# Fine-grained Provenance for Linear Algebra Operators

Zhepeng Yan Val Tannen Zachary Ives

University of Pennsylvania

# Motivation

- Provenance is well-understood for relational data / queries.
  - E.g., view maintenance, delete propagation, computing trust, prob. db
- But increasingly analysts are performing more complex tasks:
  - Machine learning, data mining, image analysis, graph analytics
- Array data and matrix algebra are commonly used!

# Motivation

- Provenance is well-understood for relational data / queries.
  - E.g., view maintenance, delete propagation, computing trust, prob. db
- But increasingly analysts are performing more complex tasks:
  - Machine learning, data mining, image analysis, graph analytics
- Array data and matrix algebra are commonly used!
- Question: How do we track provenance in this setting?

# Inspiration: Provenance Semirings

- An algebra framework [Green et al. PODS'07] for
  - annotating tuples in a relation
  - propagating annotations through relational queries (SPJU and aggregation)
- Enables efficient delete propagation, view maintenance, etc

# Semiring example: input data

## P

| PID | PValue |
|---|---|
| 101 | 11 |
| 102 | 12 |
| 2003 | 13 |
| 2004 | 14 |
| 2005 | 15 |
| 2006 | 16 |

## Q

| QID | QValue |
|---|---|
| 2003 | 0 |
| 2004 | 0 |
| 2005 | 0 |
| 2006 | 0 |

## R

| RID | RValue |
|---|---|
| 101 | 13 |
| 102 | 14 |
| 5005 | 15 |
| 5006 | 16 |
| 5007 | 17 |
| 5008 | 18 |
| 5009 | 19 |

# Semiring example: query

## P

| PID | PValue |
|---|---|
| 101 | 11 |
| 102 | 12 |
| 2003 | 13 |
| 2004 | 14 |
| 2005 | 15 |
| 2006 | 16 |

## Q

| QID | QValue |
|---|---|
| 2003 | 0 |
| 2004 | 0 |
| 2005 | 0 |
| 2006 | 0 |

## R

| RID | RValue |
|---|---|
| 101 | 13 |
| 102 | 14 |
| 5005 | 15 |
| 5006 | 16 |
| 5007 | 17 |
| 5008 | 18 |
| 5009 | 19 |

**S(Value) :- P(x, Value), R(x, _)**
**S(Value) :- P(x, Value), Q(x, _)**
**S(Value) :- R(_, Value)**

# Semiring example: query

## P

| PID | PValue |
|---|---|
| 101 | 11 |
| 102 | 12 |
| 2003 | 13 |
| 2004 | 14 |
| 2005 | 15 |
| 2006 | 16 |

## Q

| QID | QValue |
|---|---|
| 2003 | 0 |
| 2004 | 0 |
| 2005 | 0 |
| 2006 | 0 |

## R

| RID | RValue |
|---|---|
| 101 | 13 |
| 102 | 14 |
| 5005 | 15 |
| 5006 | 16 |
| 5007 | 17 |
| 5008 | 18 |
| 5009 | 19 |

**S(Value) :- P(x, Value), R(x, _)**

S(Value) :- P(x, Value), Q(x, _)

S(Value) :- R(_, Value)

# Semiring example: query

## P

| PID | PValue |
| --- | --- |
| 101 | 11 |
| 102 | 12 |
| 2003 | 13 |
| 2004 | 14 |
| 2005 | 15 |
| 2006 | 16 |

## Q

| QID | QValue |
| --- | --- |
| 2003 | 0 |
| 2004 | 0 |
| 2005 | 0 |
| 2006 | 0 |

## R

| RID | RValue |
| --- | --- |
| 101 | 13 |
| 102 | 14 |
| 5005 | 15 |
| 5006 | 16 |
| 5007 | 17 |
| 5008 | 18 |
| 5009 | 19 |

**S(Value) :- P(x, Value), R(x, _)**

**S(Value) :- P(x, Value), Q(x, _)**

**S(Value) :- R(_, Value)**

# Semiring example: query

## P

| PID | PValue |
|---|---|
| 101 | 11 |
| 102 | 12 |
| 2003 | 13 |
| 2004 | 14 |
| 2005 | 15 |
| 2006 | 16 |

## Q

| QID | QValue |
|---|---|
| 2003 | 0 |
| 2004 | 0 |
| 2005 | 0 |
| 2006 | 0 |

## R

| RID | RValue |
|---|---|
| 101 | 13 |
| 102 | 14 |
| 5005 | 15 |
| 5006 | 16 |
| 5007 | 17 |
| 5008 | 18 |
| 5009 | 19 |

**S(Value) :- P(x, Value), R(x, _)**

**S(Value) :- P(x, Value), Q(x, _)**

**S(Value) :- R(_, Value)**

# Semiring example: output tuples

## S

| Value |
|---|
| 11 |
| 12 |
| 13 |
| 14 |
| 15 |
| 16 |
| 17 |
| 18 |
| 19 |

**S(Value) :- P(x, Value), R(x, _)**
**S(Value) :- P(x, Value), Q(x, _)**
**S(Value) :- R(_, Value)**

# Semiring example: annotated output

**S**

| Value | Annotation |
|---|---|
| 11 | $pr$ |
| 12 | $pr$ |
| 13 | $pq + r$ |
| 14 | $pq + r$ |
| 15 | $pq + r$ |
| 16 | $pq + r$ |
| 17 | $r$ |
| 18 | $r$ |
| 19 | $r$ |

**S(Value) :- P(x, Value), R(x, _)**
**S(Value) :- P(x, Value), Q(x, _)**
**S(Value) :- R(_, Value)**

# Semiring example: delete propagation

## S

| Value | Annotation |
|---|---|
| 11 | $pr$ |
| 12 | $pr$ |
| 13 | $pq + r$ |
| 14 | $pq + r$ |
| 15 | $pq + r$ |
| 16 | $pq + r$ |
| 17 | $r$ |
| 18 | $r$ |
| 19 | $r$ |

What if we remove tuples from **P**?

Set $p = 0$!

# Semiring example: delete propagation

## S

| Value | Annotation |
|---|---|
| 11 | *0* |
| 12 | *0* |
| 13 | *r* |
| 14 | *r* |
| 15 | *r* |
| 16 | *r* |
| 17 | *r* |
| 18 | *r* |
| 19 | *r* |

What if we remove tuples from **P**?

Set $p = 0$!

# Semimodule example: aggregation

## S

**Query: SUM(Value)**

| Value | Annotation |
|---|---|
| 11 | $pr$ |
| 12 | $pr$ |
| 13 | $pq + r$ |
| 14 | $pq + r$ |
| 15 | $pq + r$ |
| 16 | $pq + r$ |
| 17 | $r$ |
| 18 | $r$ |
| 19 | $r$ |

# Semimodule example: annotation

## S

| Value | Annotation |
|---|---|
| 11 | $pr$ |
| 12 | $pr$ |
| 13 | $pq + r$ |
| 14 | $pq + r$ |
| 15 | $pq + r$ |
| 16 | $pq + r$ |
| 17 | $r$ |
| 18 | $r$ |
| 19 | $r$ |

**Query: SUM(Value)**

**Annotated aggregation**

$pr * (11 + 12) + (pq + r) * (13 + 14 + 15 + 16) + r * (17 + 18 + 19)$

# Semimodule example: annotation

## S

| Value | Annotation |
|---|---|
| 11 | *pr* |
| 12 | *pr* |
| 13 | *pq + r* |
| 14 | *pq + r* |
| 15 | *pq + r* |
| 16 | *pq + r* |
| 17 | *r* |
| 18 | *r* |
| 19 | *r* |

**Query: SUM(Value)**

**Annotated aggregation**

$\mathbf{pr} * (11 + 12) + \mathbf{(pq + r)} * (13 + 14 + 15 + 16) + \mathbf{r} * (17 + 18 + 19)$

**First term: annotation**

# Semimodule example: annotation

## S

| Value | Annotation |
|---|---|
| 11 | *pr* |
| 12 | *pr* |
| 13 | *pq + r* |
| 14 | *pq + r* |
| 15 | *pq + r* |
| 16 | *pq + r* |
| 17 | *r* |
| 18 | *r* |
| 19 | *r* |

**Query: SUM(Value)**

**Annotated aggregation**

$pr * \mathbf{(11 + 12)} + (pq + r) * \mathbf{(13 + 14 + 15 + 16)} + r * \mathbf{(17 + 18 + 19)}$

First term: annotation

**Second term: value**

# Semimodule example: annotation

## S

| Value | Annotation |
|---|---|
| 11 | *pr* |
| 12 | *pr* |
| 13 | *pq + r* |
| 14 | *pq + r* |
| 15 | *pq + r* |
| 16 | *pq + r* |
| 17 | *r* |
| 18 | *r* |
| 19 | *r* |

**Query: SUM(Value)**

**Annotated aggregation**

$pr * (11 + 12) + (pq + r) * (13 + 14 + 15 + 16) + r * (17 + 18 + 19)$

or $pr * 23 + (pq + r) * 58 + r * 54$

# Semimodule example: annotation

## S

| Value | Annotation |
|---|---|
| 11 | *pr* |
| 12 | *pr* |
| 13 | *pq + r* |
| 14 | *pq + r* |
| 15 | *pq + r* |
| 16 | *pq + r* |
| 17 | *r* |
| 18 | *r* |
| 19 | *r* |

