by:

$$\mathbf{w*} = (Q + \lambda A^T A)^{-1} \lambda A c \quad where$$

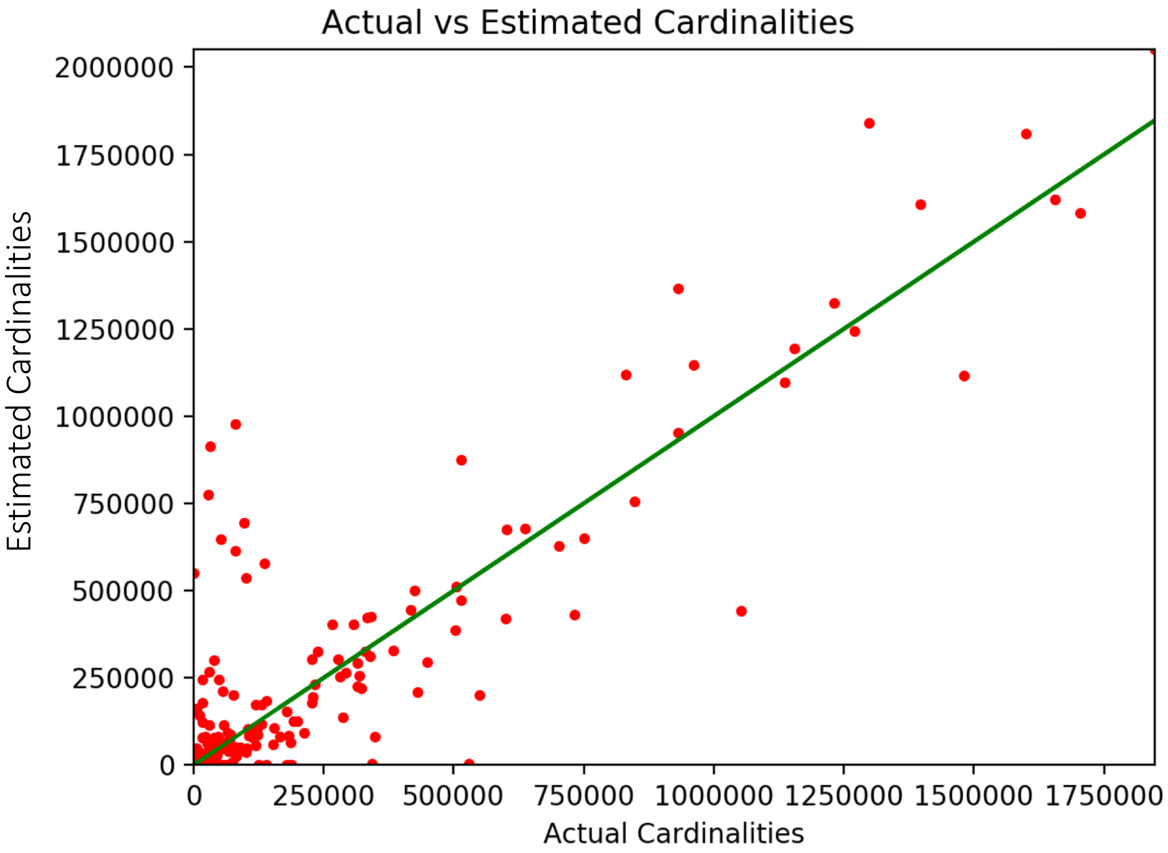
$$(Q)_{ij} = \frac{|G_i \cap G_j|}{|G_i||G_j|} \quad (A)_{ij} = \frac{|B_i \cap G_j|}{|G_j|}$$

Experiments

DATASET : Instacart [sale records of an online grocery store]

- TABLE orders(…., order_hour_of_the_day, days_since_prior)
- #rows = 3.2 million
- Attributes with ranges (0,23) and (0,31)

