Topics in Probabilistic and Statistical Databases

Lecture 9: Histograms and Sampling

Dan Suciu
University of Washington

References

- *Fast Algorithms For Hierarchical Range Histogram Construction*, Guha, Koudas, Srivastava, PODS 2002
- *Selectivity Estimation using Probabilistic Models*, Getoor, Taskar, Koller, SIGMOD 2001
- *Consistently estimating the selectivity of conjuncts of predicates*, Markl et al, VLDB 2005
- *On random sampling over joins*, Chaudhuri, Motwani, Narasayya, SIGMOD'99
- *Towards a robust query optimizer,* Babcock, Chaudhuri, SIGMOD 2005

Example

```
SELECT count(*)
FROM R
WHERE R.A=10 and R.B=20 and R.C=30
```

Think of this query as being issued during query optimization:
Optimizer wants to find out the size of a subplan

Assume |R| = 1,000,000,000
Can't scan R. Will use statistics instead

Histograms to the Rescue !

R.A =	…	9	10	11	…
count =	…	…	100,000,000	…	…

R.B =	…	19	20	21	…
count =	…	…	200,000,000	…	…

R.C =	…	29	30	31	…
count =	…	…	250,000,000	…	…

[Guha'2002]

Histogram Basics

- Main goal: estimate the size of range queries:

```
SELECT *
FROM   R
WHERE v1 ≤ R.A and R.A ≤ v2
```

- Special case: v=R.A

[Guha'2002]

Histogram Basics

- Given: an array A[1,n] of non-negative reals
- Define: $A[a,b] = (A[a]+...+A[b])/(b-a+1)$

Definition. A histogram of array A[1,n] using B buckets is specified by B+1 integers $0 \le b_1 \le \ldots \le b_{B+1} = n$.

$[b_i+1, b_{i+1}]$ is called a "bucket"; its value is $A[b_i+1, b_{i+1}]$

[Guha'2002]

Answering Range Queries

Definition. A range query is R_{ij} and its answer is:
$s_{ij} = A[i] + \ldots + A[j]$

The answer $\hat{s}_{ij}$ to a range query R_{ij} using a histogram is computed by using the "uniformity assumption". [Formula on the white board]

Definition. The error of R_{ij} is $(\hat{s}_{ij} - s_{ij})^2$

[Guha'2002]

Optimal Histograms

- Given:
  - A workload W of range queries R_{ij}
  - A weight w_{ij} for each query
- Compute a histogram that minimizes

$$\boxed{\Sigma\, w_{ij}\, (\hat{s}_{ij} - s_{ij})^2}$$

Optimal Histograms

- V-optimal histograms:
  - Single point queries: $W=\{R_{11}, \ldots, R_{nn}\}$
  - All weights are equal
  - Computing V-optimal histogram [IN CLASS]
- Optimal histograms for hierarchical queries
  - Workload forms a hierarchy
  - Computable in PTIME

Multidimensional Histograms

- Main goal: estimate the size of multi-range queries:

```
SELECT *
FROM   R
WHERE u1 ≤ R.A and R.A ≤ v1
        and u2 ≤ R.B and R.B ≤ v2
        and …
```

Multidimensional Histograms

Two issues:

- Which dimensions to choose ?
- How do we compute the optimal histogram ?
  - NP-hard for 2 dimensions [S. Muthukrishnan, V. Poosala, and T. Suel, ICDT 1999]

Will discuss only issue 1

[Getoor'2001]

Which Dimensions to Choose

- Use graphical models and exploit conditional independences

[Getoor'2001]

Probabilistic Model of a Histogram

- $R(A_1, \ldots, A_n)$ = relation with n attributes
  - Duplicates possible, e.g. there are more attrs
- The joint probability distribution is:

$$\boxed{P(a_1, \ldots, a_n) = |\sigma_{A1=a1,\ldots,An=an}(R)| \ / \ |R|}$$

- Queries are now point queries

$$\boxed{Q(a_1, \ldots, a_n) = P(a_1, \ldots, a_n) * |R|}$$

[Getoor'2001]

Conditional Independences

Person(Name, Education, Income, Home-owner)

Education = high-school, college, MS

Income = low, medium, high

Home-owner = false, true

Assumption:

$$P(H \mid E, I) = P(H \mid I)$$

Then the point query becomes:

$$Q(H, E, I) = P(H \mid I) * P(I) * |R|$$

[Getoor'2001]

