Data thinning to overcome double dipping

Anna Neufeld
Final Exam
May 9, 2023

What is double dipping?

Classical statistical methods assume that we only ever test pre-specified hypotheses about pre-specified models.

What is double dipping?

Classical statistical methods assume that we only ever test pre-specified hypotheses about pre-specified models.

In reality, we explore our data, fit several models, evaluate these models, select our favorite model, then test hypotheses about this model.

What is double dipping?

Classical statistical methods assume that we only ever test pre-specified hypotheses about pre-specified models.

In reality, we explore our data, fit several models, evaluate these models, select our favorite model, then test hypotheses about this model.

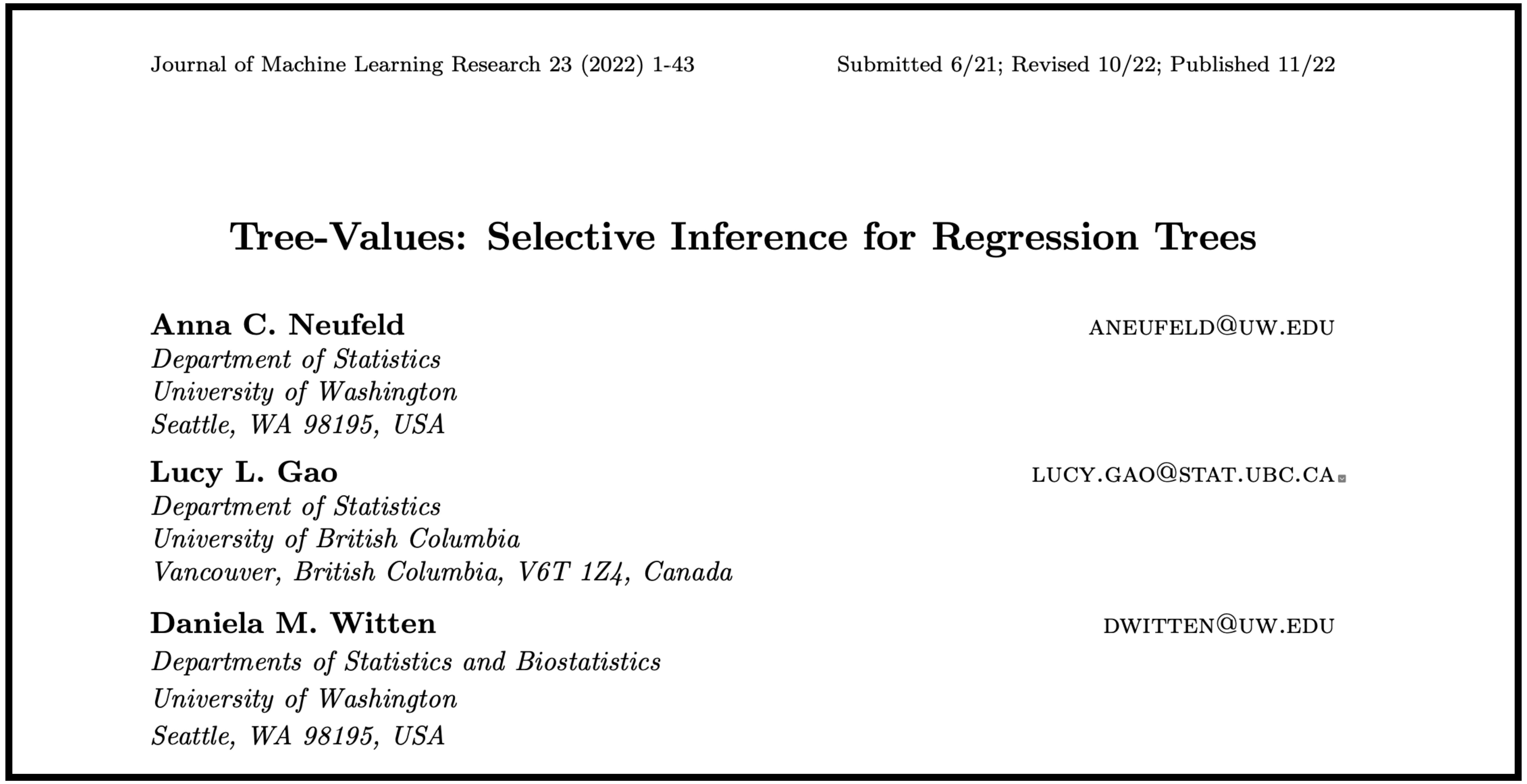
Double Dipping: Using the same data for two tasks, such as:

