Errata and Proofs for Quickr [2]

Srikanth Kandula

1 Errata

We point out some errors in the SIGMOD version of our Quickr [2] paper.

- The transitivity theorem, in Proposition 1 of Quickr, has a revision in item iii. The revised version is Proposition 5 in this document.
- Definition 1 in Appendix B.2, which defines dominance, has a revision in the condition for c-dominance. The revised version is Definition 1 in this document. We also offer new definitions of equivalence and weak equivalence.
- The dominance pushdown rules in Appendix B.3 have, in general, been revised and substantially expanded. In particular, Proposition 7 in Quickr, which relates to projections that drop columns is simple but incomplete; Propositions 8 and 9 in this document consider the case of projections that rename columns and create new columns respectively. Proposition 8 in Quickr, which relates to pushing samplers past selections, is replaced with Proposition 10 in this document. Finally, we separate the Proposition 9 in Quickr, which pushes samplers past joins, into two separate propositions; Proposition 2 considers the special-case of foreign key joins and Proposition 12 considers other equijoins.

Overall, we believe this document supersedes Quickr [2] in terms of accuracy analysis and proofs.

2 Base case

2.1 Setup

Sampler immediately precedes a GROUP-BY that has one or more aggregates possibly with group-by columns.

2.2 Results

We offer:

- an unbiased estimator for each aggregate
- the variance of said estimator
- the group-miss probability
- a method to execute the above estimates in one effective pass over data

2.3 Details

QUICKR uses the *Horvitz-Thompson (HT) estimator*. Let $\mathcal{E}$ be a query plan with samplers. The core of $\mathcal{E}$, represented as $\Lambda(\mathcal{E})$, is a plan that is identical to $\mathcal{E}$ in all respects but without samplers. For each group G in the answer of $\Lambda(\mathcal{E})$, the sampled plan $\mathcal{E}$ outputs a subset of the rows in G, $\mathcal{E}(G) \subseteq G$. For each aggregate $w(G) = \sum_{t \in G} w(t)$, we offer an estimator $\hat{w}_{\mathcal{E}}(G)$.

$$\hat{w}_{\mathcal{E}}(G) = \sum_{t \in \mathcal{E}(G)} \frac{w(t)}{\mathbf{Pr}\left[t \in \mathcal{E}(G)\right]}. \tag{1}$$

It is easy to see that the above (HT) estimator is unbiased, i.e., $\mathbf{E}\left[\hat{w}_{\mathcal{E}}(G)\right] = w(G)$. Further, its variance is:

$$\mathbf{Var}\left[\hat{w}_{\mathcal{E}}(G)\right] = \sum_{i,j \in G} \left(\frac{\mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right]}{\mathbf{Pr}\left[i \in \mathcal{E}(G)\right] \mathbf{Pr}\left[j \in \mathcal{E}(G)\right]} - 1 \right) \cdot w(i)w(j). \tag{2}$$

From the sample $\mathcal{E}(G)$, $\mathbf{Var}\left[\hat{w}_{\mathcal{E}}(G)\right]$ can be estimated as:

$$\hat{\mathbf{Var}}\left[\hat{w}_{\mathcal{E}}(G)\right] = \sum_{i,j \in \mathcal{E}(G)} \left(\frac{\mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right]}{\mathbf{Pr}\left[i \in \mathcal{E}(G)\right] \mathbf{Pr}\left[j \in \mathcal{E}(G)\right]} - 1 \right) \cdot \frac{w(i)w(j)}{\mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right]}. \tag{3}$$

The probability that the answer will have group G is $\mathbf{Pr}\left[G\right]$.

$$\mathbf{Pr}\left[G\right] = 1 - \mathbf{Pr}\left[\wedge_{i \in G} \; i \notin \mathcal{E}(G)\right]. \tag{4}$$

2.4 Specific formulae for samplers used by Quickr

Recall our three samplers: *uniform sampler* Γ^{U}_p (uniform sampling probability p), *distinct sampler* $\Gamma^{\mathsf{D}}_{p,\mathcal{C},\delta}$ (each value of column set $\mathcal{C}$ has support at least δ in the sample), and *universe sampler* $\Gamma^{\mathsf{V}}_{p,\mathcal{C}}$ (sampling values of column set $\mathcal{C}$ with probability p). It could help to think of the universe sampler as first a random choice of a p fraction of the possible values of columns $\mathcal{C}$ and then a predicate that passes only the rows whose values of $\mathcal{C}$ have been chosen.

We apply the HT estimator to compute variance for all the samplers. To do so, we compute the terms $\mathbf{Pr}\left[i \in \mathcal{E}(G)\right]$ and $\mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right]$ for each sampler.

Proposition 1 (To Compute HT Estimator and the Variance)**.**

- *For Γ^{U}_p, for any tuples $i, j \in G$, we have $\mathbf{Pr}\left[i \in \mathcal{E}(G)\right] = p$, and, if $i \neq j$, $\mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right] = p^2$.*
- *For $\Gamma^{\mathsf{D}}_{p,\mathcal{C},\delta}$, let $g(i)$ be the set of tuples with the same values on $\mathcal{C}$ as tuple i in the input relation. We have*

  $$\mathbf{Pr}\left[i \in \mathcal{E}(G)\right] = \begin{cases} 1 & |g(i)| \leq \delta \\ \max(\frac{\delta}{|g(i)|}, p) & |g(i)| > \delta \end{cases};$$

  if $i \neq j$, $\mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right] = \mathbf{Pr}\left[i \in \mathcal{E}(G)\right] \mathbf{Pr}\left[j \in \mathcal{E}(G)\right]$.
- *For $\Gamma^{\mathsf{V}}_{p,\mathcal{C}}$, let $v_{\mathcal{C}}(i)$ be the value of tuple i on columns $\mathcal{C}$. We have $\mathbf{Pr}\left[i \in \mathcal{E}(G)\right] = p$, and*

  $$\text{if } i \neq j, \mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right] = \begin{cases} p & \text{if } v_{\mathcal{C}}(i) = v_{\mathcal{C}}(j) \\ p^2 & \text{otherwise} \end{cases}.$$

A few clarifications are worth mentioning. For the universe sampler, tuples that have the same value on $\mathcal{C}$ will either all belong to the sample or not; this leads to a higher variance in the estimator. To see this, substitute $\mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right]$ defined above into Eqn 3. Our implementation of the distinct sampler is rather involved. In order to finish in one pass and reduce the memory footprint, we implemented an approximate version of $\Gamma^{\mathsf{D}}_{p,\mathcal{C},\delta}$: we always include the first δ tuples for each value of $\mathcal{C}$, and then rely on a heavy hitter sketch to decide whether to use reservoir sampling or to use fixed-rate sampling for the rest of the tuples. The resulting probability $\mathbf{Pr}\left[i \in \mathcal{E}(G)\right]$ is lower bounded by the one stated in the above proposition and is a function of the position of the tuple i. Further, the joint probability $\mathbf{Pr}\left[i, j \in \mathcal{E}(G)\right]$ is upper bounded by $\mathbf{Pr}\left[i \in \mathcal{E}(G)\right] \mathbf{Pr}\left[j \in \mathcal{E}(G)\right]$.