Conditional Independence → Histograms

E	I	H	$P(E, I, H)$
h	l	f	0.27
h	l	t	0.03
h	m	f	0.105
h	m	t	0.045
h	h	f	0.005
h	h	t	0.045
c	l	f	0.135
c	l	t	0.015
c	m	f	0.063
c	m	t	0.027
c	h	f	0.006
c	h	t	0.054
a	l	f	0.018
a	l	t	0.002
a	m	f	0.042
a	m	t	0.018
a	h	f	0.012
a	h	t	0.108

(a)

E	$P(E)$
h	0.5
c	0.3
a	0.2

I	E	$P(I \mid E)$
l	h	0.6
m	h	0.3
h	h	0.1
l	c	0.5
m	c	0.3
h	c	0.2
l	a	0.1
m	a	0.3
h	a	0.6

H	I	$P(H \mid I)$
t	l	0.1
f	l	0.9
t	m	0.3
f	m	0.7
t	h	0.9
f	h	0.1

(b)

E	$P(E)$
h	0.5
c	0.3
a	0.2

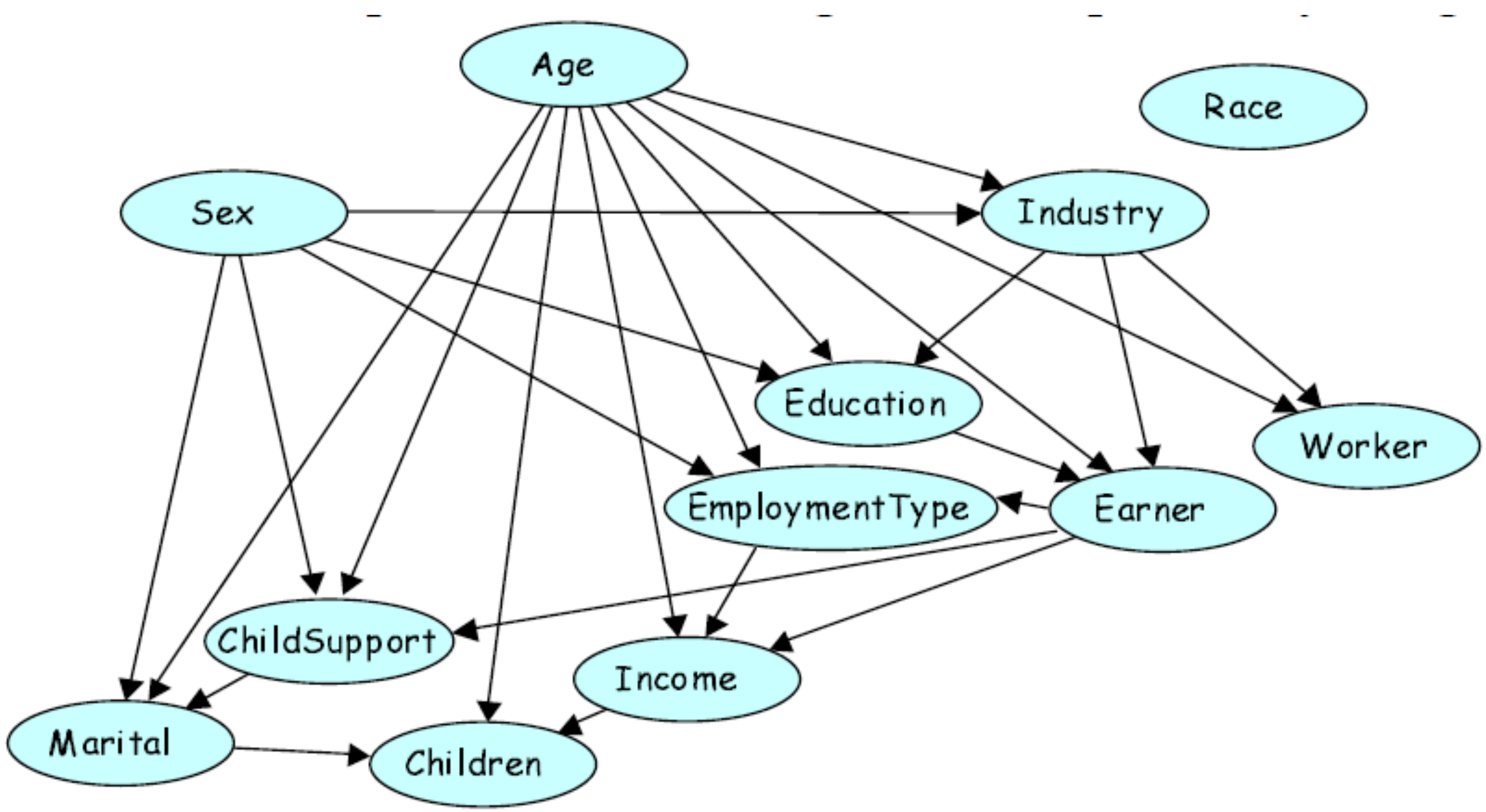
I	$P(I)$
l	0.47
m	0.30
h	0.23

H	$P(H)$
t	0.344
f	0.656

(c)

[Getoor'2001]

Bayesian Networks



Discussion

Multidimensional histograms remain difficult to use:

- Conditional independences may not hold
- Difficult to learn the BN
- Computing buckets remains expensive

[Markl'2005]

Consistent Estimation Problem

Recall: histogram entries are probabilities

R.A =	…	10	…
s_1 =	…	0.1	…

R.B =	…	20	…
s_2 =	…	0.2	…

R.C =	…	30	…
s_3 =	…	0.25	…

```
SELECT count(*)
FROM R
WHERE R.A=10 and R.B=20 and R.C=30
```

What's your estimate ?

[Markl'2005]

Consistent Estimation Problem

R.A =	…	10	…
s_1 =	…	0.1	…

R.B =	…	20	…
s_2 =	…	0.2	…

R.C =	…	30	…
s_3 =	…	0.25	…

R.AB	…	10,20	…
s_{12} =	…	0.05	…

R.BC	…	20,30	…
s_{13} =	…	0.03	…

```
SELECT count(*)
FROM R
WHERE R.A=10 and R.B=20 and R.C=30
```

What's your estimate now ?

[Markl'2005]

Problem Statement

- Given