1. Generating and testing a null hypothesis.
2. Fitting and evaluating a model.

Approach 1: develop specialized procedures that account for double dipping

Approach 1: develop specialized procedures that account for double dipping

Project 1



Journal of Machine Learning Research 23 (2022) 1-43 Submitted 6/21; Revised 10/22; Published 11/22

Tree-Values: Selective Inference for Regression Trees

Anna C. Neufeld ANEUFELD@UW.EDU
Department of Statistics
University of Washington
Seattle, WA 98195, USA

Lucy L. Gao LUCY.GAO@STAT.UBC.CA
Department of Statistics
University of British Columbia
Vancouver, British Columbia, V6T 1Z4, Canada

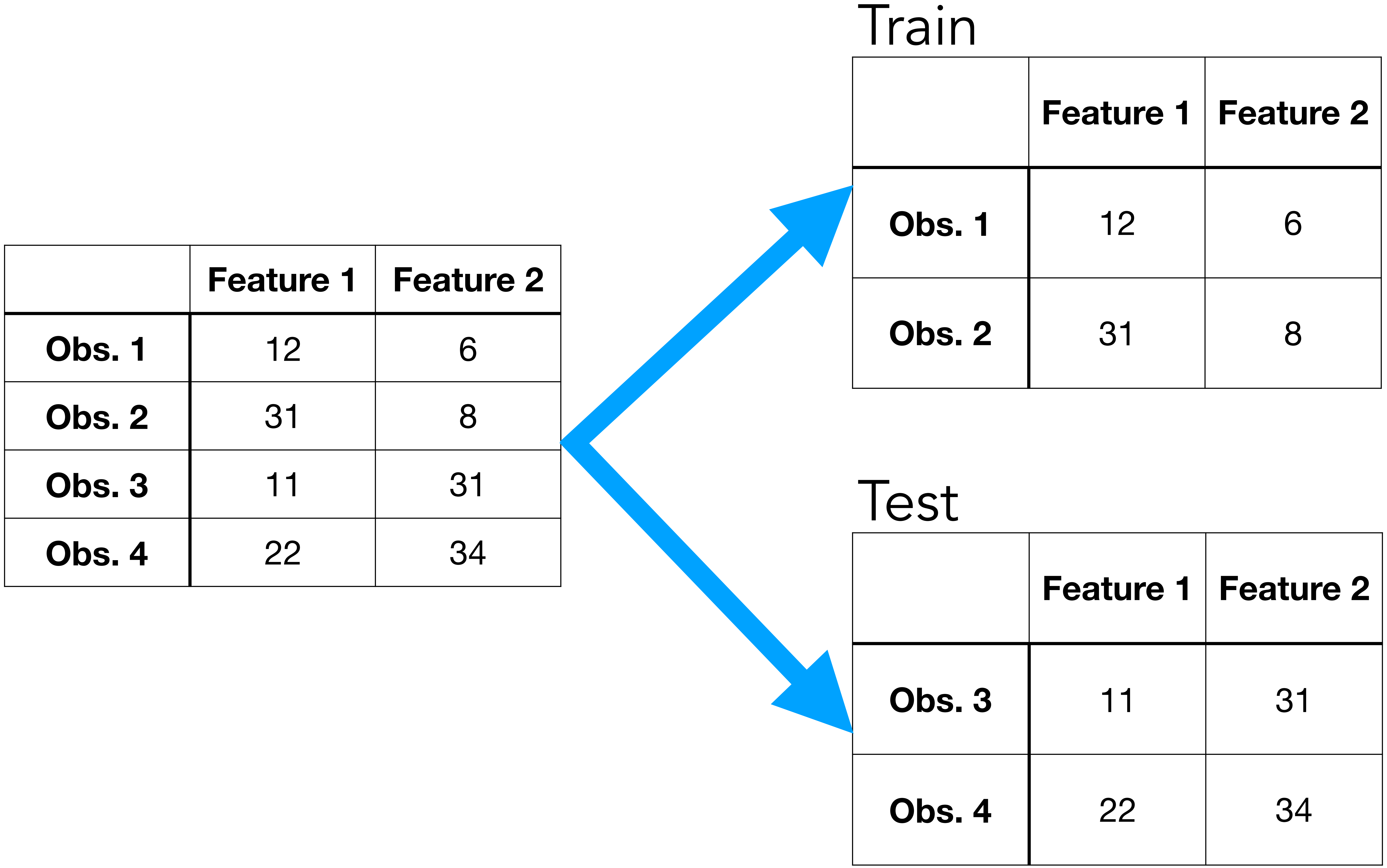
Daniela M. Witten DWITTEN@UW.EDU
Departments of Statistics and Biostatistics
University of Washington
Seattle, WA 98195, USA