Proposition 2 (Group Coverage Probability)**.** *When samplers immediately precede the aggregate, the probability that a group G appears in the answer is:*

- *For* Γ_p^{U}, $\mathbf{Pr}[G] = 1 - (1-p)^{|G|}$.
- *For* $\Gamma_{p,\mathcal{C},\delta}^{\mathsf{D}}$,

$$\mathbf{Pr}[G] \begin{cases} = 1, & \text{if } \mathcal{C} \text{ contains the group-by dimensions} \\ \geq 1 - (1-p)^{|G|}, & \text{otherwise} \end{cases} .$$

- *For* $\Gamma_{p,\mathcal{C}}^{\mathsf{V}}$, $\mathbf{Pr}[G] = 1 - (1-p)^{|G(\mathcal{C})|}$*, where* $G(\mathcal{C})$ *is the set of distinct values of tuples in* G *on dimensions* $\mathcal{C}$*.*

Proof. The proof for uniform sampler follows from the fact that each tuple is picked independently at random with probability p. For the distinct sampler, if the group-by columns are a subset of the stratification columns $\mathcal{C}$, then the group will appear in the answer. In the converse case, note that even though tuples in a group are picked in a correlated manner (based on their order in the input sequence), the joint probability of not picking any tuple is at least the probability shown. For the universe sampler, recall that a p fraction of the values of columns $\mathcal{C}$ are picked. Hence, a group with $G(\mathcal{C})$ distinct values has the corresponding likelihood of being picked. □

Using Proposition 2, we see that both uniform and distinct samplers rarely miss groups. Recall that QUICKR checks before introducing samplers that there is enough support, i.e., $p * |G| \geq k$. For example, when $k = 30$ and $p = 0.1$, the likelihood of missing G is below 10^{-14}. For the universe sampler, note that $|G(\mathcal{C})| \in [1, |G|]$. However, QUICKR uses the universe sampler only when stratification is not required (e.g., no groups), or when $|G(\mathcal{C})|$ is high and hence missing groups is rare (e.g., stratification on a column-set that is effectively independent with column-set $\mathcal{C}$).

2.5 Complexity of Computing Estimate and Error

The following proposition posits that accuracy computation requires only one effective pass over the sample.

Proposition 3 (Complexity)**.** *For each group* G *in the query output,* QUICKR *needs* $O(|\mathcal{E}(G)|)$ *time to compute the unbiased estimator of all aggregations and their estimated variance where* $|\mathcal{E}(G)|$ *is the number of sample rows from* G *output by expression* $\mathcal{E}$*.*

Proof. The proof follows from Propositions 1, 2 and Equations (1)-(4).

In more detail, QUICKR ensures that each tuple i in the sample $\mathcal{E}(G)$ also contains the probability $\mathbf{Pr}[i \in \mathcal{E}(G)]$ in its weight column. Hence, $\hat{w}_{\mathcal{E}}(G)$ can be computed in one scan using Equation (1). A naive way to compute $\hat{\mathbf{Var}}[\hat{w}_{\mathcal{E}}(G)]$ using (3) requires a self-join and can take quadratic time. However, as we will prove in detail later, all the plans generated by QUICKR will have no worse error than a corresponding plan that has one sampler at the root just below the aggregation. For such plans, we observe that only the pairs having $\mathbf{Pr}[i, j \in \mathcal{E}(G)] \neq \mathbf{Pr}[i \in \mathcal{E}(G)]\,\mathbf{Pr}[j \in \mathcal{E}(G)]$ need to be considered. For the uniform and distinct samplers, the summation in (3) goes to zero for $i \neq j$ and so their variance can be computed in one pass. For the universe sampler, there are two types of pairs: i) (i, j) with $v_{\mathcal{C}}(i) = v_{\mathcal{C}}(j)$, and ii) (i, j) with $v_{\mathcal{C}}(i) \neq v_{\mathcal{C}}(j)$; here $v_{\mathcal{C}}(i)$ denotes the value of tuple i on the universe columns $\mathcal{C}$. Per Proposition 1, the summation term is zero for pairs of the latter type. For the former type, we maintain per-group values in parallel and use a shuffle to put them back into (3). Since the number of groups can be no larger than the number of tuples, the computation is linear. Further the shuffle often has less work to do (one row per group) than the first pass leading to our one-*effective*-pass claim.

□

2.6 Types of Aggregations

For SUM and COUNT-like aggregates, the above analysis directly applies with $w(t) = t$ and $w(t) = 1$ respectively. Here, we discuss a few other common aggregates and the case where a result has multiple aggregations such as `SELECT x, SUM(y), COUNT(z)`. QUICKR requires user-defined aggregates to be annotated with functional expressions which it uses to obtain accuracy measures; details are left for future work.

Other aggregations: Analyzing `COUNT` directly follows from `SUM` by setting $w(t) = 1\ \forall t$. `AVG` translates to $\frac{\texttt{SUM}}{\texttt{COUNT}}$ but its variance is harder to analyze due to the division. In implementation, QUICKR substitutes `AVG` by `SUM/COUNT` and divides the corresponding estimators. QUICKR also supports `DISTINCT`, which translates to a group with no aggregations and `COUNT(DISTINCT)`. Error for the former is akin to missing groups analysis. For `COUNT(DISTINCT)`, we can use the GEE estimator [1] after adapting it to our samplers. We point out two specific cases where count distinct can have a better estimate than GEE. (1) For a distinct sampler with stratification columns containing the count distinct column, the estimator is same as computing count distinct on the sample. The error in this case is zero. (2) For a universe sampler with universe columns matching the count distinct column, our estimator is the value computed over the samples divided by the sampling probability. We defer analyzing the error for other aggregations to future work.

Multiple aggregation ops: QUICKR naturally extends to the case when multiple aggregations are computed over the same sampled input relation. The key observation is that the estimators for each aggregation only require the value in the sampled tuple, the corresponding *weight* which describes the probability with which the tuple was passed and in rare cases the type of the sample (e.g., for `COUNT DISTINCT`). The first two are available as columns in the sampled relation. The third we implement as a corrective rewriting after QUICKR chooses the samplers.

3 Sampling Dominance

Given two sampled expressions with the same core, we say one expression $\mathcal{E}_1$ is dominated by another expression $\mathcal{E}_2$ if and only if $\mathcal{E}_2$ has no higher variance and no higher probability of missing groups than $\mathcal{E}_1$. More formally, we have:

Definition 1 (Sampling Dominance)**.** *Given two expressions $\mathcal{E}_1$ and $\mathcal{E}_2$ with the same core and having $\mathcal{R}_1$ and $\mathcal{R}_2$ as the respective output relations and R as the output relation without samplers, we say $\mathcal{E}_2$* dominates *$\mathcal{E}_1$, or $\mathcal{E}_1 \overset{*}{\Rightarrow} \mathcal{E}_2$, iff*

$$(\textit{v-dominance } \mathcal{E}_1 \overset{\mathsf{v}}{\Rightarrow} \mathcal{E}_2) \quad \forall i, j \in R : \tag{5}$$
$$\frac{\mathbf{Pr}\left[i \in \mathcal{R}_1, j \in \mathcal{R}_1\right]}{\mathbf{Pr}\left[i \in \mathcal{R}_1\right]\mathbf{Pr}\left[j \in \mathcal{R}_1\right]} \geq \frac{\mathbf{Pr}\left[i \in \mathcal{R}_2, j \in \mathcal{R}_2\right]}{\mathbf{Pr}\left[i \in \mathcal{R}_2\right]\mathbf{Pr}\left[j \in \mathcal{R}_2\right]}, \textit{ and}$$
$$(\textit{c-dominance } \mathcal{E}_1 \overset{\mathsf{c}}{\Rightarrow} \mathcal{E}_2) \quad \forall \textit{ set of tuples } T \subseteq R : \tag{6}$$
$$\mathbf{Pr}\left[T \cap \mathcal{R}_1 = \varnothing\right] \geq \mathbf{Pr}\left[T \cap \mathcal{R}_2 = \varnothing\right]. \tag{7}$$

First, note that *sampler dominance* subsumes the *SOA-equivalence* definition [3]. Supposing p-dominance defined as follows:

$$(\text{p-dominance } \mathcal{E}_1 \overset{\mathsf{p}}{\Rightarrow} \mathcal{E}_2) \quad \forall i \in R : \mathbf{Pr}\left[i \in \mathcal{R}_1\right] \leq \mathbf{Pr}\left[i \in \mathcal{R}_2\right], \tag{8}$$

two expressions $\mathcal{E}_1, \mathcal{E}_2$ are SOA equivalent iff $\mathcal{E}_1 \overset{\mathsf{v,p}}{\Longrightarrow} \mathcal{E}_2$ and $\mathcal{E}_2 \overset{\mathsf{v,p}}{\Longrightarrow} \mathcal{E}_1$. Further, it is easy to see that c-dominance implies p-dominance but not the other way around.