**Query: SUM(Value)**

**Annotated aggregation**

$pr * (11 + 12) + (pq + r) * (13 + 14 + 15 + 16) + r * (17 + 18 + 19)$

or $pr * 23 + (pq + r) * 58 + r * 54$

or $pq * 58 + r * (p * 23 + 112)$

# Semimodule example: annotation

## S

| Value | Annotation |
|---|---|
| 11 | $pr$ |
| 12 | $pr$ |
| 13 | $pq + r$ |
| 14 | $pq + r$ |
| 15 | $pq + r$ |
| 16 | $pq + r$ |
| 17 | $r$ |
| 18 | $r$ |
| 19 | $r$ |

**Query: SUM(Value)**

**Annotated aggregation**

$pr * (11 + 12) + (pq + r) * (13 + 14 + 15 + 16) + r * (17 + 18 + 19)$

or $pr * 23 + (pq + r) * 58 + r * 54$

or $pq * 58 + r * (p * 23 + 112)$

Delete propagation:
set $r = 0$ and obtain $pq * 58$

# Tracking matrix provenance

- We want to get the same benefits!
  - Algebraically manipulate annotated matrices
  - Hypothetical deletion

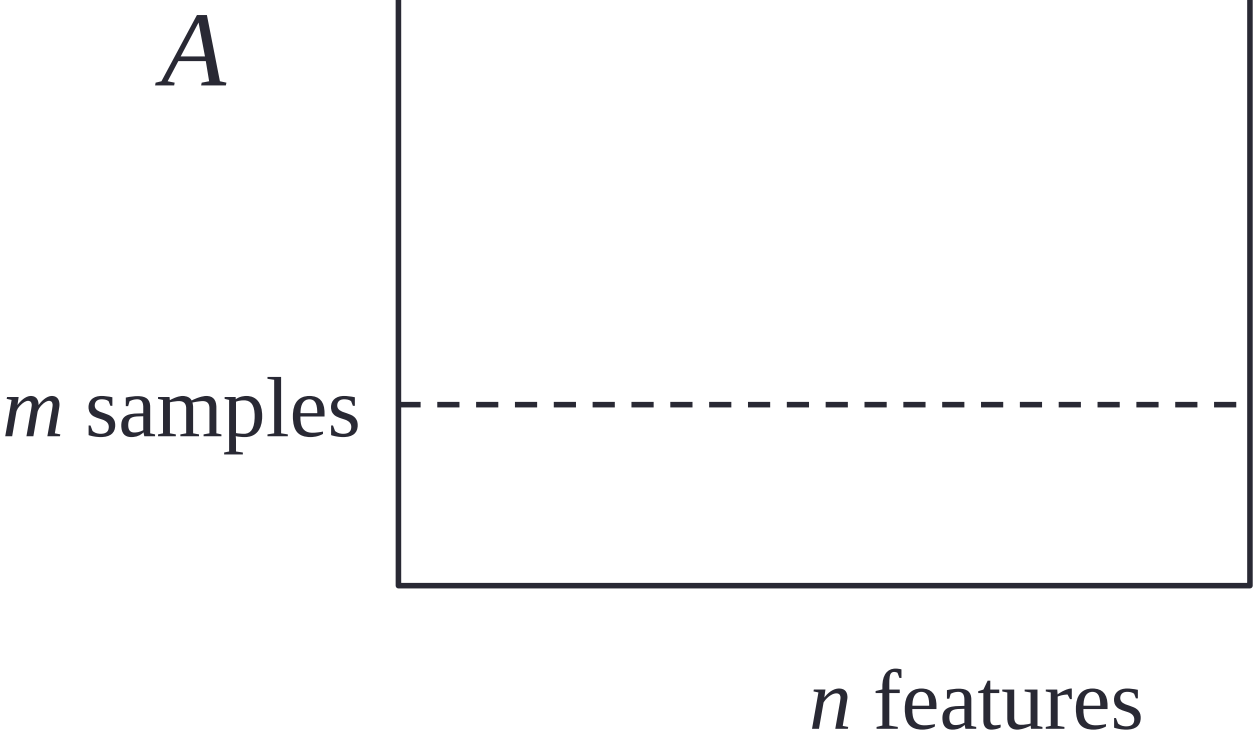
# Tracking matrix data

$A$

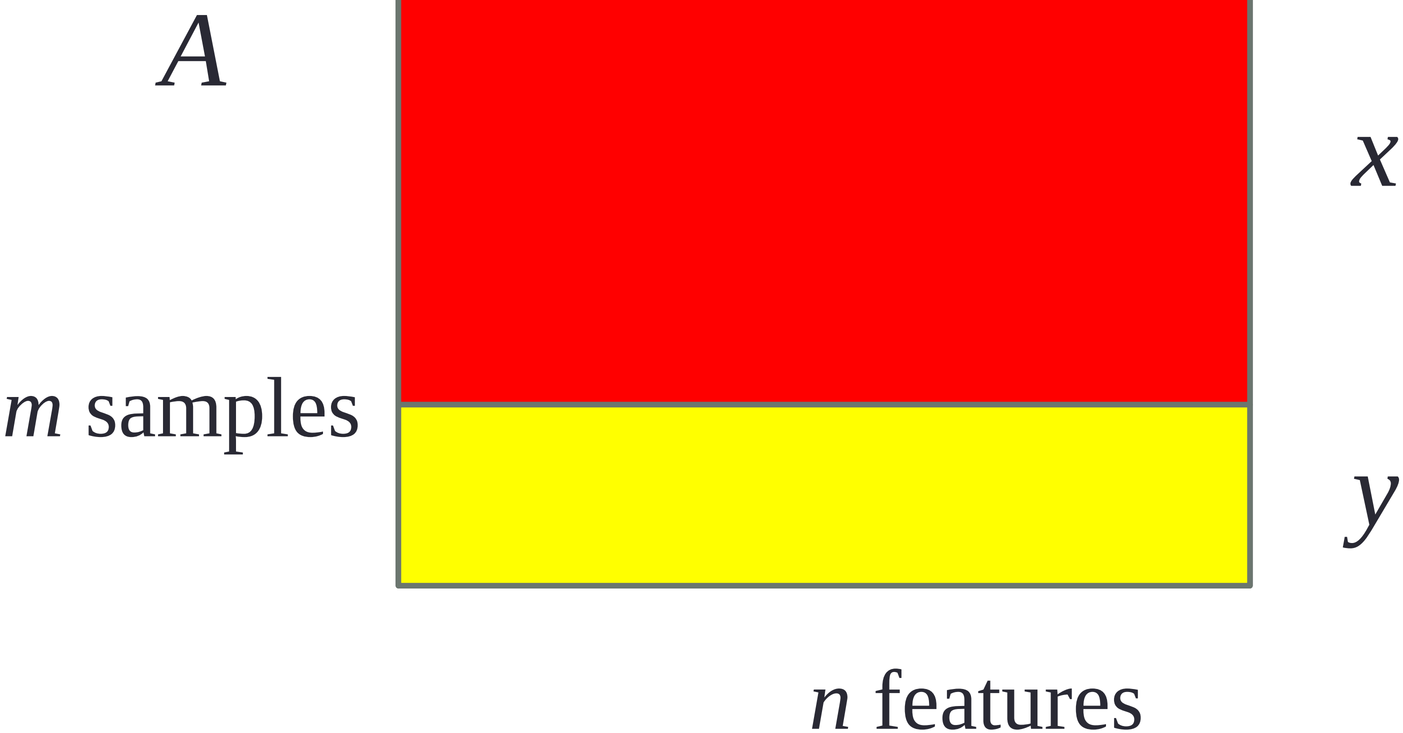
$m$ samples

$n$ features

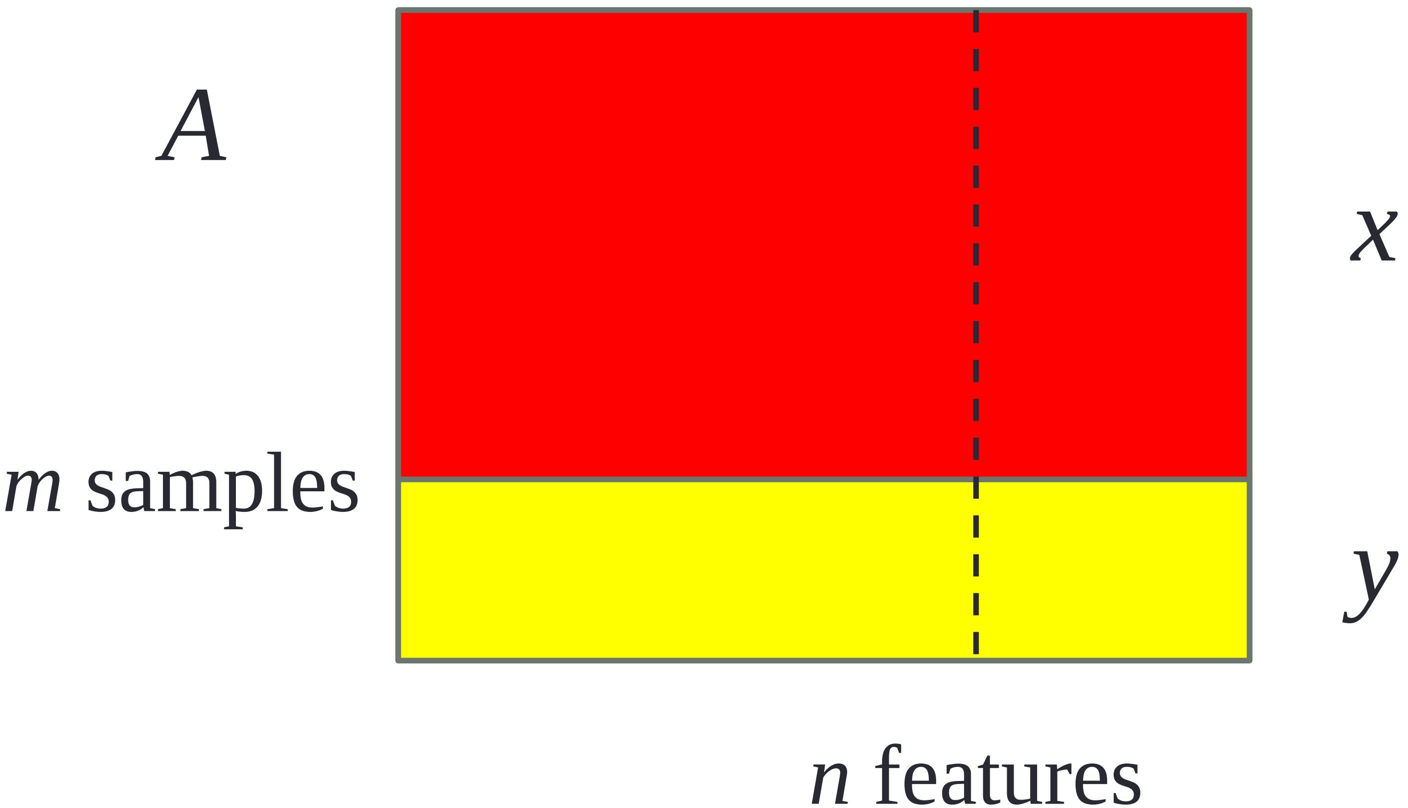
# Tracking matrix data: partitioning



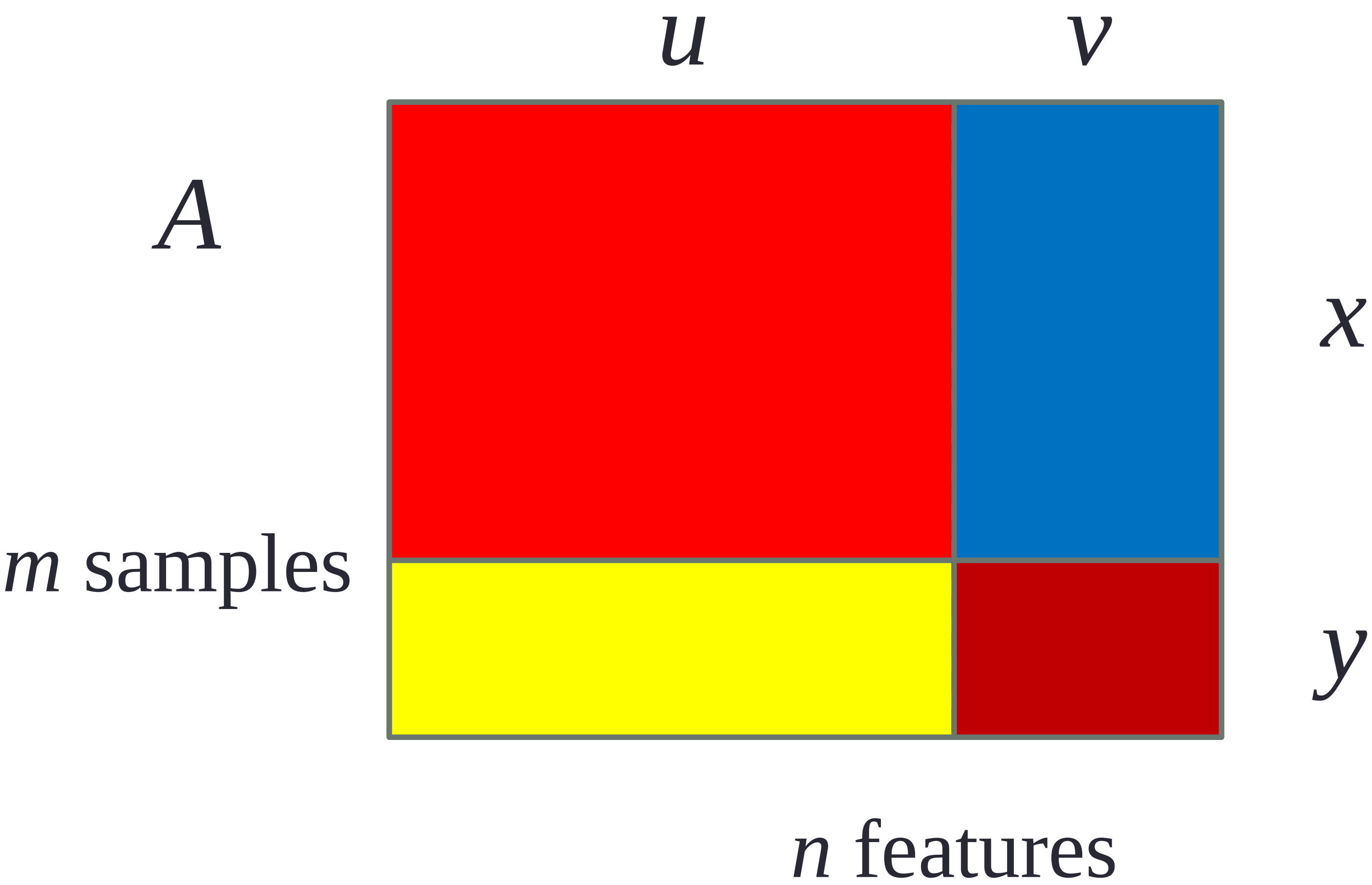
# Tracking matrix data: annotating



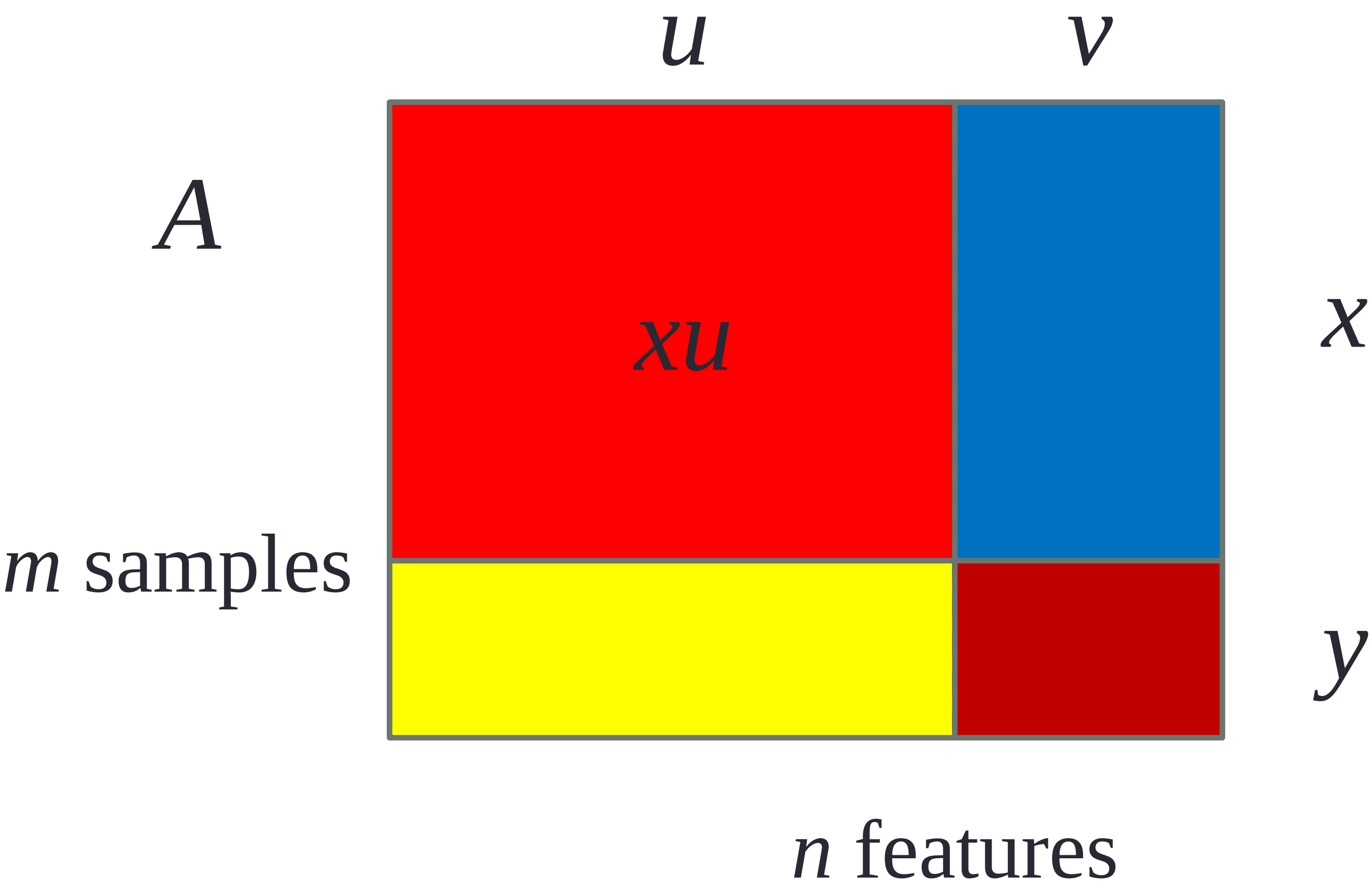