R package and tutorials: https://anna-neufeld.github.io/treevalues/

Approach 2: avoid double dipping entirely via sample splitting

	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

Approach 2: avoid double dipping entirely via sample splitting



	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

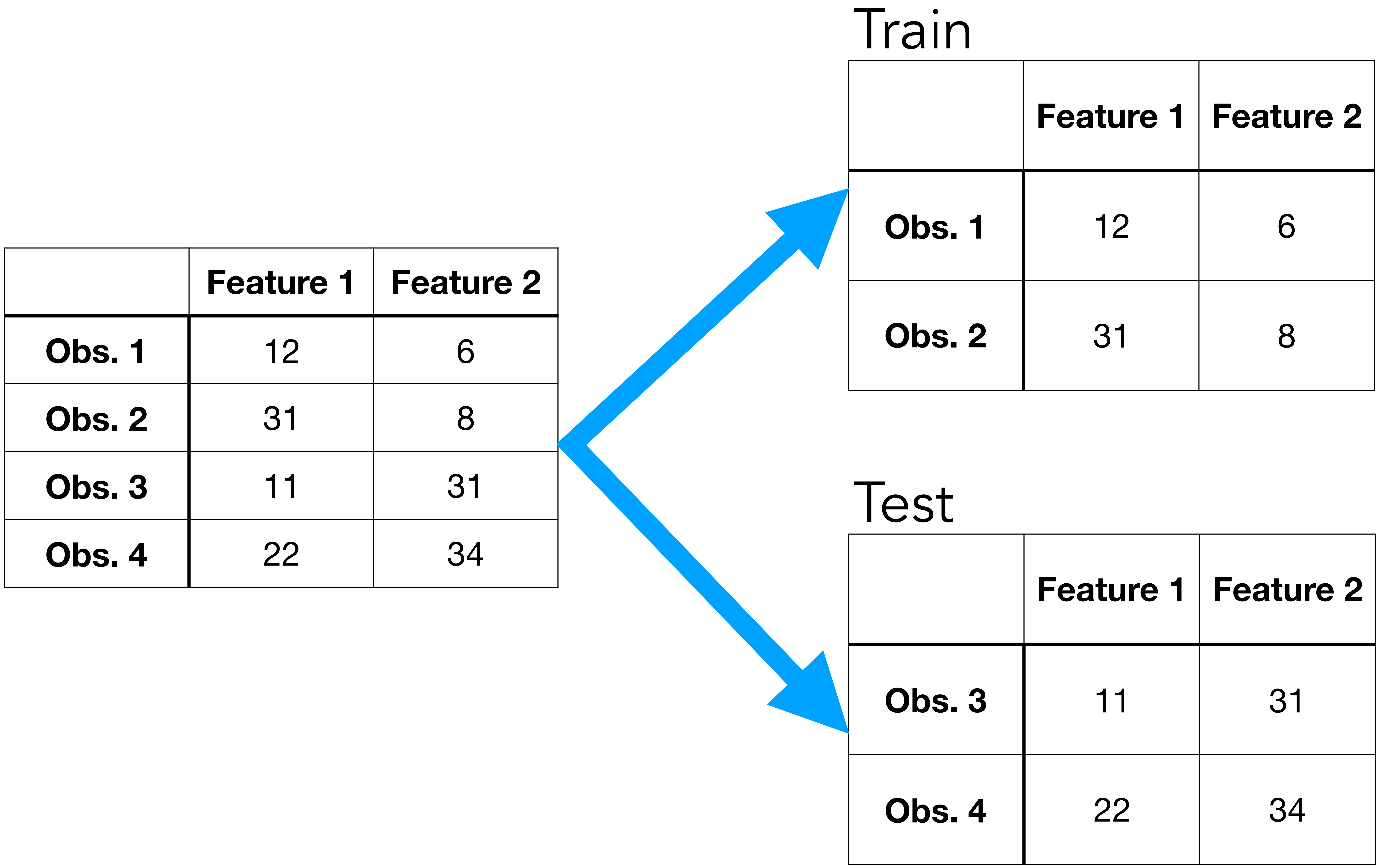
Train