Next, we note the intuitions: c-dominance says that all tuple-sets are more likely to appear in the output and helps to relate the group missing probability. Similarly, v-dominance helps relate the variance.

Hence, it is not hard to see that if $\mathcal{E}_2$ dominates $\mathcal{E}_1$, then the estimate $\hat{w}_{\mathcal{E}_2}(G)$ is better than $\hat{w}_{\mathcal{E}_1}(G)$ in terms of variance and group coverage probability. We formally state this result below.

Proposition 4 (Dominance and Accuracy)**.** *For any group G in the output of a SUM-like aggregate query, consider two execution plans $\mathcal{E}_1$ and $\mathcal{E}_2$ that have the same core $\Lambda(\mathcal{E}_1) = \Lambda(\mathcal{E}_2)$. We have:*

$$\textit{if } \mathcal{E}_1 \overset{\mathsf{v}}{\Rightarrow} \mathcal{E}_2, \quad \mathbf{Var}\left[\hat{w}_{\mathcal{E}_1}(G)\right] \geq \mathbf{Var}\left[\hat{w}_{\mathcal{E}_2}(G)\right].$$

$$\textit{if } \mathcal{E}_1 \overset{\mathsf{c}}{\Rightarrow} \mathcal{E}_2, \quad \mathbf{Pr}\left[G \textit{ is missed in } \mathcal{E}_1\right] \geq \mathbf{Pr}\left[G \textit{ is missed in } \mathcal{E}_2\right].$$

Proof. The proof follows by plugging Equations 5 and 6 in Equations 2 and 4 respectively. □

Colloquially, $\mathcal{E}_1 \overset{*}{\Rightarrow} \mathcal{E}_2$, implies that the latter plan yields a *better* answer.

Necessary conditions: Proposition 4 shows that equations 5, 6 are sufficient conditions to ensure no higher variance and no higher likelihood of missing groups respectively. We conjecture that Eqn. 5 is also a necessary condition. Eqn. 6 is not a necessary condition; we will note a instance later when discussing rules. Discovering the appropriate necessary condition for group coverage is future work.

Equivalence: We use $\overset{*}{\Leftrightarrow}$ to denote cases when the sampled expressions are equivalent. That is, $\mathcal{E}_1 \overset{*}{\Leftrightarrow} \mathcal{E}_2$ iff $\mathcal{E}_1 \overset{*}{\Rightarrow} \mathcal{E}_2$ and $\mathcal{E}_2 \overset{*}{\Rightarrow} \mathcal{E}_1$.

Conditional equivalence: We use $\mathcal{E}_1 \overset{\sim}{\Leftrightarrow} \mathcal{E}_2$ to denote cases where one or more of v-dominance and c-dominance may not hold in general. Yet, the variance of estimators and the group miss probability are equivalent when some *data properties* hold. These properties are described below. Conditional equivalence covers the cases where the error comparison between plans depends on properties of the data. QUICKR makes these data-dependent decisions in order to improve the performance of sampled plans.

Property 1. *Given relation R and column-sets $\mathcal{C}, \mathcal{D}$, the number of distinct values of $\mathcal{C}$ in R is equal to the number of distinct values of $\mathcal{C} \cup \mathcal{D}$ in R.*

This is equivalent to *functional dependency*. An example, using table and names from TPC-DS, follows:

$R = \texttt{store\_sales} \bowtie_{\texttt{item\_sk}} \texttt{item}, \mathcal{C} = \{\texttt{ss\_item\_sk}\}, \mathcal{D} = \{\texttt{i\_brand}\}$.

The property holds for this example because the column ss_item_sk is a foreign-key and hence has a many-to-one relationship with the i_brand column.

Property 2. *Given a relation R with tuples denoted as t, a selection (predicate) σ over columns $\mathcal{C}$, $\sigma^+_{\mathcal{C}}, \sigma^-_{\mathcal{C}}$ are the sets of tuples in R that are selected and filtered by the predicate respectively and another columnset $\mathcal{D}$ we have $\forall$ values d of $\mathcal{D}$, $\mathbf{Pr}\left[t_{\mathcal{D}} = d \mid \sigma^+_{\mathcal{C}}\right] = \mathbf{Pr}\left[t_{\mathcal{D}} = d \mid \sigma^-_{\mathcal{C}}\right]$.*

An example is $\mathcal{D} = \{\texttt{i\_color}\}$ and predicate $\sigma_{\mathcal{C}}$ is $\texttt{d\_year} > 2000$.

The property holds for this example because the color of items can be independent of their purchase or return date.

Property 3. *Given a foreign-key equijoin $R \bowtie_{\mathcal{C}} S$ with tuples denoted by t, where $\mathcal{C}$ is a foreign key of S and another columnset $\mathcal{D}$, we have $\forall$ values d of $\mathcal{D}$, $\mathbf{Pr}\left[t_{\mathcal{D}} = d \mid R \bowtie_{\mathcal{C}} S\right] = \mathbf{Pr}\left[t_{\mathcal{D}} = d \mid R - R \bowtie_{\mathcal{C}} S\right]$.*

An example is $\left(\texttt{store\_sales} \bowtie_{\texttt{ss\_item\_sk}=\texttt{i\_item\_sk}} \left(\sigma_{\texttt{i\_color} \neq \text{red}}(\texttt{item})\right)\right)$ and $\mathcal{D} = \texttt{ss\_store\_sk}$.

The property holds for this example because all stores have the same likelihood of having red colored items. This is analogous to Property 2.

Property 4. *Given a relation R and two columnsets $\mathcal{C}$ and $\mathcal{D}$, the values of columns in $\mathcal{C}$ are independent of the values of $\mathcal{D}$.*

An example is $R = \texttt{item}$, $\mathcal{C} = \{\texttt{i\_item\_sk}\}$, $\mathcal{D} = \{\texttt{i\_color}\}$.

The property holds because the key is an opaque indicator akin to a hash value and is hence independent of the color of the item.

Property 5. *Given a relation R and a columnset $\mathcal{C}$, all groups defined as tuples in R that have the same value of columns in $\mathcal{C}$, have size below some constant k.*

An example is $R = \texttt{item}$, $\mathcal{C} = \texttt{i\_item\_sk}$, $k = 1$. The property holds because $\mathcal{C}$ is a primary key in R.