  - Multivariate Statistics, MVS
  - Query q
- Estimate q from the MVS

- Issue:
  - Many ways to use the MVS
  - Inconsistent answers

[Markl'2005]

Example

- Relation: R(A,B,C)
- MVS: P(A), P(B), P(C), P(A,B), P(B,C)

- Estimate query size: $\sigma_{A=a,\ B=b,\ C=c}(R)$

- Equivalently: compute P(a,b,c)

No Unique Solution !

[Markl'2005]

The Consistency Problem

Different possible answers:

- $P(a,b,c) \approx P(a,b) * P(c)$
- $P(a,b,c) \approx P(a) * P(b,c)$
- $P(a,b,c) \approx P(a) * P(b) * P(c)$
- $P(a,b,c) \approx P(a,b) * P(b,c) / P(b)$

Which independence(s) does each formula assume ?

[Markl'2005]

Simplify Probabilities

- New probability space on $\{(x,y,z) \mid (x,y,z) \in \{0,1\}^3\}$ defined by:
- Randomly select a tuple t from R
  - $x=1$ iff t.A=10
  - $y=1$ iff t.B=20
  - $z=1$ iff t.C=30

- E.g. $P(1,0,1) = P(A=a, B\neq b, C=c)$

[Markl'2005]

Modeling Histograms as ProbDB

- There are eight possible worlds, need their probs
- The five histograms lead to 5+1 = 6 constraints:

x	y	z	P
0	0	0	x_{000}
0	0	1	x_{001}
0	1	0	x_{010}
0	1	1	x_{011}
1	0	0	x_{100}
1	0	1	x_{101}
1	1	0	x_{110}
1	1	1	x_{111}

$x_{000}+x_{001}+x_{010}+x_{011}+x_{100}+x_{101}+x_{110}+x_{111}= 1$

$x_{100}+x_{101}+x_{110}+x_{111}= P(a)$

$x_{010}+x_{011}+x_{110}+x_{111}= P(b)$

$x_{001}+x_{011}+x_{101}+x_{111}= P(c)$

$x_{110}+x_{111}= P(a,b)$

$x_{011}+x_{111}= P(b,c)$

But underdetermined.
How do we choose ?