	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8

Test

	Feature 1	Feature 2
Obs. 3	11	31
Obs. 4	22	34

Approach 2: avoid double dipping entirely via sample splitting



	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

Train

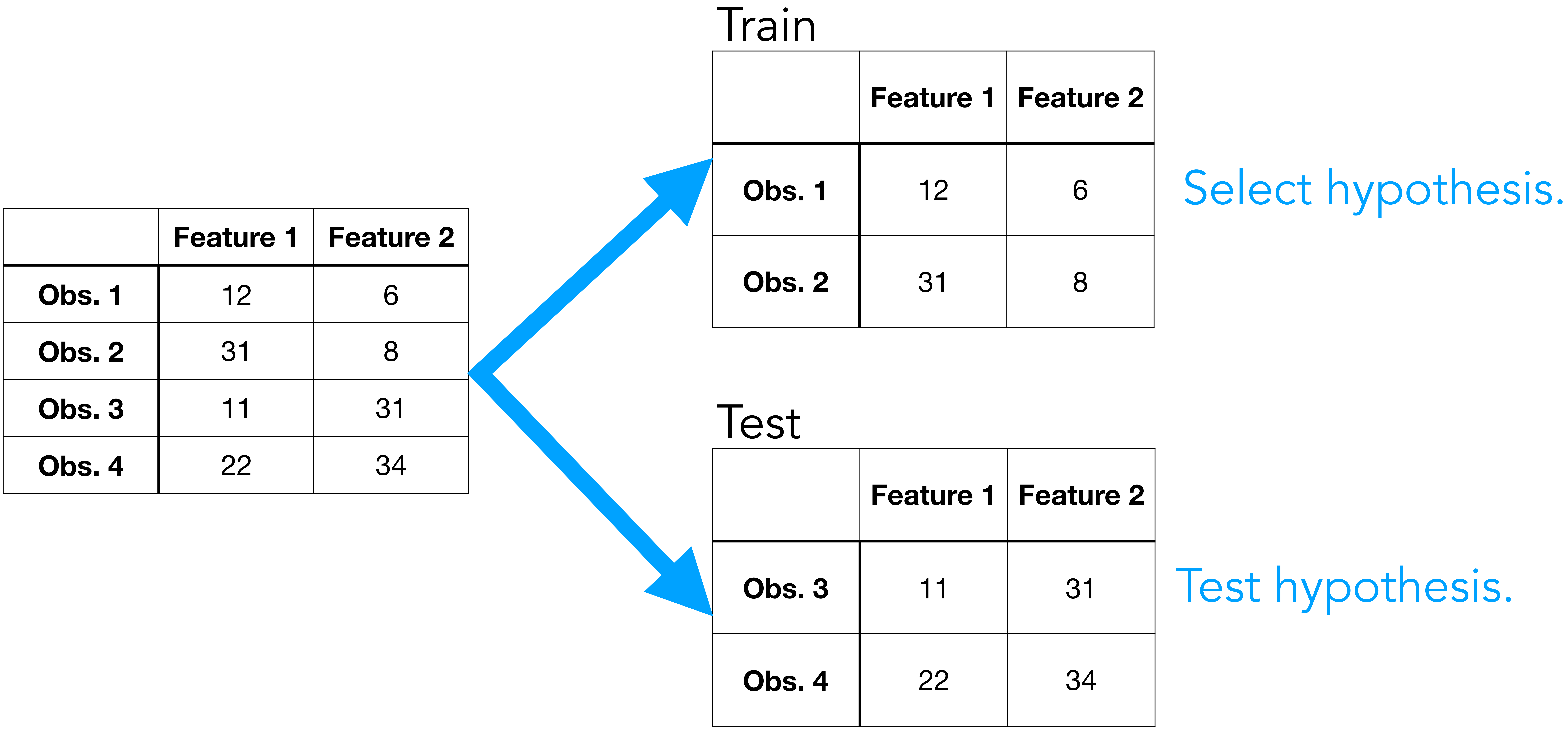
	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8

Test

	Feature 1	Feature 2
Obs. 3	11	31
Obs. 4	22	34

Select hypothesis.