Probabilistic equivalence: We use $\mathcal{E}_1 \overset{\rightrightarrows}{\Leftrightarrow} \mathcal{E}_2$ to denote cases where the variance of estimators and the group miss probability of the two expressions *converge* under some conditions; that is, the expressions have similar error only in a probabilistic sense. Similar to conditional equivalence, probabilistic equivalence, is dependent on data properties. However, in addition it is also only a probabilistic guarantee. QUICKR uses probabilistic equivalence in order to improve the performance of sampled plans.

Intuition: Given a relation R and groups defined as tuples that have the same value on a columnset $\mathcal{D}$, consider the condition when the number of tuples in each group tends to ∞. It is possible to have error converge when the support of every group increases.

An example is $R = \texttt{store\_sales}, \mathcal{D} = \{ss\_store\_sk\}$.

Every store has thousands or more tuples in the store_sales table; further, this *support* increases when the table contains data for longer time-periods.

We offer a precise setup where the above intuition (large support) holds.

Condition 1. *Given a database D, let D^i be a database generated by concatenating i copies of database D; that is each relation in D^i is exactly i times larger. To compare the error for two sampled plans for some query q, in a probabilistic sense, compare the error of those plans on database D^i as i goes to ∞.*

An example is $R = \texttt{store\_sales}, q = \textbf{SELECT } ss\_store\_sk, COUNT(*)$ $\textbf{FROM } store\_sales \textbf{ WHERE } SelectivePredicate = true$. The two plans being compared are an unsampled plan that applies on all of the input and another that applies on a uniform sample of the input.

Since the predicate can have a very small selectivity, the uniform sample can miss some groups and have high variance in its estimate of the COUNT. However, as the size of the database grows via concatenation, the error of the sampled plan will converge to zero (the error of the unsampled plan).

4 Establishing transitivity of dominance

We ask whether dominance holds transitively. That is: given execution plans $\mathcal{E}_1 \stackrel{*}{\Rightarrow} \mathcal{E}_2$, does $\sigma_C(\mathcal{E}_1) \stackrel{*}{\Rightarrow} \sigma_C(\mathcal{E}_2)$? Similarly for project and join. That is, also given $\mathcal{F}_1 \stackrel{*}{\Rightarrow} \mathcal{F}_2$, does $\mathcal{E}_1 \bowtie_C \mathcal{F}_1 \stackrel{*}{\Rightarrow} \mathcal{E}_2 \bowtie_C \mathcal{F}_2$?

4.1 Result

Proposition 5 (Dominance Transitivity)**.** *For pairs of expressions $\mathcal{E}_1$, $\mathcal{E}_2$ and $\mathcal{F}_1$, $\mathcal{F}_2$ that are equivalent if all samplers are removed:*

i) $\mathcal{E}_1 \stackrel{*}{\Rightarrow} \mathcal{E}_2$ *implies* $\pi(\mathcal{E}_1) \stackrel{*}{\Rightarrow} \pi(\mathcal{E}_2)$*;*

ii) $\mathcal{E}_1 \stackrel{*}{\Rightarrow} \mathcal{E}_2$ *implies* $\sigma(\mathcal{E}_1) \stackrel{*}{\Rightarrow} \sigma(\mathcal{E}_2)$*;*

iii) $\mathcal{E}_1 \stackrel{*}{\Rightarrow} \mathcal{E}_2$ *and* $\mathcal{F}_1 \stackrel{*}{\Rightarrow} \mathcal{F}_2$ *implies* $\mathcal{E}_1 \bowtie \mathcal{F}_1 \stackrel{*}{\Rightarrow} \mathcal{E}_2 \bowtie \mathcal{F}_2$*, if samplers in $\mathcal{E}_i$ are independent on samplers in $\mathcal{F}_i$.*

As a corollary, dominance holds transitively with projects, selects and joins for the samplers used by QUICKR. Specifically, uniform and distinct samplers are independent of each other. And, below an aggregation, QUICKR uses universe sampling for only one set of columns and the universe samplers pick the same random portion of the hashspace of those columns.

4.2 Proof

4.2.1 Projection

Projection only affects the columns in each tuple, hence transitivity trivially holds.

4.2.2 Selection

Observe that Equations 5 and 6 need only hold for tuples that appear in the answer *after* the selection. Tuples that do not pass the select will not appear in the answer. Further, for any tuple i that appears in the answer $\mathbf{Pr}\left[i \in \sigma_C(\mathcal{E}_1)\right] = \mathbf{Pr}\left[i \in \mathcal{E}_1\right]$. Plugging this into equations 5 and 6 concludes this proof.

4.2.3 Join

A join can be thought of as a cross-product followed by a selection. Per the above case for selection, it suffices to show transitivity for a cross-product. Let $\mathcal{R}_i$ and $\mathcal{S}_i$ be the output relations for the expressions $\mathcal{E}_i$ and $\mathcal{F}_i$ respectively. We want to prove $\mathcal{E}_1 \times \mathcal{F}_1 \stackrel{*}{\Rightarrow} \mathcal{E}_2 \times \mathcal{F}_2$.

For v-dominance, we have to show that:

$$\frac{\mathbf{Pr}\left[(r,s)\in\mathcal{R}_1\times\mathcal{S}_1,(r',s')\in\mathcal{R}_1\times\mathcal{S}_1\right]}{\mathbf{Pr}\left[(r,s)\in\mathcal{R}_1\times\mathcal{S}_1\right]\mathbf{Pr}\left[(r',s')\in\mathcal{R}_1\times\mathcal{S}_1\right]}$$
$$\geq\frac{\mathbf{Pr}\left[(r,s)\in\mathcal{R}_2\times\mathcal{S}_2,(r',s')\in\mathcal{R}_2\times\mathcal{S}_2\right]}{\mathbf{Pr}\left[(r,s)\in\mathcal{R}_2\times\mathcal{S}_2\right]\mathbf{Pr}\left[(r',s')\in\mathcal{R}_2\times\mathcal{S}_2\right]}. \tag{9}$$

Further, for c-dominance, for all tuple-sets T we have to show that:

$$\mathbf{Pr}\left[T\notin\mathcal{R}_1\times\mathcal{S}_1\right]\geq\mathbf{Pr}\left[T\notin\mathcal{R}_2\times\mathcal{S}_2\right]. \tag{10}$$

Since the samplers on either side of the cross-product are independent of each other, we have:

$$\mathbf{Pr}\left[(r,s)\in\mathcal{R}_i\times\mathcal{S}_i\right]=\mathbf{Pr}\left[r\in\mathcal{R}_i\right]\mathbf{Pr}\left[s\in\mathcal{S}_i\right] \tag{11}$$

and

$$\mathbf{Pr}\left[(r,s)\in\mathcal{R}_i\times\mathcal{S}_i,(r',s')\in\mathcal{R}_i\times\mathcal{S}_i\right]=\mathbf{Pr}\left[r\in\mathcal{R}_i,r'\in\mathcal{R}_i\right]\cdot\mathbf{Pr}\left[s\in\mathcal{S}_i,s'\in\mathcal{S}_i\right]. \tag{12}$$

The proof follows by plugging equations 11, 12 in equations 9, 10.

5 Dominance Pushdown rules

5.1 Preliminaries

Proposition 6 (Switching Rule)**.** *For any relation R, we have*

$$\Gamma^{\mathsf{V}}_{p,\mathcal{C}}(R) \overset{*}{\Rightarrow} \Gamma^{\mathsf{U}}_{p}(R) \overset{*}{\Rightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{C},\delta}(R).$$

Proof. To see $\Gamma^{\mathsf{V}}_{p,\mathcal{C}}(R) \overset{*}{\Rightarrow} \Gamma^{\mathsf{U}}_{p}(R)$: consider that with both samplers a tuple is picked with probability p. However, tuples having the same value on universe columns are either all picked or none are picked. Hence, a pairs of such tuples (same value on $\mathcal{C}$) is picked with probability p as opposed to p^2 with the uniform sampler. Worse, a set of n such tuples is missed with probability $1-p$ as opposed to $(1-p)^n$ for the uniform sampler.