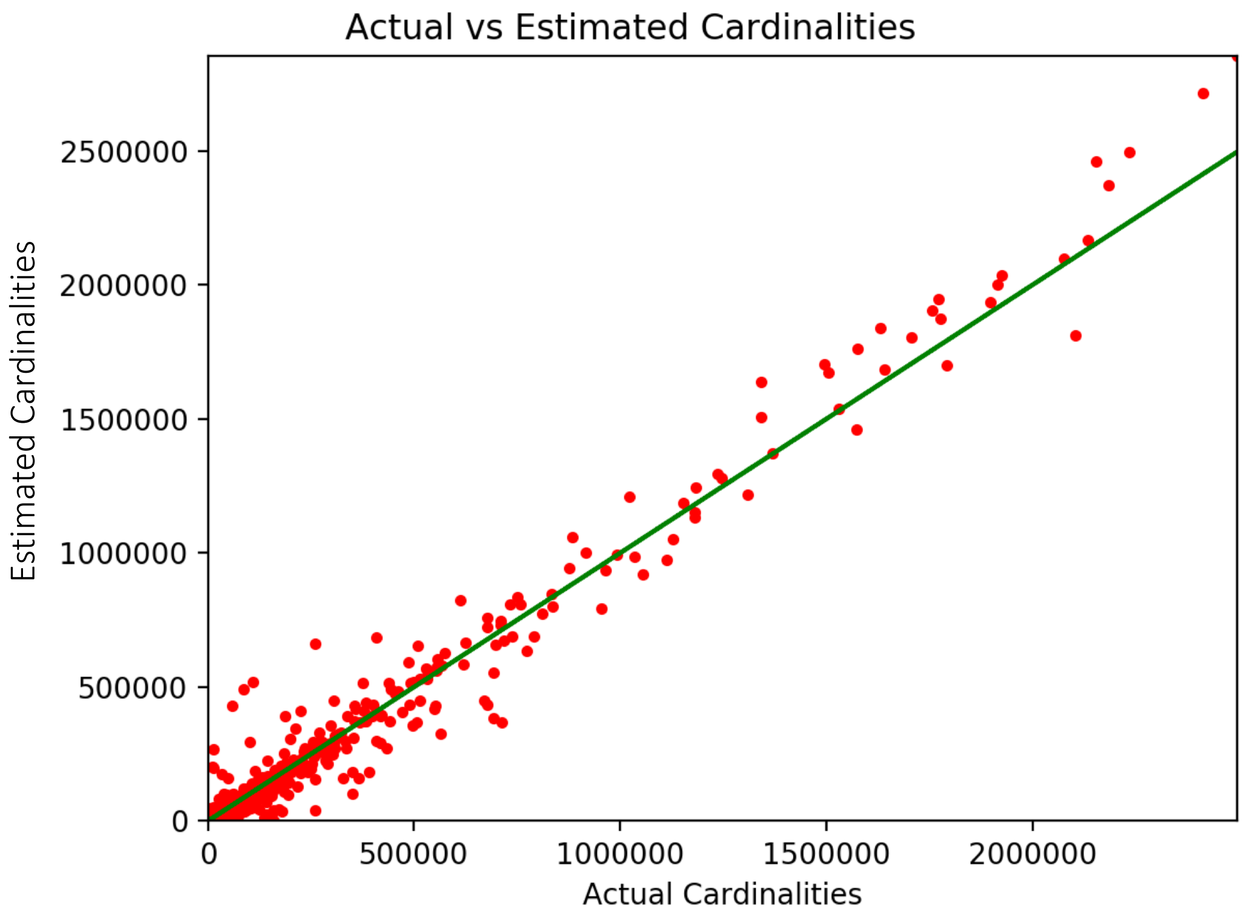
Experiments



MIXTURE MODEL
p=10, m=2000, k=30

Training set : 1k
Test set : 200
Relative error : 34%

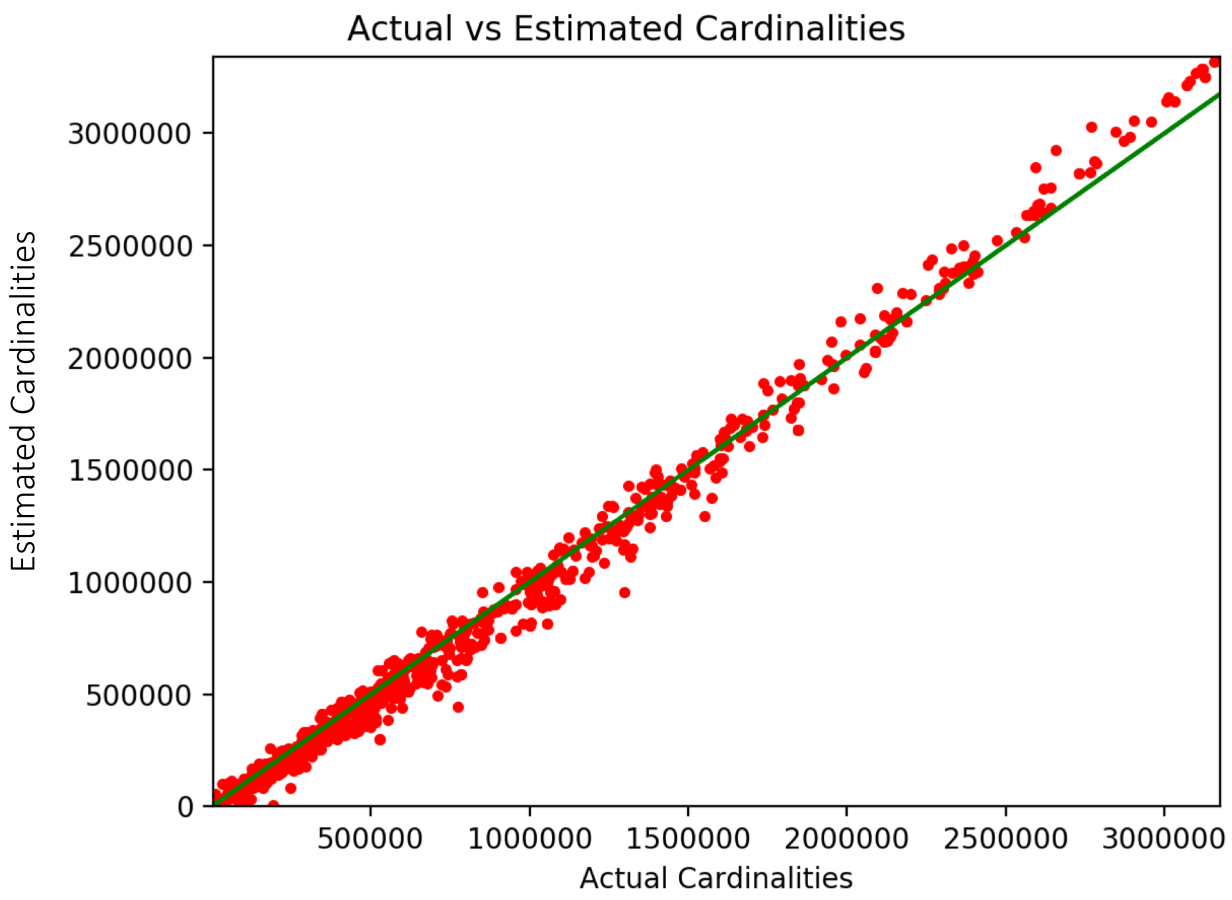
Experiments



MIXTURE MODEL
p=10, m=2000, k=30