Approach 2: avoid double dipping entirely via sample splitting



	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

Train

	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8

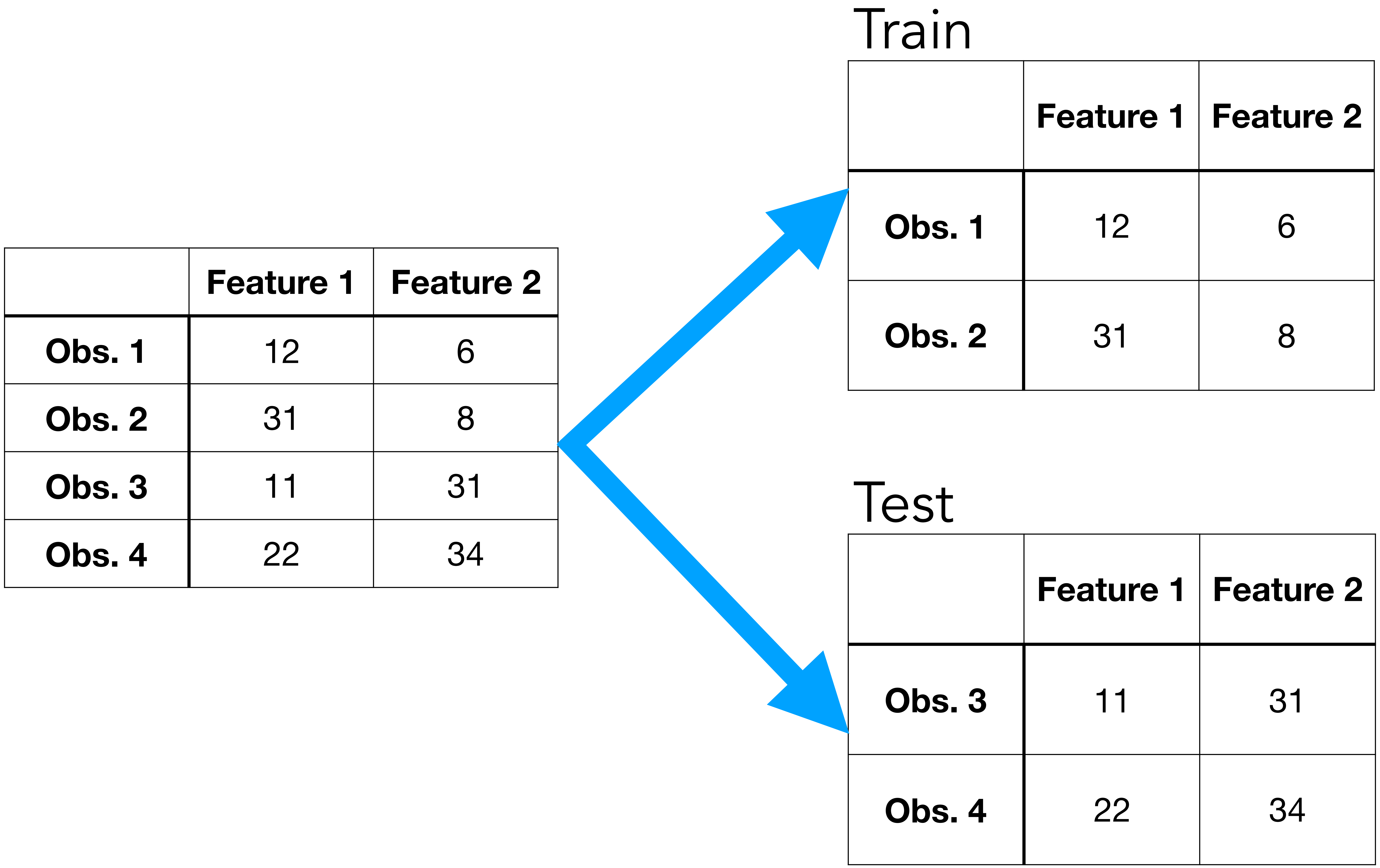
Select hypothesis.