For the distinct sampler, note that each tuple is picked with probability at least p; if the tuple is seen before the frequency threshold is hit, the probability is higher. Further, the likelihood of picking a pair or set of tuples is never worse than picking each tuple independently with probability p. Hence, $\Gamma^{\mathsf{U}}_{p}(R) \overset{*}{\Rightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{C},\delta}(R)$. □

Proposition 7 (Picking sampler parameters)**.** *For any relation R, we have*

$$\Gamma^{\mathsf{U}}_{p_1}(R) \overset{*}{\Rightarrow} \Gamma^{\mathsf{U}}_{p_2}(R) \text{ if } p_1 \le p_2,$$

$$\Gamma^{\mathsf{V}}_{p_1,\mathcal{C}}(R) \overset{*}{\Rightarrow} \Gamma^{\mathsf{V}}_{p_2,\mathcal{C}}(R) \text{ if } p_1 \le p_2,$$

$$\Gamma^{\mathsf{D}}_{p_1,\mathcal{C}_1,\delta_1}(R) \overset{*}{\Rightarrow} \Gamma^{\mathsf{D}}_{p_2,\mathcal{C}_2,\delta_2}(R) \text{ if } p_1 \le p_2, \mathcal{C}_1 \subseteq \mathcal{C}_2, \delta_1 \le \delta_2.$$

Proof. The proof follows trivially. The case of universe sampler is worth noting; instances with a different column set are not comparable with each other. A caveat w.r.t. the distinct sampler. The dominance always holds for $\mathcal{C}_1 = \mathcal{C}_2, \delta_1 = \delta_2$. For the case of $\mathcal{C}_1 \subseteq \mathcal{C}_2, \delta_1 \le \delta_2$, dominance holds iff the two samplers process input in the same order; else, c-dominance does not hold for tuple-sets that have tuples picked with weight of 1 (early due to frequency check). We continue to make this "process input in the same order" assumption in the rest of this section. □

5.2 Projections

Consider a general projection $\pi_{\mathcal{C}_d^-,\mathcal{C}_a\to\mathcal{C}_b,f_i(\mathcal{C}_i)=c_i}$; it may drop columns in $\mathcal{C}_d^-$, rename columns in $\mathcal{C}_a$ with corresponding names in $\mathcal{C}_b$ and create some new columns c_i using functions f_i over sets of columns $\mathcal{C}_i$.

Dropping columns does not affect sampler pushdown because the sampler, being outside the project, cannot possibly use these columns.

We first consider pushing past column renames first and then consider pushing past one function that creates a new column. It is easy to see that combining these cases leads to the general case.

Proposition 8 (Pushing past Projection (column renaming))**.** *Consider a relation R and a projection $\pi_{\mathcal{C}_a \to \mathcal{C}_b}$. Let the notation $\mathcal{D}_{\mathcal{C}_b \to \mathcal{C}_a}$ denote replacing columns in $\mathcal{D}$ that are in the set $\mathcal{C}_b$ with corresponding columns from the set $\mathcal{C}_a$. We have:*
Rule-U1: $\Gamma^{\mathsf{U}}_{p}(\pi_{\mathcal{C}_a \to \mathcal{C}_b}(R)) \overset{*}{\Leftrightarrow} \pi_{\mathcal{C}_a \to \mathcal{C}_b}(\Gamma^{\mathsf{U}}_{p}(R))$;
Rule-V1: $\Gamma^{\mathsf{V}}_{p,\mathcal{D}}(\pi_{\mathcal{C}_a \to \mathcal{C}_b}(R)) \overset{*}{\Leftrightarrow} \pi_{\mathcal{C}_a \to \mathcal{C}_b}(\Gamma^{\mathsf{V}}_{p,\mathcal{D}_{\mathcal{C}_b \to \mathcal{C}_a}}(R))$;
Rule-D1: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\pi_{\mathcal{C}_a \to \mathcal{C}_b}(R)) \overset{*}{\Leftrightarrow} \pi_{\mathcal{C}_a \to \mathcal{C}_b}(\Gamma^{\mathsf{D}}_{p,\mathcal{D}_{\mathcal{C}_b \to \mathcal{C}_a},\delta}(R))$.

Proof. The proof is trivial: the expressions on both sides have the same probability of picking any tuple set in the answer. Do note that the stratification and the universe columnset are edited to revert the effect of renaming. □

Proposition 9 (Pushing past Project (new columns))**.** *Consider a relation R and a projection $\pi_{f(\mathcal{C})=c}$ (which we denote by just π for brevity), we have these rules:*
Rule-U1a: $\Gamma^{\mathsf{U}}_{p}(\pi(R)) \overset{*}{\Leftrightarrow} \pi(\Gamma^{\mathsf{U}}_{p}(R))$;
Rule-V1a: $\Gamma^{\mathsf{V}}_{p,\mathcal{D}}(\pi(R)) \overset{*}{\Leftrightarrow} \pi(\Gamma^{\mathsf{V}}_{p,\mathcal{D}}(R))$, *if $c \notin \mathcal{D}$*;
Rule-D1a: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\pi(R)) \overset{*}{\Leftrightarrow} \pi(\Gamma^{\mathsf{D}}_{p,\mathcal{D}-\{c\},\delta}(R))$, *if $c \notin \mathcal{D}$ or $\mathcal{C} \subseteq \mathcal{D}$*;
Rule-D1b: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\pi(R)) \overset{*}{\Rightarrow} \pi(\Gamma^{\mathsf{D}}_{p,\mathcal{D} \cup \mathcal{C}-\{c\},\delta}(R))$, *if $c \in \mathcal{D}$ and $\mathcal{C} \nsubseteq \mathcal{D}$*;
Rule-D1c: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\pi(R)) \overset{\sim}{\Leftrightarrow} \pi(\Gamma^{\mathsf{D}}_{p,\mathcal{D}-\{c\},\delta}(R))$ *(additional conditions needed).*

Proof. The proof for Rule-U1a is from noting that each row has the same likelihood to occur in either expression and that rows are sampled independently.

For the universe sampler, Rule-V1a, consider the case when the newly generated column c belongs to the universe columnset. We know the universe sampler uses the value of column c. It is not possible to compute the value of c without having the sampler subsume function f. In the converse case (i.e. $c \notin \mathcal{D}$), the proof follows as above: each tuple-set is sampled with the same probability in the expressions on left and right.

For the distinct sampler, Rule-D1a, note that the likelihood of a row being picked by the sampler only depends on values of columnset $\mathcal{D}$. If the newly generated column is not in $\mathcal{D}$, sampling before project is equivalent. Furthermore, because f is a function, if the domain of the function (column set $\mathcal{C}$) is contained in $\mathcal{D}$, then stratifying on $\mathcal{D} - \{c\}$ suffices (functions cannot be one-many).

Rule-D1b considers the converse case to Rule-D1a; the newly generated column is in the columnset $\mathcal{D}$ and the domain of the function is not contained in $\mathcal{D}$. The expression on the right stratifies additionally on the domain of the function.

By definition, a function is either one-to-one or many-to-one; hence every distinct value of the desired column set $\mathcal{D}$ will be adequately represented by the expression on the right. However, the stratification columnset is strictly larger on the right, hence equivalence does not hold but dominance holds (proof follows from invoking the respective portion of Proposition 7).