Entropy Maximization Principle

- Let $\mathbf{x}=(x_1,x_2, \ldots)$ be a probability distribution
- The entropy is:

$$\boxed{H(\mathbf{x}) = -(x_1 \log(x_1) + x_2 \log(x_2)+\ldots)}$$

- The ME principle is:
  "among multiple probability distributions, choose the one with maximum entropy"

Solving ME

- In our example: find $x_{000}, \ldots, x_{111}$ s.t.:

$$p_{\varnothing} = x_{000} + \ldots + x_{111} - 1 = 0$$
$$p_a = x_{100} + x_{101} + x_{110} + x_{111} - P(a) = 0$$
$$p_b = x_{010} + x_{011} + x_{110} + x_{111} - P(b) = 0$$
$$p_c = x_{001} + x_{011} + x_{101} + x_{111} - P(c) = 0$$
$$p_{ab} = x_{110} + x_{111} - P(a,b) = 0$$
$$p_{bc} = x_{011} + x_{111} - P(b,c) = 0$$

maximize(H)

where $H = -(x_{000} \log(x_{000}) + \ldots + x_{111} \log(x_{111}))$

Solving ME

- The Lagrange multipliers: define a constant λ_s for every constraint p_s, then define:

$$\boxed{f(x_{000}, \ldots, x_{111}) = \Sigma_s \lambda_s p_s - H}$$

- Solve the following:

$$\boxed{\begin{array}{l} \partial f / \partial x_{000} = 0 \\ \ldots \\ \partial f / \partial x_{111} = 0 \end{array}}$$

Solving ME

- The system becomes:

$$\boxed{\forall t \text{ in } \{0,1\}^3: \Sigma_{s \subseteq t} \lambda_s + \log(x_t) + 1 = 0}$$

- In our example, this is:

$$\begin{array}{ll}
t=000: & \lambda_\varnothing + \log(x_{000}) + 1 = 0 \\
t=001: & \lambda_\varnothing + \lambda_c + \log(x_{001}) + 1 = 0 \\
t=010: & \lambda_\varnothing + \lambda_b + \log(x_{010}) + 1 = 0 \\
t=011: & \lambda_\varnothing + \lambda_b + \lambda_b + \lambda_{bc} + \log(x_{011}) + 1 = 0 \\
\ldots\ldots &
\end{array}$$

Solving ME

- The solution has the following form:

$$\boxed{\forall t \text{ in } \{0,1\}^3\text{:}\ \ x_t = \Pi_{s \subseteq t}\, \alpha_s}$$

- Here α_s are parameters: one parameter for each MVS
- To solve for the parameters ➔ nonlinear system of equations

Solving ME

- In our example, this is:

$$
\begin{aligned}
x_{000} &= \alpha_\emptyset \\
x_{001} &= \alpha_\emptyset \alpha_c \\
x_{010} &= \alpha_\emptyset \alpha_b \\
x_{011} &= \alpha_\emptyset \alpha_b \alpha_c \alpha_{bc} \\
x_{100} &= \alpha_\emptyset \alpha_a \\
x_{101} &= \alpha_\emptyset \alpha_a \alpha_c \\
x_{110} &= \alpha_\emptyset \alpha_a \alpha_b \alpha_{ab} \\
x_{111} &= \alpha_\emptyset \alpha_a \alpha_b \alpha_c \alpha_{ab} \alpha_{bc}
\end{aligned}
$$

- Next, need to solve a nonlinear system
- [WHICH ONE ?]
- Good luck solving it !

Summary of Histograms

- Naïve probabilistic model:
  - Select randomly a tuple from the relation R
- Limited objective:
  - Estimate range queries
  - But they do this pretty well
- Widely used in practice

A Much Simpler Approach: Sampling

- R has N=1,000,000,000 tuples
- Compute (offline) a sample of size n =500

```
SELECT count(*)
FROM R
WHERE R.A=10 and R.B=20 and R.C=30
```

Evaluate the query on the sample ➔ 8 tuples

What is your estimate ?