Test

	Feature 1	Feature 2
Obs. 3	11	31
Obs. 4	22	34

Test hypothesis.

Approach 2: avoid double dipping entirely via sample splitting



	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

Train

	Feature 1	Feature 2
Obs. 1	12	6
Obs. 2	31	8

Fit model.

Test

	Feature 1	Feature 2
Obs. 3	11	31
Obs. 4	22	34

Approach 2: avoid double dipping entirely via sample splitting

	Feature 1	**Feature 2**
Obs. 1	12	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

Train

	Feature 1	**Feature 2**
Obs. 1	12	6
Obs. 2	31	8

Fit model.

Test

	Feature 1	**Feature 2**
Obs. 3	11	31
Obs. 4	22	34

Evaluate model.

Outline

1. **Motivation: settings where sample splitting doesn’t work**

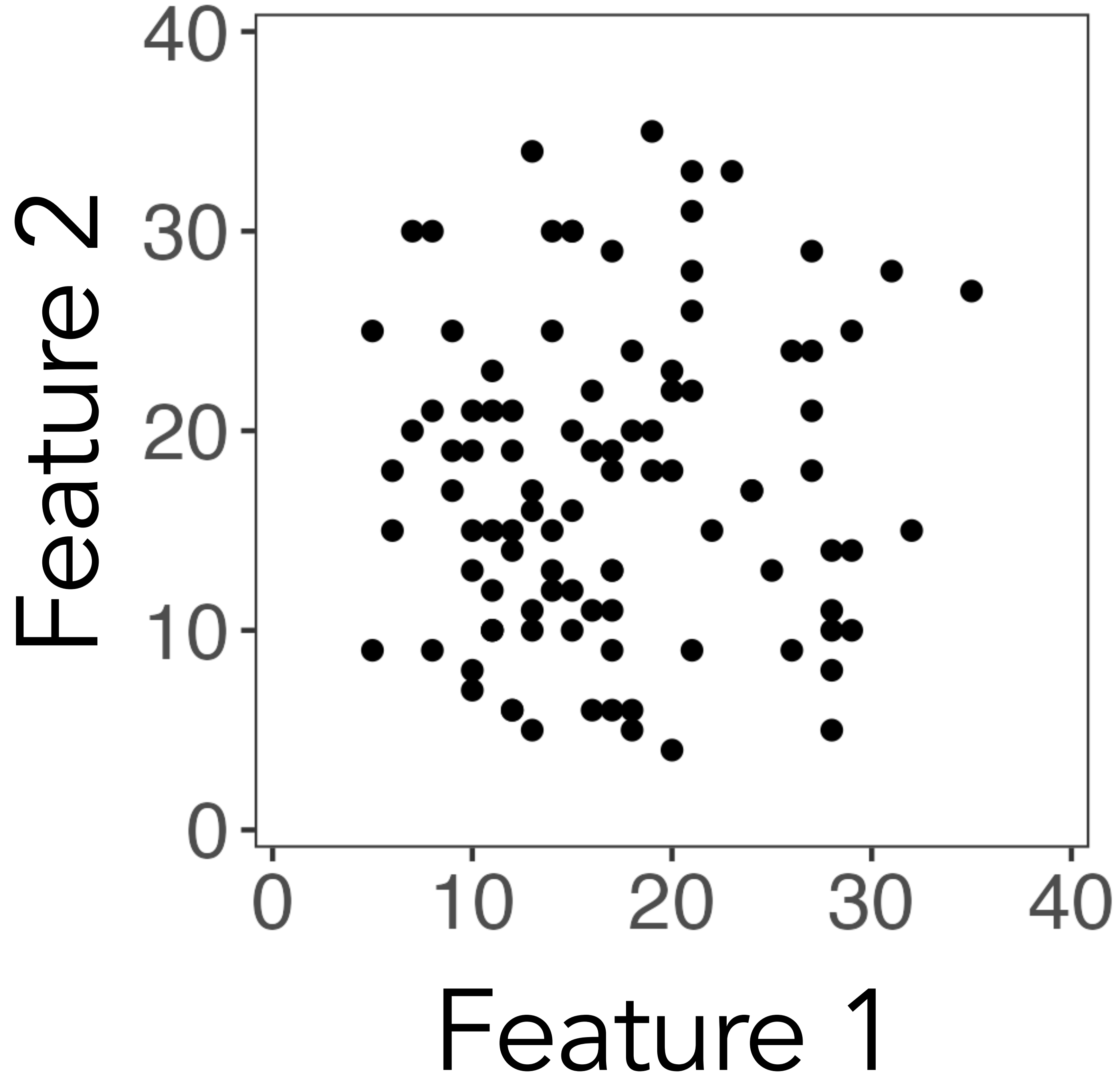
2. Poisson thinning

3. Data thinning

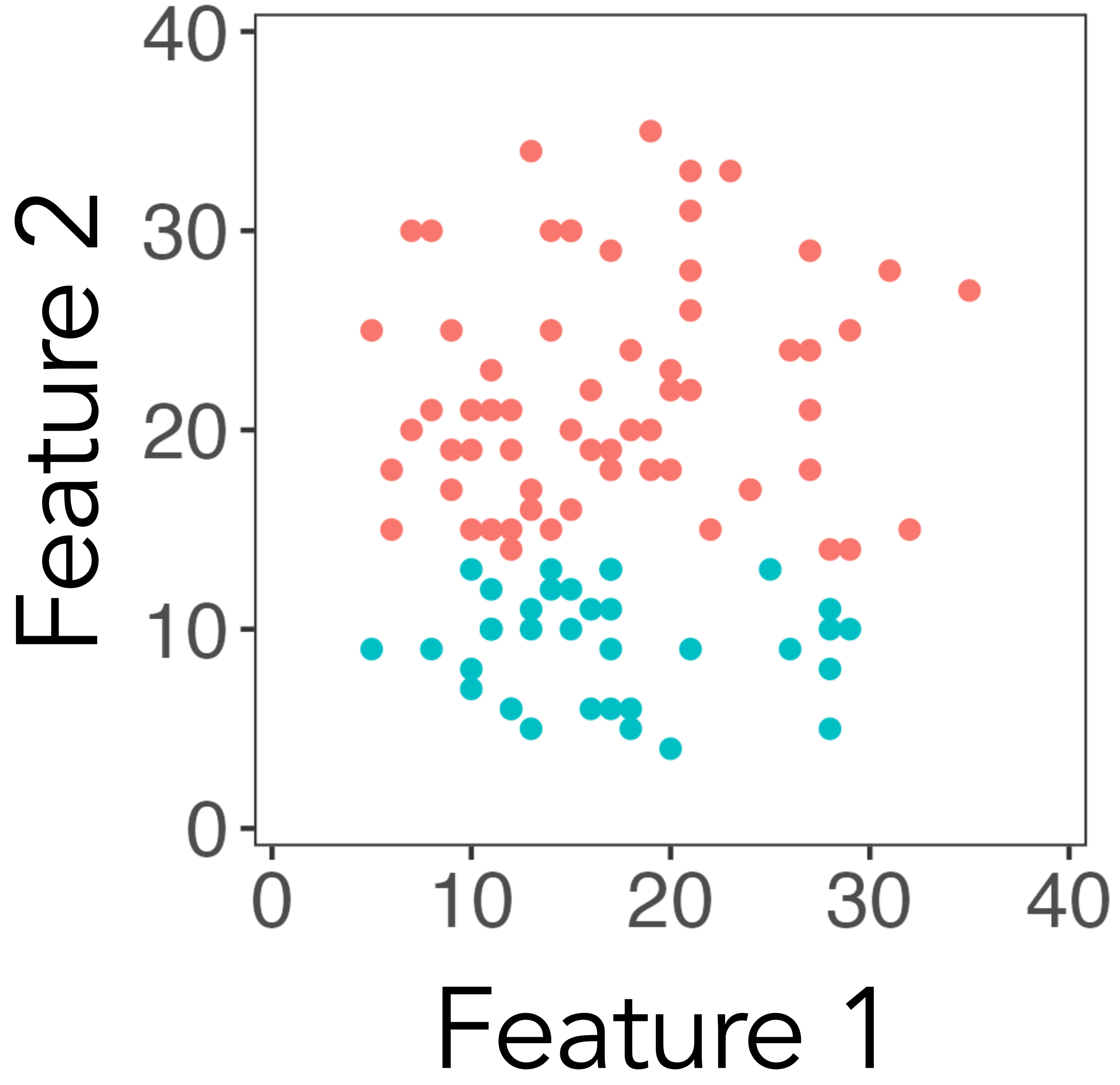
4. Application to single-cell RNA sequencing data