Rule-D1c is a conditional equivalence that holds under Property 1. Intuitively, the additional stratification is an over-kill when some columns in $\mathcal{D} - \{c\}$ have a many-to-one relationship with either the column c or columnset $\mathcal{C}$ (i.e. $\mathcal{D} - \{c\} \vdash c$ or $\mathcal{D} - \{c\} \vdash \mathcal{C}$). □

Note that it is possible to invoke Rule-D1b for some functions and Rule-D1c for others.

5.3 Selections

We now consider the case of a selection with a single predicate specified over columns $\mathcal{C}$ of R. Conjunctions of predicates can be decomposed as a sequence of predicates. Disjunctions are, for simplicity, collapsed into one predicate. QUICKR uses standard QO methods to explore alternatives here.

Proposition 10 (Pushing past Selection)**.** *For any relation R and a selection $\sigma_{\mathcal{C}}$ with selection formula on a subset $\mathcal{C}$ of columns of R,*
Rule-U2: $\Gamma^{\mathsf{U}}_{p}(\sigma_{\mathcal{C}}(R)) \overset{*}{\Leftrightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{U}}_{p}(R))$;
Rule-V2: $\Gamma^{\mathsf{V}}_{p,\mathcal{D}}(\sigma_{\mathcal{C}}(R)) \overset{*}{\Leftrightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{V}}_{p,\mathcal{D}}(R))$;
Rule-D2: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\sigma_{\mathcal{C}}(R)) \overset{*}{\Rightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{D}}_{p,\mathcal{D}\cup\mathcal{C},\delta}(R))$;
Rule-D2a: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\sigma_{\mathcal{C}}(R)) \overset{\sim}{\Leftrightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{D}}_{p,\mathcal{D},\frac{\delta}{\sigma^{\mathsf{SS}}}}(R))$, *where* $\sigma^{\mathsf{SS}} = \frac{|\sigma_{\mathcal{C}}(R)|}{|R|}$ *is the selectivity of $\sigma_{\mathcal{C}}$ on R (additional conditions needed);*
Rule-D2b: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\sigma_{\mathcal{C}}(R)) \overset{*}{\Leftrightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R))$, *if* $\mathcal{C} \subseteq \mathcal{D}$;
Rule-D2c: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\sigma_{\mathcal{C}}(R)) \overset{\sim}{\Leftrightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R))$ *(additional conditions needed);*

Proof. The proof for uniform sampler, Rule-U2, is similar to the case with Rule-U1: the expressions on either side have the same likelihood of picking any tuple-set in the answer.

The proof for the universe sampler, Rule-V2 follows from a similar observation: the expressions on either side retrieve the exact same tuples because QUICKR makes the two universe samplers pick the same values of columnset $\mathcal{D}$.

The proof for distinct sampler, Rule-D2, relies on the observation that when the expressions on either side process input in the same order, every set of tuples is strictly more likely to be passed by the expression on the right. Without stratifying on the selection columns $\mathcal{C}$, it is possible that the tuples picked by the sampler for some group (distinct value of columnset $\mathcal{D}$) may all be filtered by the selection. The added stratification makes the sampler pass some tuples for every distinct value of columnset $\mathcal{D} \cup \mathcal{C}$; since the predicate is on the value of columns $\mathcal{C}$, every group will receive some tuples regardless of which tuples the predicate filters out.

Rule-D2b is an important simple case. When the predicate columns are already contained in the stratification set, the two expressions pass every tuple-set with the same probability.

Rule-D2c is a conditional equivalent alternative to the above rule. It applies when Property 1 holds. If there is a functional dependence between the stratification columns $\mathcal{D}$ and the columns used in the predicate $\mathcal{C}$ (i.e. $\mathcal{D} \vdash \mathcal{C}$), the expression on the right mimics the expression on the right in Rule-D2b and the same proof applies.

Rule-D2a is more complex but often useful. The expression on the right compensates for the fact that the selection will filter some of the passed tuples. It passes more tuples per group. The expression on the right is conditionally equivalent when Property 5 holds. That is, when all groups over $\mathcal{D}$ have support on R below $\frac{\delta}{\sigma^{\mathsf{ss}}}$. In the general case, equivalence holds precisely for groups that have support below the threshold while for groups with a larger support, the equivalence only holds probabilistically as detailed in Rule-D2x below.

As a corollary, note that $\Gamma^{\mathsf{U}}_{p}(\sigma_{\mathcal{C}}(R)) \overset{*}{\Rightarrow} \Gamma^{\mathsf{D}}_{p,\varnothing,\delta}(\sigma_{\mathcal{C}}(R)) \overset{*}{\Rightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{D}}_{p,\mathcal{C},\delta}(R))$. The proof follows by applying Proposition 6 and then Rule-D2. $\square$

Conjecture 1 (Pushing past Selection)**.** *For any relation R and a selection $\sigma_{\mathcal{C}}$ with selection formula on a subset $\mathcal{C}$ of columns of R,*
Rule-D2x: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\sigma_{\mathcal{C}}(R)) \overset{\rightleftarrows}{\Leftrightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{D}}_{p,\mathcal{D},\frac{\delta}{\sigma^{\mathsf{SS}}}}(R))$, *where* $\sigma^{\mathsf{ss}} = \frac{|\sigma_{\mathcal{C}}(R)|}{|R|}$ *is the selectivity of $\sigma_{\mathcal{C}}$ on R (additional conditions needed);*
Rule-D2y: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(\sigma_{\mathcal{C}}(R)) \overset{\rightleftarrows}{\Leftrightarrow} \sigma_{\mathcal{C}}(\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R))$ *(additional conditions needed).*

The proofs for probabilistic equivalence are complex and we leave them for future work.

Rule-D2y is a probabilistic equivalent alternative to Rule-D2c that holds under Condition 1. The expression on the right has larger error in general. However, as the support for all groups in the answer increases, the error for the right expression will converge to that of the left. Further, if Property 2 holds, that is, the selection is not correlated with the group, then, the convergence happens faster, i.e. with fewer support for groups.

Rule-D2x is the probabilistic equivalent version of Rule-D2a. It holds under Condition 1 and guarantees equivalence for groups with support below $\frac{\delta}{\sigma^{\mathsf{ss}}}$. Note that this rule dominates Rule-D2y and in general converges faster. If Property 2 holds, the convergence is faster, i.e. a smaller sample may suffice for the same size of the relation.

5.4 Equi-Joins

We begin with the special-case of a foreign-key join because it applies often and is a stepping stone to the general case.