Training set : 1.5k
Test set : 300
Relative error : 24%

Experiments



MIXTURE MODEL
p=10, m=2000, k=30

Training set : 1k + 0.5k 1d
Test set : 500
Relative error : 10.9%

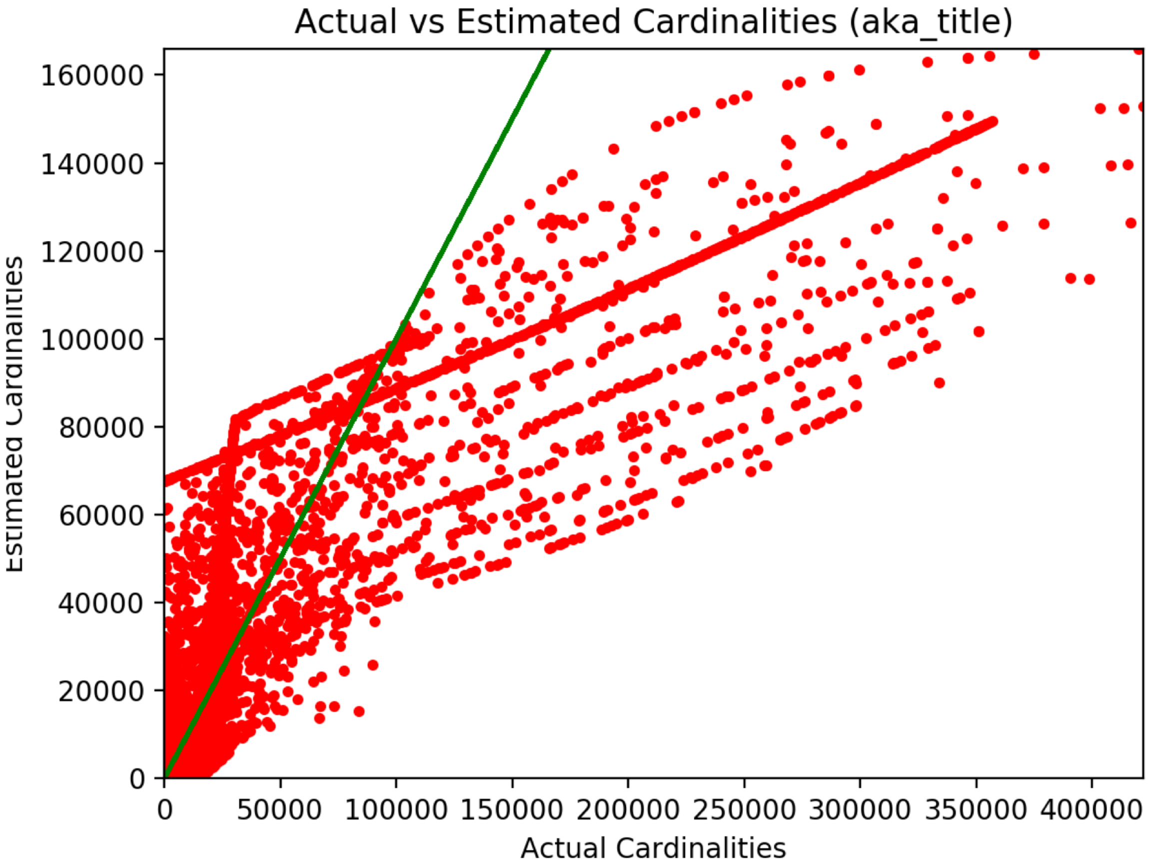
Relative error on Training set = 4%

Experiments

DATASET : IMDB (movie records)

- Table : aka_title (id, kind_id, movie_id, production_year)
- #rows = 4.3 million
- 4 attributes with ranges (1, 4.3 million), (1, 7), (0, 3.4 million) and (1875, 2022)