# Tracking matrix data: annotating



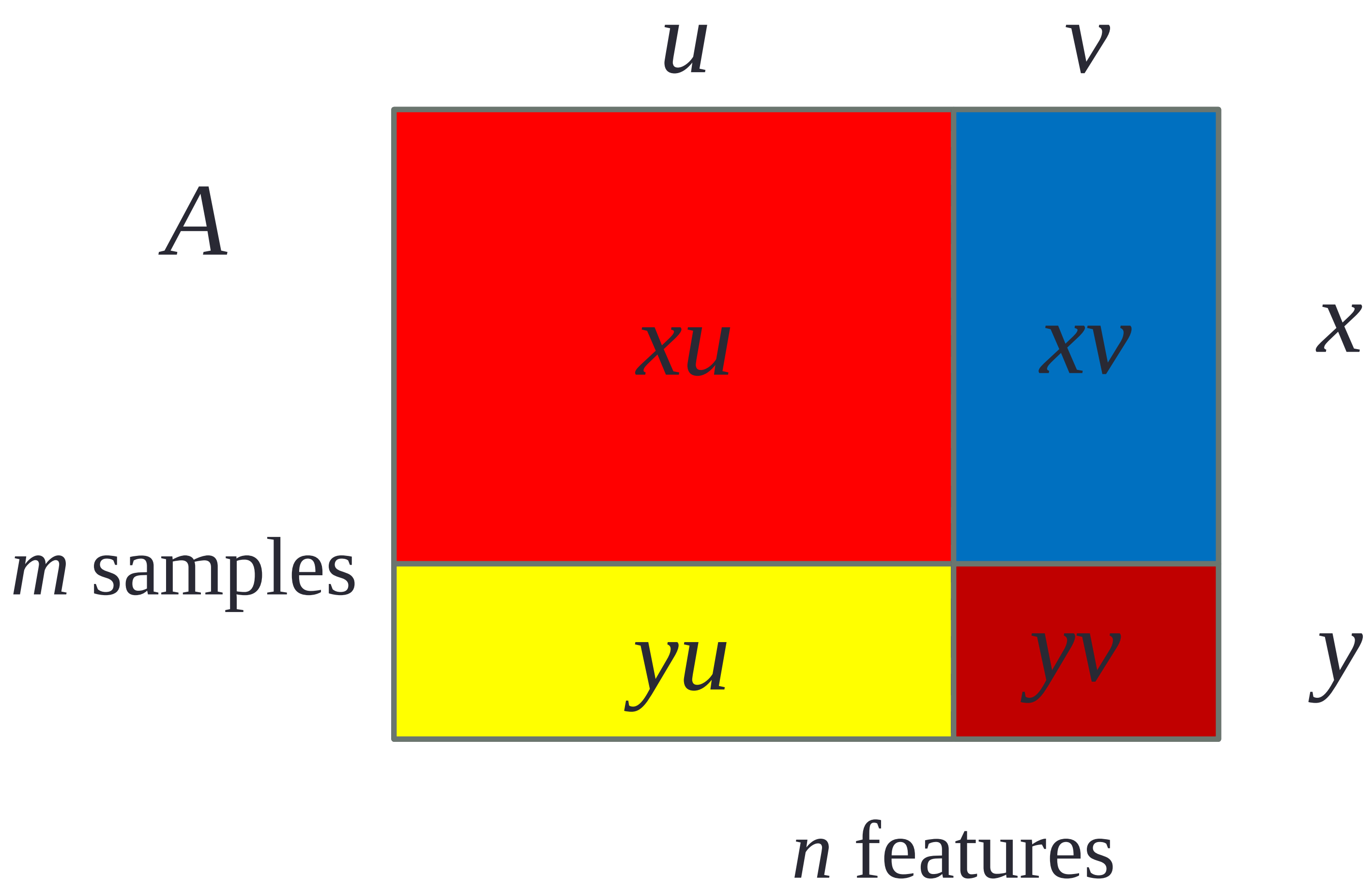
# Tracking matrix data: annotating



# Tracking matrix data: annotating



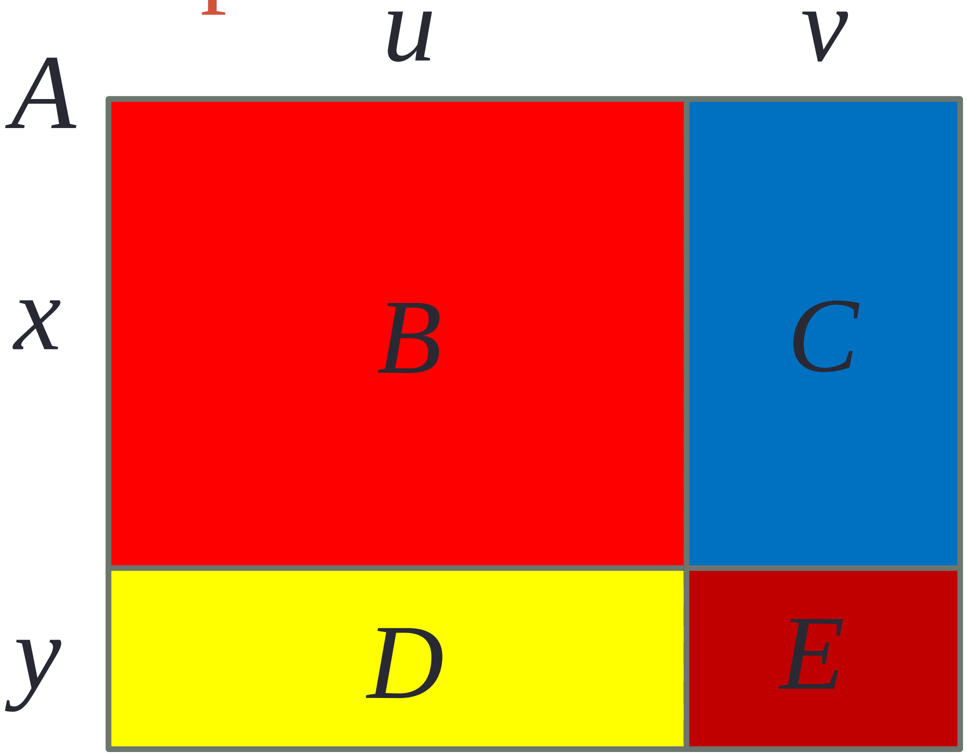
# Tracking matrix data: annotating



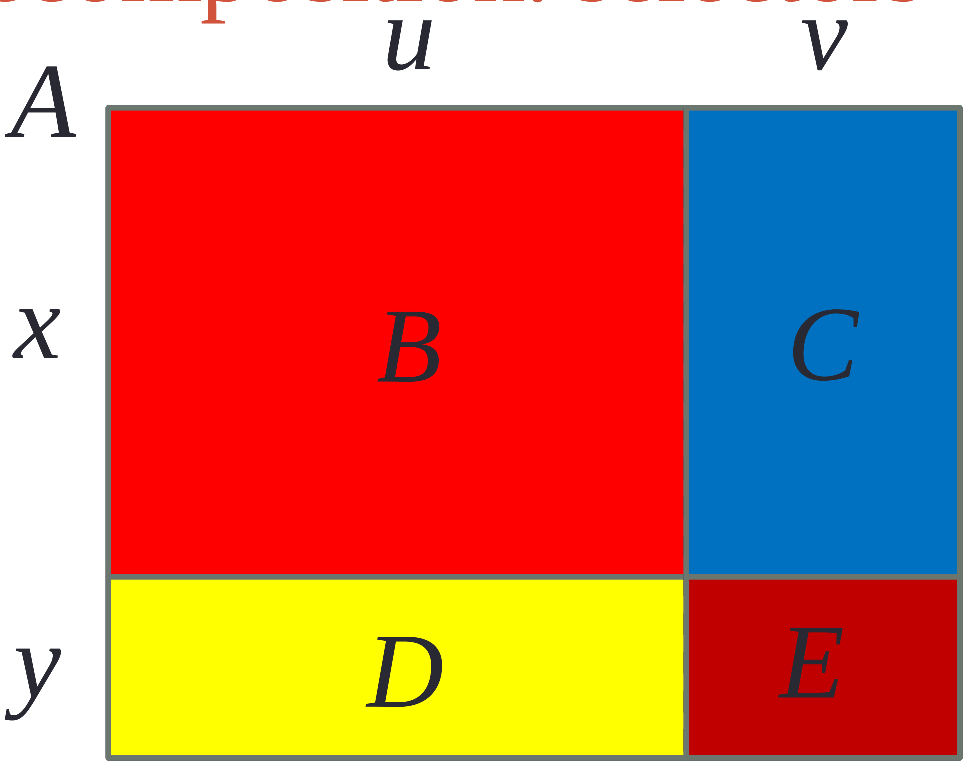
# Challenges

- Specify and relate different parts of a given matrix
  - Matrix decomposition through selector matrices
- Specify connection between derived and source matrices
  - Embed matrix algebra and provenance into a semialgebra