Proposition 11 (Pushing past foreign-key Join)**.** *Given relations R and S being joined on columns $\mathcal{C}$ where $\mathcal{C}$ is a foreign-key in R and primary-key in S, let R_c and S_c denote the columns of R and S respectively and suppose $\mathcal{D}_R$ is a set generated from set $\mathcal{D}$ by replacing the columns in set $\mathcal{D}$ that belong to S_c with equivalent columns in R_c as per the equijoin conditions* [1]*, we have:*
Rule-U3: $\Gamma^{\mathsf{U}}_{p}(R \bowtie_{\mathcal{C}} S) \overset{*}{\Leftrightarrow} \Gamma^{\mathsf{U}}_{p}(R) \bowtie_{\mathcal{C}} S$;
Rule-V3: $\Gamma^{\mathsf{V}}_{p,\mathcal{D}}(R \bowtie_{\mathcal{C}} S) \overset{*}{\Leftrightarrow} \Gamma^{\mathsf{V}}_{p,\mathcal{D}_R}(R) \bowtie_{\mathcal{C}} S,$ *if* $\mathcal{D}_R \subseteq R_c$;
Rule-D3: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{*}{\Rightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R\cup\mathcal{C},\delta}(R) \bowtie_{\mathcal{C}} S,$ *if* $\mathcal{D}_R \subseteq R_c$;
Rule-D3a: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\approx}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R,\frac{\delta}{\bowtie^{\mathtt{ss}}}}(R) \bowtie_{\mathcal{C}} S,$ *if* $\mathcal{D}_R \subseteq R_c$ *where* $\bowtie^{\mathtt{ss}} = \frac{|R \bowtie_{\mathcal{C}} S|}{|R|}$*, is the selectivity of the join (additional conditions needed);*
Rule-D3b: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{*}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R,\delta}(R) \bowtie_{\mathcal{C}} S,$ *if* $\mathcal{C} \subseteq \mathcal{D}_R \subseteq R_c$;
Rule-D3c: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\approx}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R,\delta}(R) \bowtie_{\mathcal{C}} S,$ *if* $\mathcal{D}_R \subseteq R_c$ *(additional conditions needed);*
Rule-D3d: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\approx}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R-S_c,\delta}(R) \bowtie_{\mathcal{C}} S$ *(additional conditions needed);*
Rule-D3e: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\approx}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R-S_c,\frac{\delta}{\bowtie^{\mathtt{ss}}}}(R) \bowtie_{\mathcal{C}} S,$ *if* $\mathcal{D}_R \not\subseteq R_c$;
Rule-D3f: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{*}{\Rightarrow} \Gamma^{\mathsf{D}}_{p,(\mathcal{D}_R-S_c)\cup\mathcal{C},\delta}(R) \bowtie_{\mathcal{C}} S,$ *if* $\mathcal{D}_R \not\subseteq R_c$;
Rule-D3g: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\approx}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,(\mathcal{D}_R-S_c)\cup\mathcal{C},\frac{\delta}{\mathtt{sfm}}}(R) \bowtie_{\mathcal{C}} S,$ *if* $\mathcal{D}_R \not\subseteq R_c$ *where* $\mathtt{sfm} = \frac{\mathtt{NumDV}(R,(\mathcal{D}_R-S_c)\cup\mathcal{C})}{\mathtt{NumDV}(R,\mathcal{D}_R)}$;

Proof. Note that a foreign-key join implies that at most one row will match from the relation S for each row in R.

Rule-U3 follows directly; in the expressions on either side, the likelihood of a tuple appearing is p and tuples are sampled independently.

The proof for universe sampler, Rule-V3 has one additional caveat; if all the columns in set $\mathcal{D}$ appear from the relation R (that is, $\mathcal{D}_R \subseteq R_c$), then, the same

[1] Assume $\mathcal{D} = \{r_1, s_1, s_2\}$ and the join condition is $r_3 = s_1$ where columns r_i and s_i are from the relations R and S respectively. Then, $\mathcal{D}_R = \{r_1, r_3, s_2\}$.

reasoning as for Rule-U2 applies. In the converse case, the universe sampler cannot be pushed. Note that we use column equivalence here.

When $\mathcal{D}_R \subseteq R_c$, that is, all stratification columns are from R, a foreign-key join can be considered to be a select on the join columns. Hence, Rule-D3 ... Rule-D3c mimic Rule-D2 ... Rule-D2c respectively. In particular, Rule-D3a is conditionally equivalent only when the support for all groups is below the threshold (noted as Property 5). In general, the expressions are equivalent for groups that have support below the threshold and only probabilistically equivalent for the remaining groups (noted below as Rule-D3x). Next, Rule-D3c is conditionally equivalent if Property 1 holds (functional dependence). Specifically, $\mathcal{D}_R \vdash \mathcal{C}$ and $\mathcal{D}_R \vdash \mathcal{D}$. The proof follows from corresponding proofs for selection.

Rule-D3d ... Rule-D3g cover the case when $\mathcal{D}_R \not\subseteq R_c$, that is the stratification column-set has some columns that are uniquely from the relation S. We will refer to these as missing stratcolumns.

Rule-D3d represents conditional equivalence under Property 1. Specifically, the missing stratcolumns are functionally dependent to some other columns in $\mathcal{D}_R - S_c$ (i.e. $\mathcal{D}_R - S_c \vdash D$); hence the expression on the right can simply ignore these missing columns. Further, the join columns should also be functionally dependent to the remaining strat columns (i.e. $\mathcal{D}_R - S_c \vdash \mathcal{C}$).

Rule-D3e represents conditional equivalence under Property 1 and Property 5. The expression on the right ignores stratification on the missing stratcolumns but raises required support per group. From Property 1, functional dependence is required, i.e. $\mathcal{D}_R - S_c \vdash \mathcal{D}$. Further, by Property 5 all the groups have support below $\frac{\delta}{\bowtie^{\mathtt{ss}}}$. In the general case, the two expressions are equivalent for groups with support below the threshold; for larger groups they are probabilistically equivalent (noted below as Rule-D3z).

Rule-D3f stratifies on the join columns, which are foreign keys, to substitute for the missing stratcolumns. By definition of foreign-key join, there is a many-one relationship between the values of join columns $\mathcal{C}$ and the missing stratcolumns $\mathcal{D}_R \cap S_c$. The proof follows from a similar reasoning to the case with Rule-D2 (i.e. $(\mathcal{D}_R - S_c) \cup \mathcal{C} \vdash \mathcal{D}$).

Rule-D3g states conditional equivalence when Property 5 holds; that is all groups have support below the threshold $\frac{\delta}{\mathtt{sfm}}$. In general, only probabilistic equivalence holds for the larger groups (noted below as Rule-D3w). Intuitively, the join columns being foreign keys and can have more distinct values than the columns that they are replacing in the stratification columnset. Hence, this rule lowers the number of tuples needed per distinct value. The proof follows from noting that $(\mathcal{D}_R - S_c) \cup \mathcal{C} \vdash \mathcal{D}$ and the support of groups is below the specified threshold. □

Conjecture 2 (Pushing past foreign-key Join)**.** *Given relations R and S being*

joined on columns $\mathcal{C}$ where $\mathcal{C}$ is a foreign-key in R and primary-key in S, let R_c and S_c denote the columns of R and S respectively and suppose $\mathcal{D}_R$ is a set generated from set $\mathcal{D}$ by replacing the columns in set $\mathcal{D}$ that belong to S_c with equivalent columns in R_c as per the equijoin conditions [2], we have:
Rule-D3x: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\rightarrow}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R,\frac{\delta}{\bowtie^{\mathtt{ss}}}}(R) \bowtie_{\mathcal{C}} S$, *if* $\mathcal{D}_R \subseteq R_c$ *where* $\bowtie^{\mathtt{ss}} = \frac{|R \bowtie_{\mathcal{C}} S|}{|R|}$, *is the selectivity of the join (additional conditions needed);*
Rule-D3y: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\rightarrow}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R,\delta}(R) \bowtie_{\mathcal{C}} S$, *if* $\mathcal{D}_R \subseteq R_c$ *(additional conditions needed);*
Rule-D3z: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\rightarrow}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,\mathcal{D}_R - S_c,\frac{\delta}{\bowtie^{\mathtt{ss}}}}(R) \bowtie_{\mathcal{C}} S$, *if* $\mathcal{D}_R \nsubseteq R_c$;
Rule-D3w: $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R \bowtie_{\mathcal{C}} S) \overset{\rightarrow}{\Leftrightarrow} \Gamma^{\mathsf{D}}_{p,(\mathcal{D}_R - S_c)\cup\mathcal{C},\frac{\delta}{\mathtt{sfm}}}(R) \bowtie_{\mathcal{C}} S$, *if* $\mathcal{D}_R \nsubseteq R_c$ *where* $\mathtt{sfm} = \frac{\mathtt{NumDV}(R,(\mathcal{D}_R - S_c)\cup\mathcal{C})}{\mathtt{NumDV}(R,\mathcal{D}_R)}$.