[Chaudhuri'99]

Sampling from Databases

Two usages:

- For query size estimation:
  - Keep a random sample, use it to estimate queries
- Approximate query answering:
  - Answer a query by sampling from the database and computing the query only on the sample

[Chaudhuri'99]

Sampling from Databases

SAMPLE(R, f), where $f \in [0,1]$, and $|R|=n$

Three semantics:

- Sampling with replacement WR
  - Sample fn elements from R, each independently
- Sampling without replacement WoR
  - Sample a subset of size fn from R
- Bernoulli sample, or coin flip CF
  - For each element in R, flip a coin with prob f

Random Sampling from Databases

- Given a relation $R = \{t_1, \ldots, t_n\}$
- Compute a sample S of R

Random Sample of Size 1

- Given a relation $R = \{t_1, \ldots, t_n\}$
- Compute random element s of R

Q: What is the probability space ?

Random Sample of Size 1

- Given a relation $R = \{t_1, \ldots, t_n\}$
- Compute random element s of R

Q: What is the probability space ?

A: Atomic events: $t_1, \ldots, t_n$,
Probabilities: $1/n, 1/n, \ldots, 1/n$

Random Sample of Size 1

```
Sample(R) {
  r = random_number(0..2^32–1);
  n = |R|;
  s = "the (r % n)'th element of R"
  return s;
}
```

Random Sample of Size 1

Sequential scan

```
Sample(R) {
  forall x in R do {
        r = random_number(0..1);
        if (r  < ???) s = x;
  }
  return s;
}
```

Random Sample of Size 1

Sequential scan

```
Sample(R) {  k = 1;
   forall x in R do {
         r = random_number(0..1);
         if (r< 1/k++) s = x;
   }
   return s;
}
```

Note: need to scan R fully. How can we stop early ?

Random Sample of Size 1

Sequential scan: use the size of R

```
Sample(R) {  k = 0;
   forall x in R do { k++;
         r = random_number(0..1);
         if (r< 1/(n - k +1) return x;
    }
   return s;
}
```

Binomial Sample or Coin Flip

In practice we want a sample > 1

```
Sample(R) {  S = emptyset;
   forall x in R do {
         r = random_number(0..1);
         if (r< p) insert(S,x);
   return S;
}
```

What is the problem with binomial sample ?

Binomial Sample

- The size of the sample S is not fixed
- Instead it is a random binomial variable of expected size pn
- In practice we want a guarantee on the sample size, i.e. we want the sample size = m

Fixed Size Sample WoR

Problem:

- Given relation R with n elements
- Given $m > 0$
- Sample m distinct values from R

What is the probability space ?

Fixed Size Sample WoR

Problem:

- Given relation R with n elements
- Given m > 0
- Sample m distinct values from R

What is the probability space ?

A: all subsets of R of size m, each has probability $1/\binom{n}{m}$

Reservoir Sampling: known population size

Here we want a sample S of fixed size m
from a set R of known size n

```
Sample(R) {  S = emptyset; k = 0;
  forall x in R do { k++;
        p = (m-|S|)/(n-k+1)
        r = random_number(0..1);
        if (r< p) insert(S,x);
  return S;
}
```

Reservoir Sampling: unknown population size

```
Sample(R) {  S = emptyset; k = 0;
   forall x in R do
         p = |S|/k++
         r = random_number(0..1);
         if (r< p) { if (|S|=m) remove a random
                                       element from S;
                     insert(S,x);}
   return S;
}
```

Question

- What is the disadvantage of not knowing the population size ?

Example: Using Samples

R has N=1,000,000,000 tuples
Compute (offline) a sample X of size n =500

```
SELECT count(*)
FROM R
WHERE R.A=10 and R.B=20 and R.C=30
```

Evaluate the query on the sample → 8 tuples
Thus E[p] = 8/500 = 0.0016

The Join Sampling Problem

- SAMPLE($R_1 \bowtie R_2$, f) without computing the join $J = R_1 \bowtie R_2$
- Example:
  $R_1(A,B) = \{(a_1,b_0), (a_2,b_1), \ldots, (a_2,b_k)\}$
  $R_2(A,C) = \{(a_2,c_0), (a_1,b_1), \ldots, (a_1,b_k)\}$
- A random sample of J cannot be obtained from a *uniform* random sample on R1 and on R2

Sampling over Joins

- Solution: use weighted sampling
- [IN CLASS]

Join Synopses

- [Acharya et al, SIGMOD'99]
- Idea: compute maximal key-foreign key joins
- Compute a sample S
- Then we can obtain a sample for any sub-join by projecting S

Example

R(<u>A</u>, B, C), S(<u>B</u>, D, J), T(<u>C</u>, E, F), U(<u>D</u>, G, H)
Join synopsis: sample Σ of $R \bowtie S \bowtie T \bowtie U$

```
SELECT count(*)
FROM S, U
WHERE S.D = U.D and S.J='a' and U.G='b'
```

Compute $\Sigma' = \Pi_{B,D,J,G,H}(\Sigma)$
This is an unbiased sample of $S \bowtie U$ [WHY ???]
Evaluate query on Σ' → 12 tuples
Estimate query size: $12 * |\Sigma'| / |S|$ [WHY ???]