# Decomposition



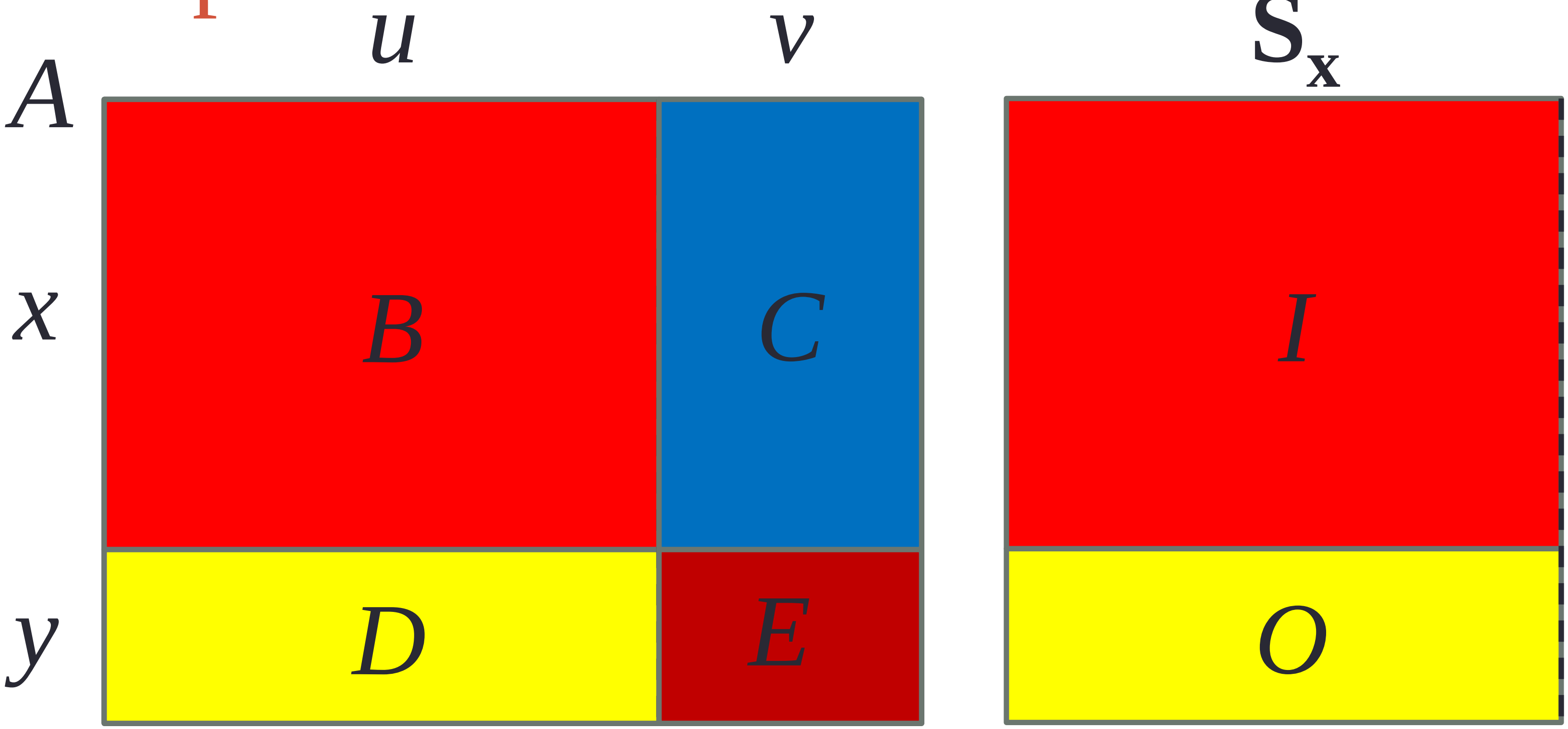
# Decomposition: selectors




$$A = \mathbf{S_x}\, B\, \mathbf{T_u} + \mathbf{S_x}\, C\, \mathbf{T_v} + \mathbf{S_y}\, D\, \mathbf{T_u} + \mathbf{S_y}\, E\, \mathbf{T_v}$$

with selectors $\mathbf{S_x}$ $\mathbf{S_y}$ $\mathbf{T_u}$ $\mathbf{T_v}$

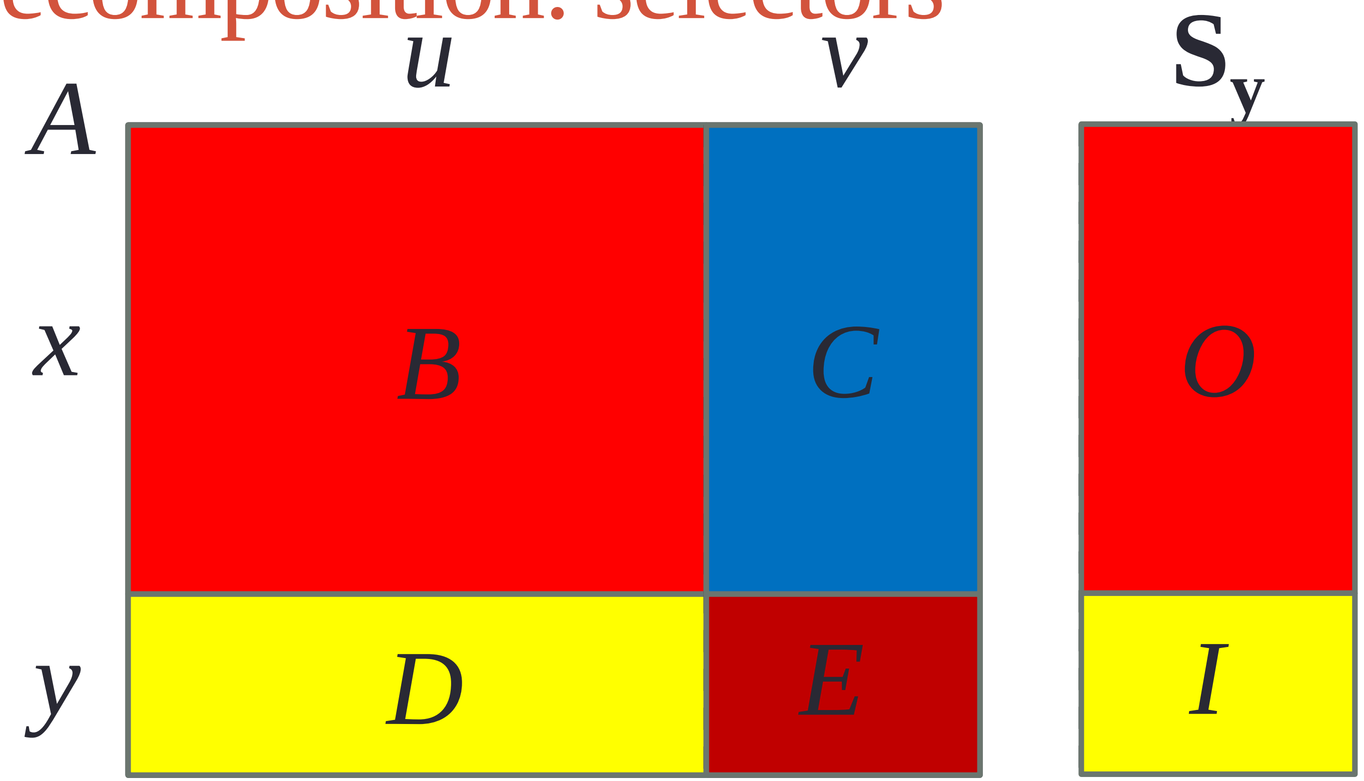
# Decomposition: selectors




$A = \mathbf{S_x}\, {\color{red}B}\, \mathbf{T_u} + \mathbf{S_x}\, {\color{blue}C}\, \mathbf{T_v} + \mathbf{S_y}\, {\color{orange}D}\, \mathbf{T_u} + \mathbf{S_y}\, {\color{darkred}E}\, \mathbf{T_v}$