These equivalence statements are offered without proof.

Rule-D3x is the analog of Rule-D2x and holds under the same conditions (Condition 1). Convergence is faster when the equijoin is independent on the group columns (noted under Property 3).

Rule-D3y is the analog of Rule-D2y and holds under Condition 1 holds. Convergence is faster when Property 3 holds for the equijoin. This rule, in general, converges slower than Rule-D3x but leads to a smaller sample.

Rule-D3z is the probabilistic equivalent version of Rule-D3e. It holds under Condition 1; that is, error converges as the support on groups over $\mathcal{D}_R - S_c$ on R tends to ∞. Convergence is faster when Property 3 and/or Property 4 hold (independence between join columns and group-by columns). That is, a smaller sample can be taken for the same relation size when these properties hold.

Rule-D3w is the probabilistic equivalent version of Rule-D3g. It holds under Condition 1. Convergence is faster when Property 3 and/or Property 4 hold (independence between join columns and group-by columns). This rule is dominated by Rule-D3z and converges more slowly. However, it leads to a smaller sample size.

Proposition 12 (Pushing past Join)**.** *For relations R_1 and R_2, with columns $\mathcal{C}_i$ respectively and an equi-join $\bowtie_{\mathcal{C}}$ on columns $\mathcal{C}$, we have*
Rule-U4: [intentionally empty]
Rule-D4: [intentionally empty]
Rule-V4: $\Gamma^{\mathsf{V}}_{p,\mathcal{D}}(R_1 \bowtie_{\mathcal{C}} R_2) \overset{*}{\Leftrightarrow} \Gamma^{\mathsf{V}}_{p,\mathcal{C}}(R_1) \bowtie_{\mathcal{C}} \Gamma^{\mathsf{V}}_{p,\mathcal{C}}(R_2)$, *if* $\mathcal{C} = \mathcal{D}$.

Proof. The proof for Rule-V4 follows from the observation that the expression on the right picks exactly the same tuples as the expression on the right because

[2] Assume $\mathcal{D} = \{r_1, s_1, s_2\}$ and the join condition is $r_3 = s_1$ where columns r_i and s_i are from the relations R and S respectively. Then, $\mathcal{D}_R = \{r_1, r_3, s_2\}$.

QUICKR ensures that the universe samplers on either side pick the same portion of the hash-space. That is, this rule only holds if samplers on either side make the same random decisions. □

Discussion: Rule-U4 is intentionally empty; we know no (non-trivial) expression that pushes uniform samplers into one or both relations that v-dominates, c-dominates, or is weakly equivalent to $\Gamma^{\mathsf{U}}_{p}(R_1 \bowtie_{\mathcal{C}} R_2)$. In particular, pushing a uniform sampler to just one of the join inputs will introduce correlation between tuples: a tuple that joins with more than one tuple from the other input will either be picked or not picked. And hence all tuples in the join that derive from this tuple are picked in one-shot rather than independently. This leads to both a higher variance and to a higher likelihood of missing groups. Pushing uniform samplers to both sides has similar issues. We note that it is possible to equalize the probability of passing individual tuples, but the correlation between tuples is harder to relate to a single sampler that follows the join.

A careful read of [3] shows that it is possible to find some GUS (generalized uniform sampler) such that $\Gamma^{\mathsf{gus}}(R \bowtie_{\mathcal{C}} S) \overset{*}{\Leftrightarrow} \Gamma^{\mathsf{U}}_{p}(R) \bowtie_{\mathcal{C}} \Gamma^{\mathsf{U}}_{q}(S)$. However, as [3] shows computing the variance of the GUS sampler on the left is non-trivial (requires a self-join on samples). This restricts our ability to reason about accuracy of the plan at query optimization time. Hence, QUICKR does not use such pushdown rules.

Rule-D4 is also intentionally empty; we know no expression that pushes distinct samplers into one or both relations that v-dominates, c-dominates or is weakly equivalent to $\Gamma^{\mathsf{D}}_{p,\mathcal{D},\delta}(R_1 \bowtie_{\mathcal{C}} R_2)$. The reasoning is similar to the above with uniform sampler.

Let us explicitly consider the case of $\Gamma^{\mathsf{D}}_{p,\mathcal{C},\delta}(R_1) \bowtie_{\mathcal{C}} \Gamma^{\mathsf{D}}_{p,\mathcal{C},\delta}(R_2)$. That is, stratified sample both join inputs on the join cols. Analogous to the above reasoning with uniform sampler, sampling before the join introduces correlation between the tuples that would not be present were the distinct sampler executed after the join. Thus, this expression does not dominate an expression with a {uniform, distinct, or universe} sampler applied after the join. This is the key advantage with the universe sampler, there is no added error from pushing the sampler to before a join.

We note that even though the above expression neither v-dominates, c-dominates or is weakly equivalent to a join-then-sample, it will ensure that groups will not be missed if the group-by columns are identical to the join columns.

Finally, we note that it is possible to do a more intricate *joint sampling* of multiple inputs by relying on indices on join columns, exchanging frequency histograms on the join columns between the two sides etc. QUICKR intentionally steers away from such samplers because of their limited applicability and difficulty to implement in the wild. For example, indices are rarely available on intermediate content as would be generated in a nested SQL statement (roughly half of TPC-H queries are not a single SQL statement). As another example, computing frequency histograms and exchanging them is a barrier in the parallel execution and such a sampler cannot be pushed past further joins.

Note that QUICKR uses $\Gamma^{\mathsf{V}}_{p,\mathcal{C}}(R_1) \bowtie_{\mathcal{C}} \Gamma^{\mathsf{V}}_{p,\mathcal{C}}(R_2)$ in place of $\Gamma^{\mathsf{U}}_{p}(R_1 \bowtie_{\mathcal{C}} R_2)$ even though the former has a somewhat higher variance and likelihood of missing groups. Empirically, error is small due to two reasons (1) it is common to have many distinct values for $\mathcal{C}$ and (2) QUICKR uses cryptographically random hash functions which avoids collisions.

5.5 Semijoins

Left- and right- semijoins are, for the purposes of pushing samplers, similar to the case of foreign key joins. The reason is that at most one row in the answer appears per row of the *fact* relation. Moreover the schema of join output matches that of the *fact* relation simplifying checks for existence of stratification and universe columns on that relation.

5.6 Other operations (Outerjoins, Union-All, UDOs, Other joins, ...)

We defer further details of pushing samplers past other operations to future work.

References

[1] M. Charikar, S. Chaudhuri, R. Motwani, and V. Narasayya. Towards estimation error guarantees for distinct values. In *POD*, 2000.

[2] S. Kandula, A. Shanbhag, A. Vitorovic, M. Olma, R. Grandl, S. Chaudhuri, and B. Ding. Quickr: Lazily approximating complex adhoc queries in bigdata clusters. In *SIGMOD*, 2016.

[3] S. Nirkhiwale, A. Dobra, and C. Jermaine. A sampling algebra for aggregate estimation. In *PVLDB*, 2013.