Example

R has N=1,000,000,000 tuples
Compute (offline) a sample X of size n =500

```
SELECT count(*)
FROM R
WHERE R.A=10 and R.B=20 and R.C=30
```

Evaluate the query on the sample → 8 tuples
Thus E[p] = 8/500 = 0.0016

[Babock et al. SIGMOD'2005]

Robust Query Optimization

Traditional optimization:

- Plan 1: use index
- Plan 2: sequential scan

- The choice between 1 and 2 depends on the estimated selectivity
- E.g. for $p < 0.26$ the Plan 1 is better

[Babock et al. SIGMOD'2005]

Robust Query Optimization

The performance/predictability tradeoff:

- Plan 1: use index
  - If it is right ➔ ☺
  - If it is wrong ➔ ☹ MUST AVOID THIS !!
- Plan 2: sequential scan ➔ 😐

Optimizing performance may result in significant penalty, with some probabililty

[Babock et al. SIGMOD'2005]

Query Plan Cost

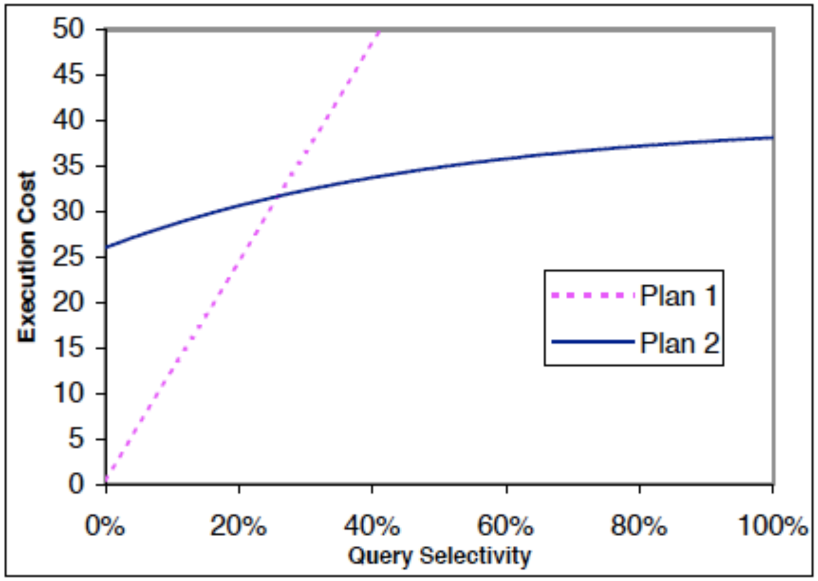


Figure 1: Execution Costs for Two Hypothetical Plans

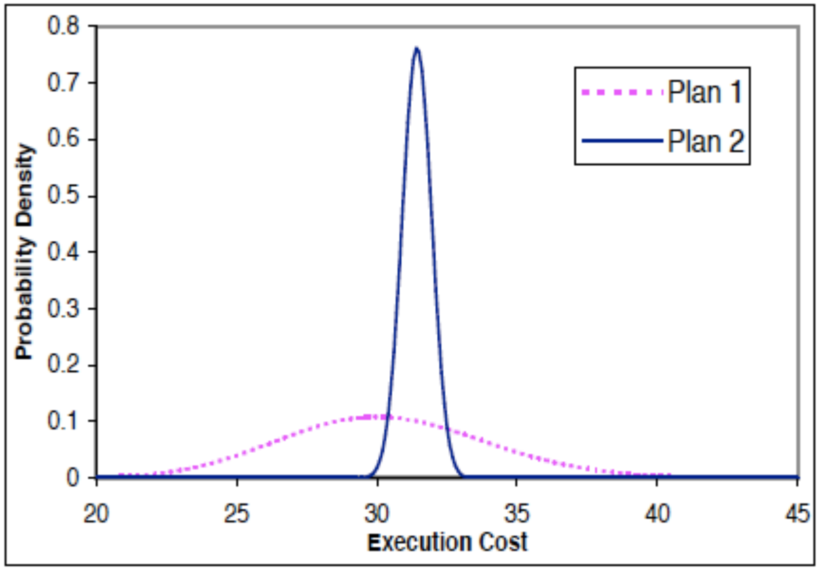
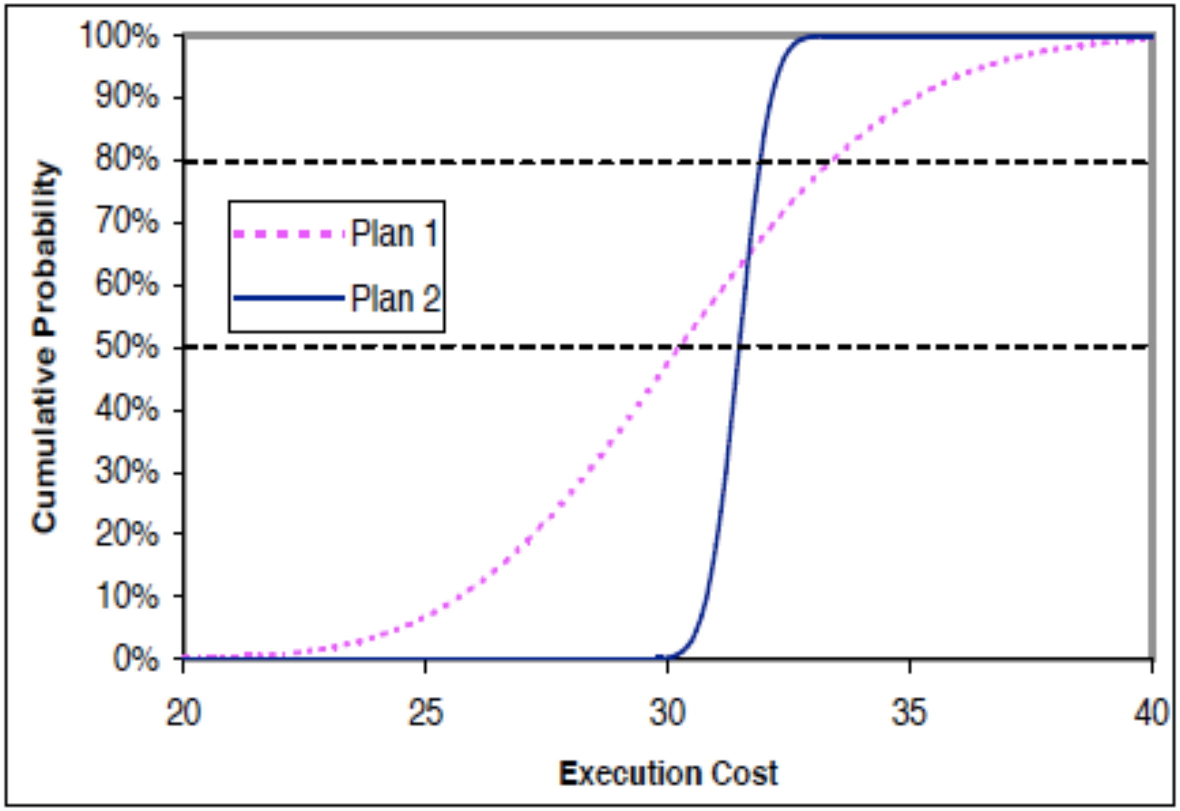


Figure 2: Probability Density Function for Execution Cost

[Babock et al. SIGMOD'2005]

Cumulative Distribution

User chooses confidence level T%.



T%=50% ➔ plans are chosen by expected cost;

T%=80% ➔ plans chosen by their cost at cumulative prob of 80%

[Babock et al. SIGMOD'2005]

The Probabilistic Database

R has N=1,000,000,000 tuples
Compute (offline) a sample X of size n =500

```
SELECT count(*)
FROM R
WHERE R.A=10 and R.B=20 and R.C=30
```

Evaluate the query on the sample ➔ 8 tuples
Thus E[p] = 8/500 = 0.0016

But what is the distribution of p ??