with selectors $\mathbf{S_x}$ $\mathbf{S_y}$ $\mathbf{T_u}$ $\mathbf{T_v}$

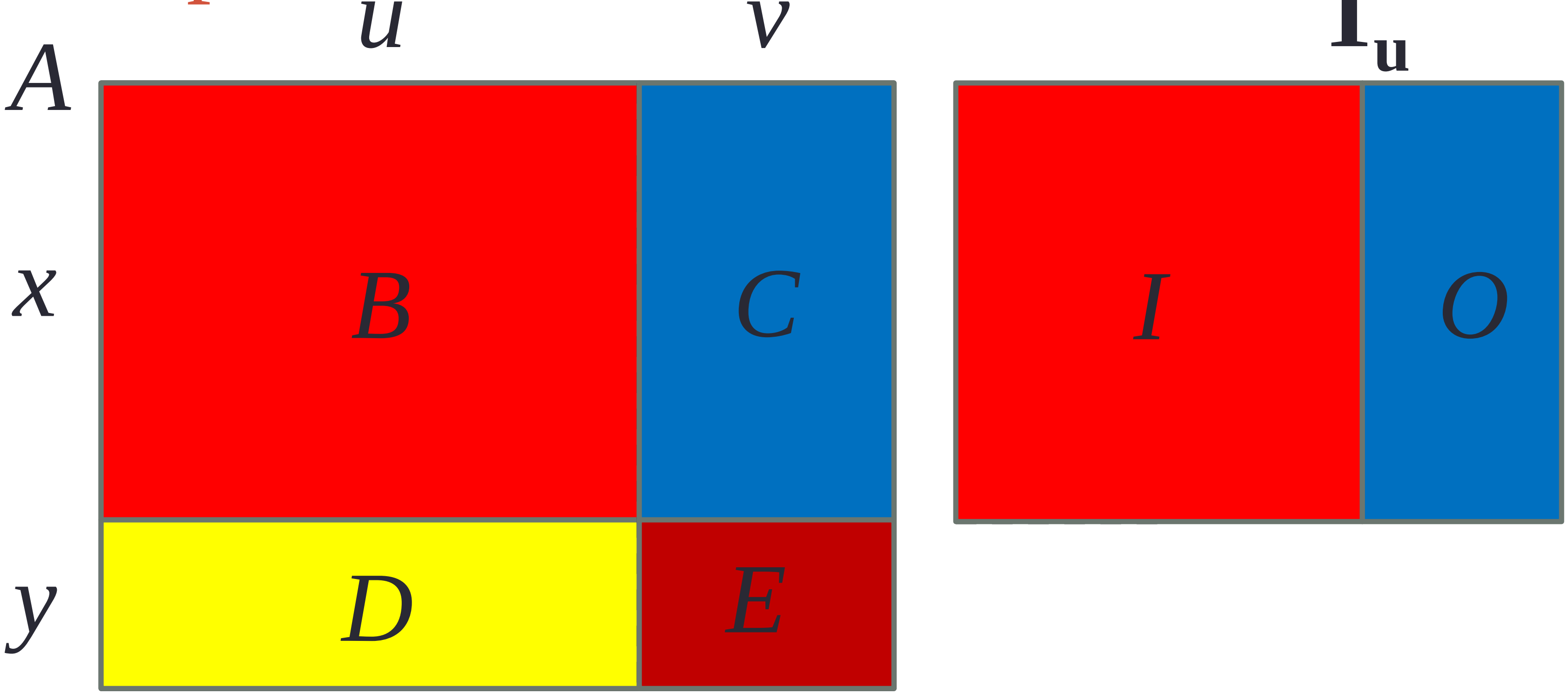
# Decomposition: selectors




$A = \mathbf{S_x}\, B\, \mathbf{T_u} + \mathbf{S_x}\, C\, \mathbf{T_v} + \mathbf{S_y}\, D\, \mathbf{T_u} + \mathbf{S_y}\, E\, \mathbf{T_v}$

with selectors $\mathbf{S_x}\ \mathbf{S_y}\ \mathbf{T_u}\ \mathbf{T_v}$

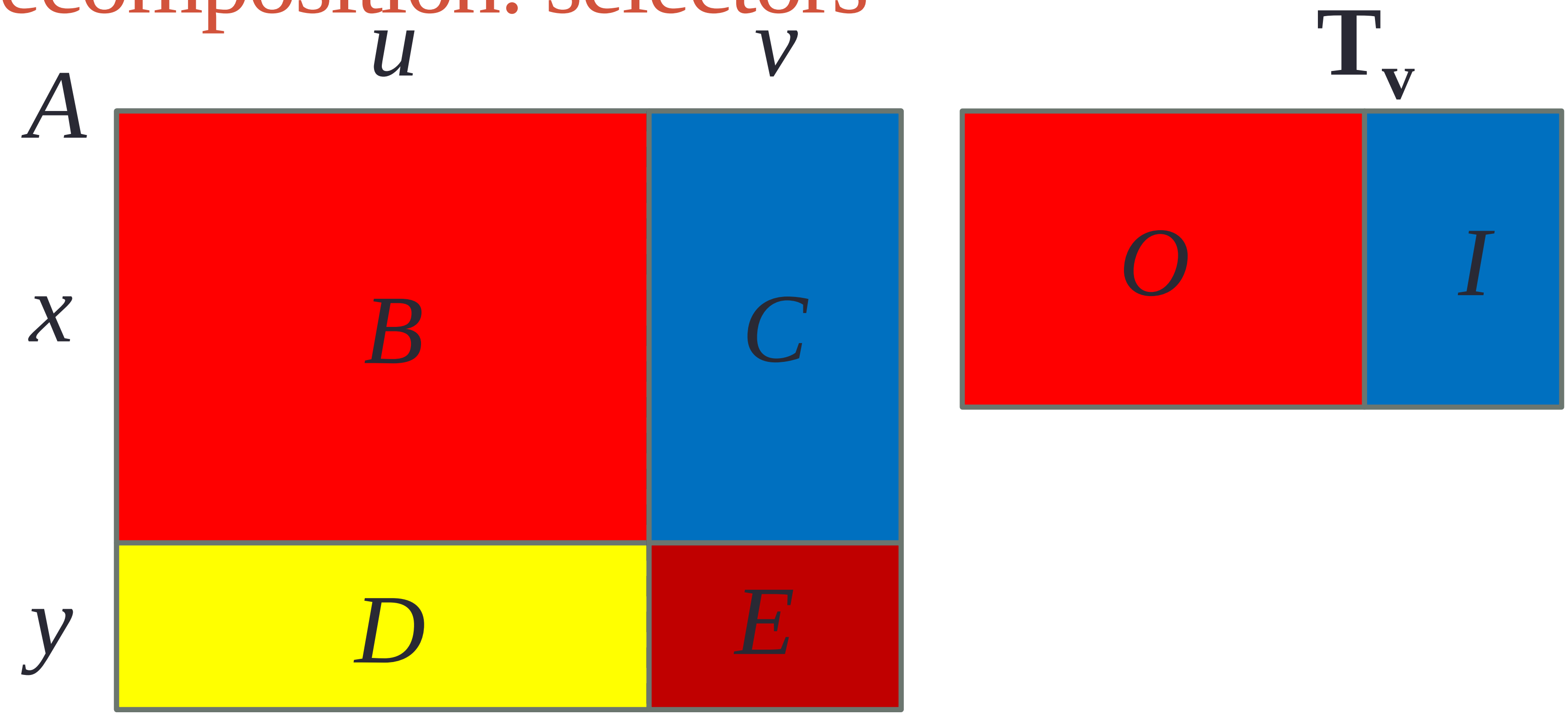
# Decomposition: selectors




$A = \mathbf{S}_x\, B\, \mathbf{T}_u + \mathbf{S}_x\, C\, \mathbf{T}_v + \mathbf{S}_y\, D\, \mathbf{T}_u + \mathbf{S}_y\, E\, \mathbf{T}_v$

with selectors $\mathbf{S}_x\ \mathbf{S}_y\ \mathbf{T}_u\ \mathbf{T}_v$

# Decomposition: selectors




$A = \mathbf{S_x}\, B\, \mathbf{T_u} + \mathbf{S_x}\, C\, \mathbf{T_v} + \mathbf{S_y}\, D\, \mathbf{T_u} + \mathbf{S_y}\, E\, \mathbf{T_v}$

with selectors $\mathbf{S_x}$ $\mathbf{S_y}$ $\mathbf{T_u}$ $\mathbf{T_v}$

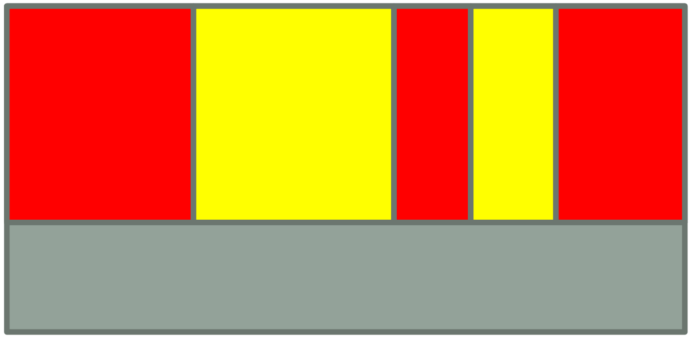
# Summary: selectors

- Relate a matrix to its sub-matrices.
- Matrices with only 0/1.
- Any row / column has at most a 1.