Experiments



NEURAL NETWORK

1 hidden layer with 10 nodes

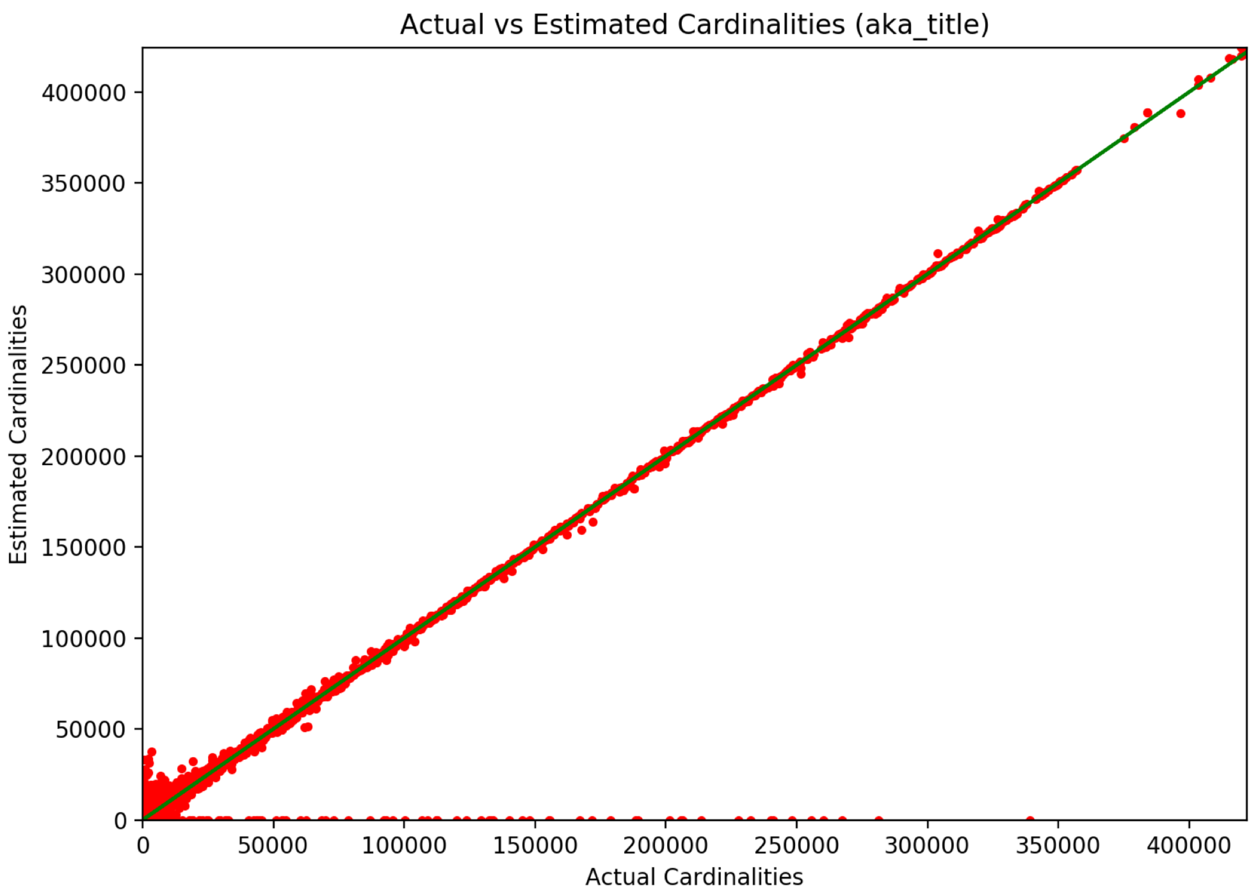
ReLU activation function

Training set : 15k

Test set : 3.7k

Relative error : 53%

Experiments

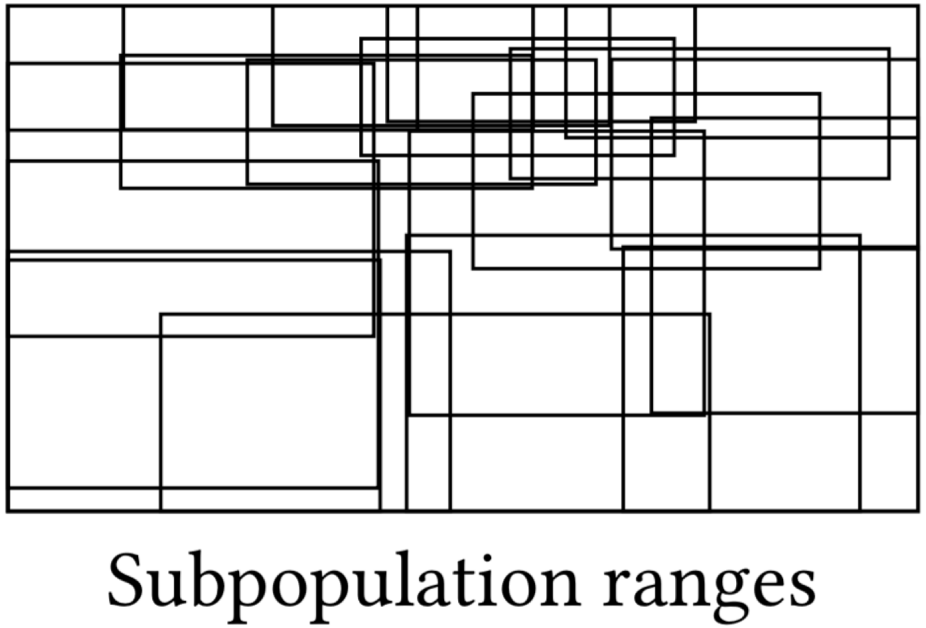


MIXTURE MODEL
p=10, m=2000, k=30

Training set : 15k
Test set : 3.7k
Relative error : 21%

Relative error on Training set = 11%

Database generation



- Generate w*|T| points in each hyper-rectangle.
- Total points = $\sum w_i * |T| = |T|$
- More the number of overlaps in a region, more points it will contain.

Our Contribution

- Implemented CEM using the mixture model approach.
- Achieved similar accuracy as the paper achieved.
- Identified the problem of good training data generation and how to tackle it.
- Compared our model’s performance with neural network.
- Suggested an approach for database generation.

Future work

- Solve the zero-cardinality problem by creating sub-populations that cover the entire domain space.
- Empirical generation of synthetic table and comparison with original table.


THANK YOU