[Babock et al. SIGMOD'2005]

The Probabilistic Database

R has N=1,000,000,000 tuples

Compute (offline) a sample X of size n =500

A fraction k = 8 of X satisfy the predicate

An unknown fraction p of R satisfy the pred.

Denote f(z) = density function for p:

$$Pr[(a \leq p \leq b)|X] = \int_a^b f(z|X)dz.$$

[Babock et al. SIGMOD'2005]

The Probabilistic Database

Bayes' rule:

$$f(z|X) = \frac{\mathbf{Pr}[X|p = z]f(z)}{\int_0^1 \mathbf{Pr}[X|p = y]f(y)dy}$$

Next, compute each term (in class)

What is Pr[X | p=z] ? Assume X= w/ replacement

Whas is "the prior" f(z) ?

[Babock et al. SIGMOD'2005]

The Probabilistic Database

$$f(z|X) = \frac{z^{k-1/2}(1-z)^{n-k-1/2}}{\int_0^1 y^{k-1/2}(y-z)^{n-k-1/2}dy}$$

[Babock et al. SIGMOD'2005]

The Probabilistic Database

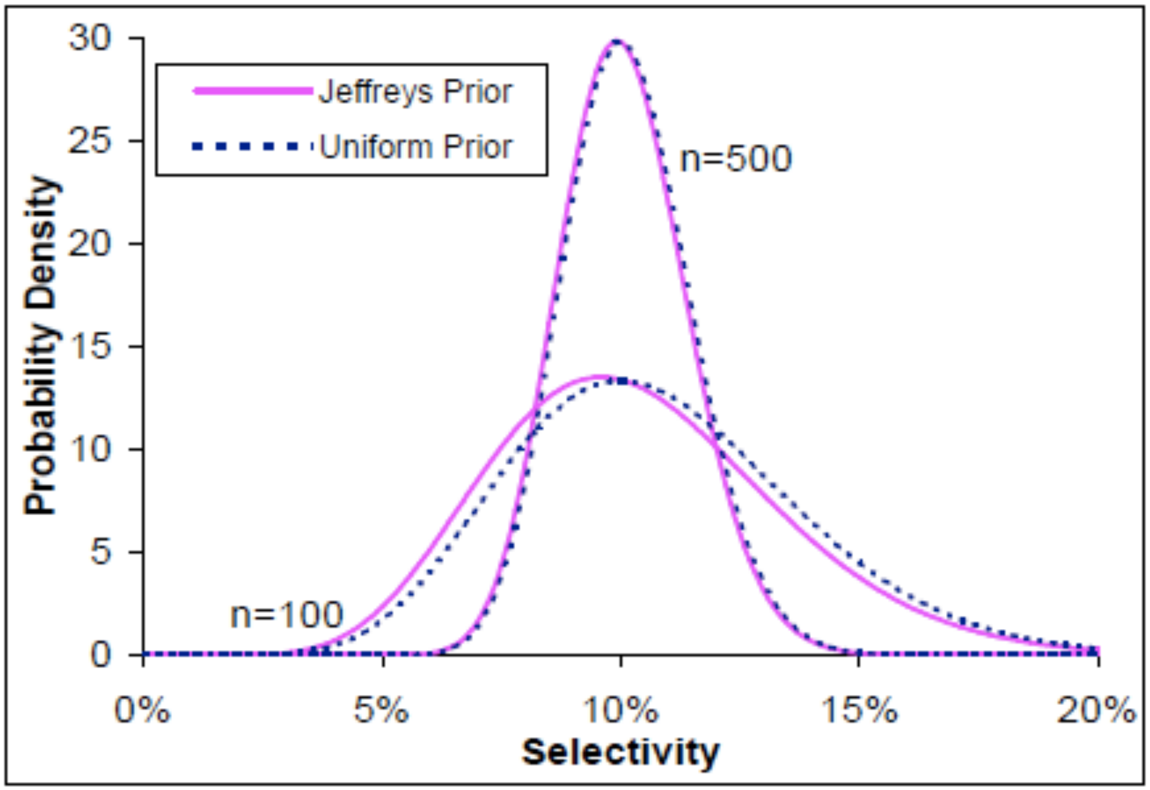


Figure 4: Sample Size Matters, Prior Doesn't