# Summary: selectors

- Relate a matrix to its sub-matrices.
- Matrices with only 0/1.
- Any row / column has at most a 1.

- Extends to non-adjacent case.
- Works for any rectangular partition.



# Provenance propagation

- We have
  - Matrices and operators over them – Algebra of matrices $\mathcal{M}$
  - Annotations – Semiring of provenance polynomials $\mathbb{N}[X]$
- Goals
  - Combine annotations in the same structure as the matrices
  - Operations should propagate data and annotations

# Provenance propagation

- We have
  - Matrices and operators over them – Algebra of matrices $\mathcal{M}$
  - Annotations – Semiring of provenance polynomials $\mathbb{N}[X]$
- Goals
  - Combine annotations in the same structure as the matrices
  - Operations should propagate data and annotations
- We do this in the space of tensor product $\mathbb{N}[X] \otimes \mathcal{M}$
  - Matrices as vectors, provenance as scalars: $p * A$
  - Satisfies all the laws of a $\mathbb{N}[X]$-**semialgebra**

# Laws of a N[*X*]-semialgebra (*K*-semialgebra)

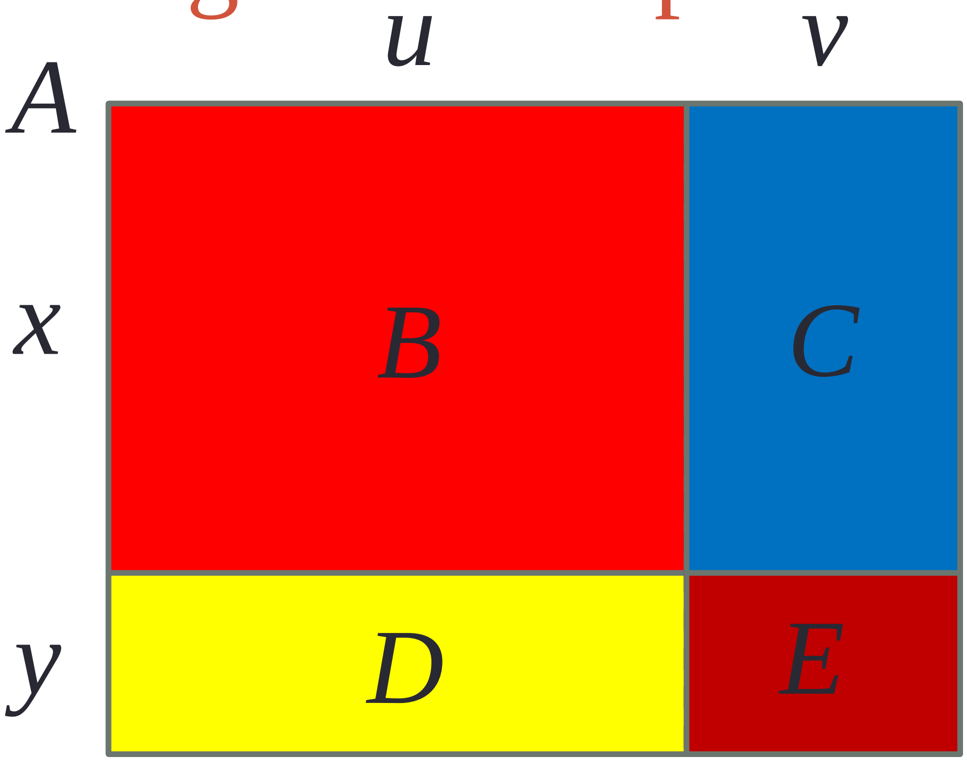
$(K, +_K, \cdot_K, 0_K, 1_K)$ commutative semiring

$(K \otimes \mathcal{M}, +, \cdot, 0, \mathsf{I})$ forms a ring (just like the matrices)

laws for scalar multiplication in a $K$-semialgebra

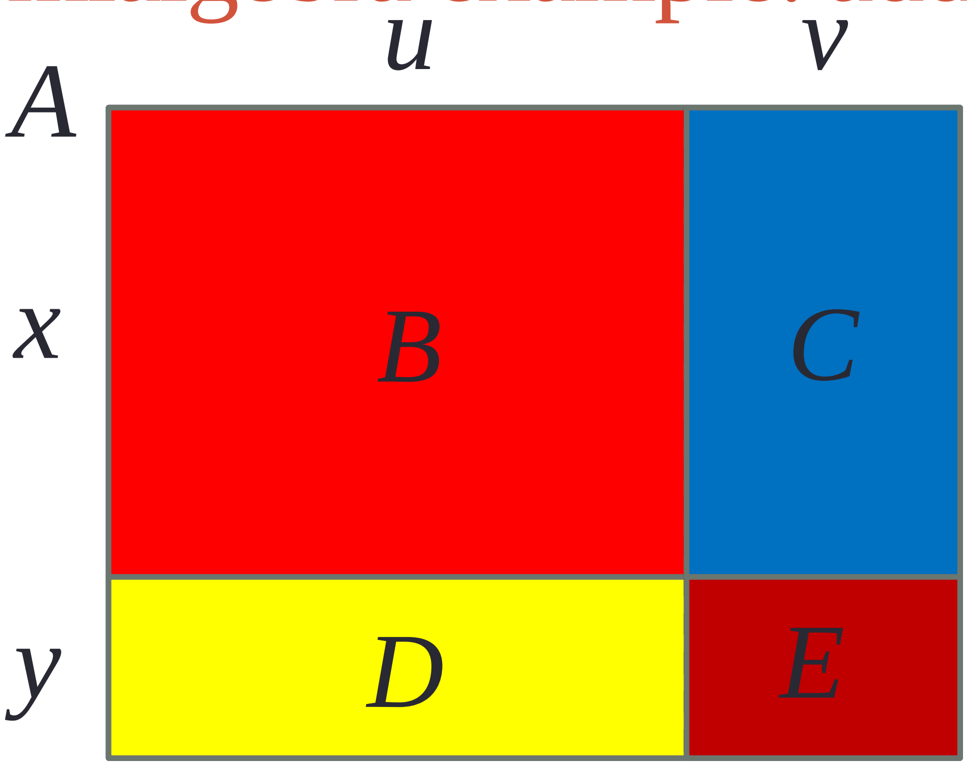
$$
\begin{aligned}
k * (A_1 + A_2) &= k * A_1 + k * A_2 \\
k * 0 &= 0 \\
(k_1 +_K k_2) * A &= k_1 * A + k_2 * A \\
0_K * A &= 0 \\
(k_1 \cdot_K k_2) * A &= k_1 * (k_2 * A) \\
1_K * A &= A \\
(k_1 * A_1)(k_2 * A_2) &= (k_1 \cdot_K k_2) * (A_1 A_2)
\end{aligned}
$$

# Semialgebra example




$$A = \mathbf{S_x}\, B\, \mathbf{T_u} + \mathbf{S_x}\, C\, \mathbf{T_v} + \mathbf{S_y}\, D\, \mathbf{T_u} + \mathbf{S_y}\, E\, \mathbf{T_v}$$

# Semialgebra example: add annotation




$$A = \mathbf{S}_{\mathbf{x}}\, xu * B\, \mathbf{T}_{\mathbf{u}} + \mathbf{S}_{\mathbf{x}}\, xv * C\, \mathbf{T}_{\mathbf{v}}$$
$$+ \mathbf{S}_{\mathbf{y}}\, yu * D\, \mathbf{T}_{\mathbf{u}} + \mathbf{S}_{\mathbf{y}}\, vy * E\, \mathbf{T}_{\mathbf{v}}$$

# Semialgebra example: propagate annotation

$$A = \mathbf{S}_{\mathbf{x}}\, xu{*}B\, \mathbf{T}_{\mathbf{u}} + \mathbf{S}_{\mathbf{x}}\, xv{*}C\, \mathbf{T}_{\mathbf{v}}$$
$$+ \mathbf{S}_{\mathbf{y}}\, yu{*}D\, \mathbf{T}_{\mathbf{u}} + \mathbf{S}_{\mathbf{y}}\, vy{*}E\, \mathbf{T}_{\mathbf{v}}$$

# Propagating annotation: transposition

$$A = \mathbf{S_x}\, xu{*}B\, \mathbf{T_u} + \mathbf{S_x}\, xv{*}C\, \mathbf{T_v}$$
$$+ \mathbf{S_y}\, yu{*}D\, \mathbf{T_u} + \mathbf{S_y}\, vy{*}E\, \mathbf{T_v}$$

$$\begin{aligned} A^T &= \mathsf{T}_u^T\,(xu * B^T)\,\mathsf{S}_x^T + \mathsf{T}_u^T\,(yu * D^T)\,\mathsf{S}_y^T \\ &+ \mathsf{T}_v^T\,(xv * C^T)\,\mathsf{S}_x^T + \mathsf{T}_v^T\,(yv * E^T)\,\mathsf{S}_y^T \end{aligned}$$

# Propagating annotation: transposition

$$A = \mathbf{S_x}\ xu{*}B\ \mathbf{T_u} + \mathbf{S_x}\ xv{*}C\ \mathbf{T_v} + \mathbf{S_y}\ yu{*}D\ \mathbf{T_u} + \mathbf{S_y}\ vy{*}E\ \mathbf{T_v}$$

$$\begin{aligned} A^T &= \mathsf{T}_u^T\,(xu * B^T)\,\mathsf{S}_x^T + \mathsf{T}_u^T\,(yu * D^T)\,\mathsf{S}_y^T \\ &+ \mathsf{T}_v^T\,(xv * C^T)\,\mathsf{S}_x^T + \mathsf{T}_v^T\,(yv * E^T)\,\mathsf{S}_y^T \end{aligned}$$

Transposition of a selector is still a selector

Still a sum of (selector × matrix × selector)

# Propagating annotation: multiplication

$$A = \mathbf{S}_{\mathbf{x}}\, xu*B\, \mathbf{T}_{\mathbf{u}} + \mathbf{S}_{\mathbf{x}}\, xv*C\, \mathbf{T}_{\mathbf{v}} + \mathbf{S}_{\mathbf{y}}\, yu*D\, \mathbf{T}_{\mathbf{u}} + \mathbf{S}_{\mathbf{y}}\, vy*E\, \mathbf{T}_{\mathbf{v}}$$

$$\begin{aligned} A^T &= \mathsf{T}_u^T\,(xu * B^T)\,\mathsf{S}_x^T + \mathsf{T}_u^T\,(yu * D^T)\,\mathsf{S}_y^T \\ &+ \mathsf{T}_v^T\,(xv * C^T)\,\mathsf{S}_x^T + \mathsf{T}_v^T\,(yv * E^T)\,\mathsf{S}_y^T \end{aligned}$$

# Propagating annotation: multiplication

$$A = \mathbf{S_x}\, xu{*}B\, \mathbf{T_u} + \mathbf{S_x}\, xv{*}C\, \mathbf{T_v} + \mathbf{S_y}\, yu{*}D\, \mathbf{T_u} + \mathbf{S_y}\, vy{*}E\, \mathbf{T_v}$$

$$\begin{aligned} A^T &= \mathsf{T}_u^T\,(xu * B^T)\,\mathsf{S}_x^T + \mathsf{T}_u^T\,(yu * D^T)\,\mathsf{S}_y^T \\ &+ \mathsf{T}_v^T\,(xv * C^T)\,\mathsf{S}_x^T + \mathsf{T}_v^T\,(yv * E^T)\,\mathsf{S}_y^T \end{aligned}$$

$$\begin{aligned} AA^T &= \mathsf{S}_x(x^2u^2 * BB^T + x^2v^2 * CC^T)\mathsf{S}_x^T \\ &+ \mathsf{S}_x(xyu^2 * BD^T + xyv^2 * CE^T)\mathsf{S}_y^T \\ &+ \mathsf{S}_y(xyu^2 * DB^T + xyv^2 * EC^T)\mathsf{S}_x^T \\ &+ \mathsf{S}_y(y^2u^2 * DD^T + y^2v^2 * EE^T)\mathsf{S}_y^T \end{aligned}$$

# Propagating annotation: multiplication

$$A = \mathbf{S_x}\, xu{*}B\, \mathbf{T_u} + \mathbf{S_x}\, xv{*}C\, \mathbf{T_v} + \mathbf{S_y}\, yu{*}D\, \mathbf{T_u} + \mathbf{S_y}\, vy{*}E\, \mathbf{T_v}$$

$$\begin{aligned} \mathsf{T}_u \mathsf{T}_u^T &= \mathsf{T}_v \mathsf{T}_v^T = \mathsf{I} \\ \mathsf{T}_u \mathsf{T}_v^T &= \mathsf{T}_v \mathsf{T}_u^T = 0 \end{aligned}$$

$$\begin{aligned} A^T &= \mathsf{T}_u^T\,(xu * B^T)\,\mathsf{S}_x^T + \mathsf{T}_u^T\,(yu * D^T)\,\mathsf{S}_y^T \\ &+ \mathsf{T}_v^T\,(xv * C^T)\,\mathsf{S}_x^T + \mathsf{T}_v^T\,(yv * E^T)\,\mathsf{S}_y^T \end{aligned}$$

$$\begin{aligned} AA^T &= \mathsf{S}_x(x^2u^2 * BB^T + x^2v^2 * CC^T)\mathsf{S}_x^T \\ &+ \mathsf{S}_x(xyu^2 * BD^T + xyv^2 * CE^T)\mathsf{S}_y^T \\ &+ \mathsf{S}_y(xyu^2 * DB^T + xyv^2 * EC^T)\mathsf{S}_x^T \\ &+ \mathsf{S}_y(y^2u^2 * DD^T + y^2v^2 * EE^T)\mathsf{S}_y^T \end{aligned}$$

# Semialgebra: delete propagation

$$
\begin{aligned}
AA^T &= \mathsf{S}_x(x^2u^2 * BB^T + x^2v^2 * CC^T)\mathsf{S}_x^T \\
&+ \mathsf{S}_x(xyu^2 * BD^T + xyv^2 * CE^T)\mathsf{S}_y^T \\
&+ \mathsf{S}_y(xyu^2 * DB^T + xyv^2 * EC^T)\mathsf{S}_x^T \\
&+ \mathsf{S}_y(y^2u^2 * DD^T + y^2v^2 * EE^T)\mathsf{S}_y^T
\end{aligned}
$$

# Semialgebra: delete propagation

$$\begin{aligned} AA^T &= \mathsf{S}_x(x^2u^2 * BB^T + x^2v^2 * CC^T)\mathsf{S}_x^T \\ &+ \mathsf{S}_x(xyu^2 * BD^T + xyv^2 * CE^T)\mathsf{S}_y^T \\ &+ \mathsf{S}_y(xyu^2 * DB^T + xyv^2 * EC^T)\mathsf{S}_x^T \\ &+ \mathsf{S}_y(y^2u^2 * DD^T + y^2v^2 * EE^T)\mathsf{S}_y^T \end{aligned}$$

**Deletion propagation:** set $y = 0$

$$AA^T = x^2 * (u^2 * \mathsf{S}_x BB^T \mathsf{S}_x^T + v^2 * \mathsf{S}_x CC^T \mathsf{S}_x^T)$$

# Preliminary application: solving equations

- $(A + B)\, x = b$, $A$ and $B$ are square matrices
- $A$ is from source $p$, $B$ is from source $q$

# Preliminary application: solving equations

- $(A + B)\,x = b$, $A$ and $B$ are square matrices
- $A$ is from source $p$, $B$ is from source $q$

- Jacobi method: iteratively compute

$$u_{k+1} = (M^{-1}N)u_k + M^{-1}b \qquad u_0 = \bar{0}$$

- $M = p * \text{diag}(A)$, $N = p * (\text{diag}(A) - A) - q * B$

# Jacobi method: example

- Iteratively compute

$$u_{k+1} = (M^{-1}N)u_k + M^{-1}b \qquad u_0 = \bar{0}$$

- $M = p * \mathrm{diag}(A)$, $N = p * (\mathrm{diag}(A) – A) – q * B$

# Jacobi method: example

- Iteratively compute

$$u_{k+1} = (M^{-1}N)u_k + M^{-1}b \qquad u_0 = \bar{0}$$

- $M = p * \mathrm{diag}(A)$, $N = p * (\mathrm{diag}(A) - A) - q * B$

$$A = p * \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, B = q * \begin{bmatrix} 0 & 0 \\ -2 & 0 \end{bmatrix}, b = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

$$u_1 = p * \begin{bmatrix} \frac{1}{2} \\ -\frac{1}{2} \end{bmatrix}, u_2 = p * \begin{bmatrix} \frac{1}{2} \\ -\frac{1}{2} \end{bmatrix} + p^3 * \begin{bmatrix} \frac{1}{4} \\ -\frac{1}{4} \end{bmatrix} + p^2 q * \begin{bmatrix} 0 \\ -\frac{1}{2} \end{bmatrix}, \cdots$$

# Preliminary applications

- Solving systems of linear equations
- Also in the paper
  - Largest eigenvalue
  - PageRank

# Contributions

- First steps towards a semantics-preserving notion of fine-grained provenance for linear algebra operators
  - Key development: decomposition, tensor-product construction, and algebraic rules

- Preliminary applications in solving equations, computing largest eigenvalues, and PageRank.

- Key benefit
  - Automatic propagation of annotations through operators
  - Ability to assign values (e.g., 0 or 1) to the annotations and propagate the effects, e.g., for deletion or trust

# Related and future work

- Provenance Semirings / Semimodules
  - Green et al. PODS'07, Amsterdamer et al. PODS'11
- Array databases
  - SciDB, RasDaMan
  - Wu et al. SubZero, Peng and Diao SIGMOD'15
- Distributed machine learning / linear algebra programs
  - SystemML, Spark, MLbase, Cumulon, MADlib, GraphX, LINView, etc

- Future work
  - Support more linear algebra operators
  - Scalable implementation

# Thank you!