5. Ongoing work

Example 1: using the same data to generate and test a hypothesis

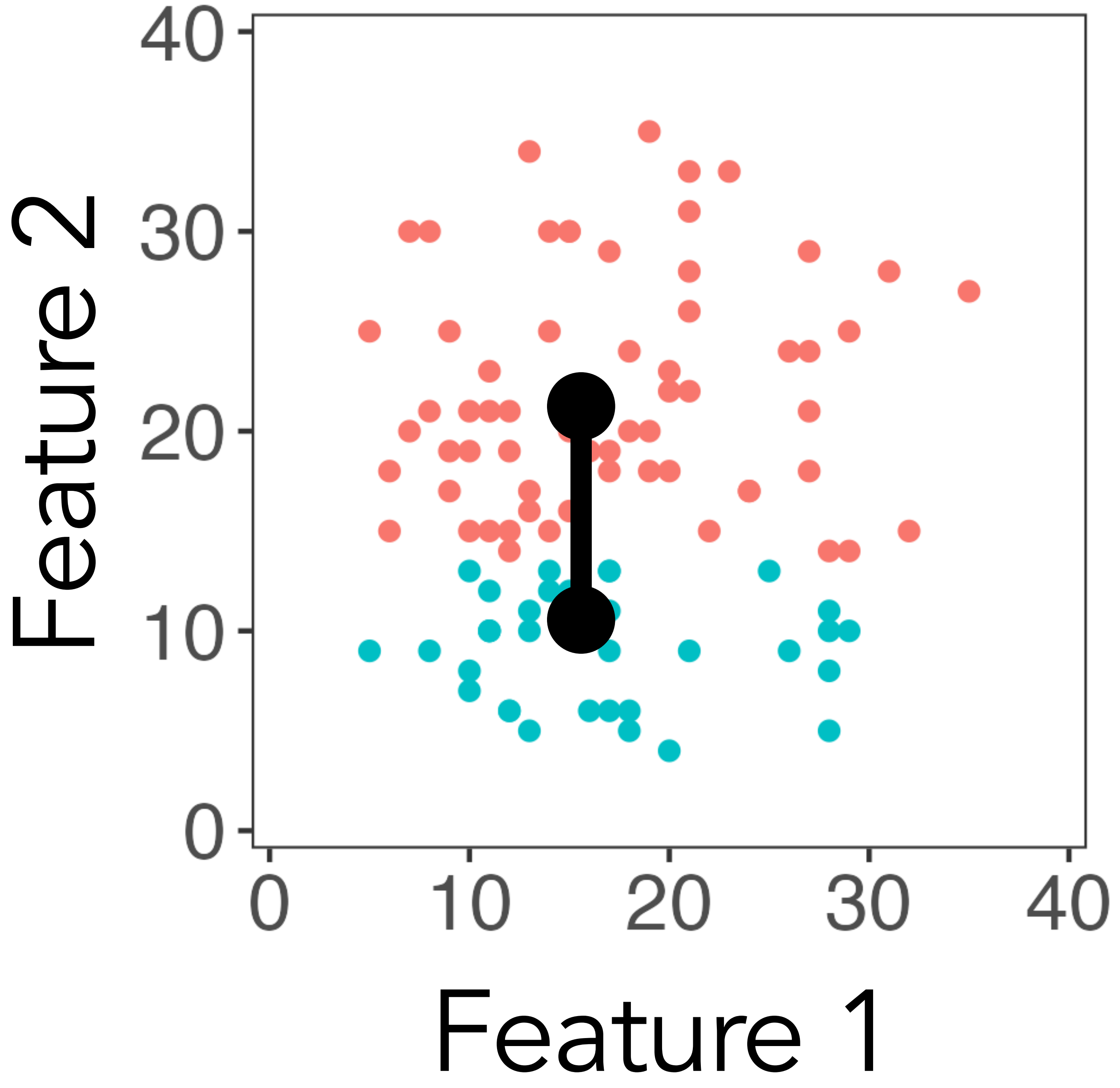


Example 1: using the same data to generate and test a hypothesis



Step 1: cluster the observations.

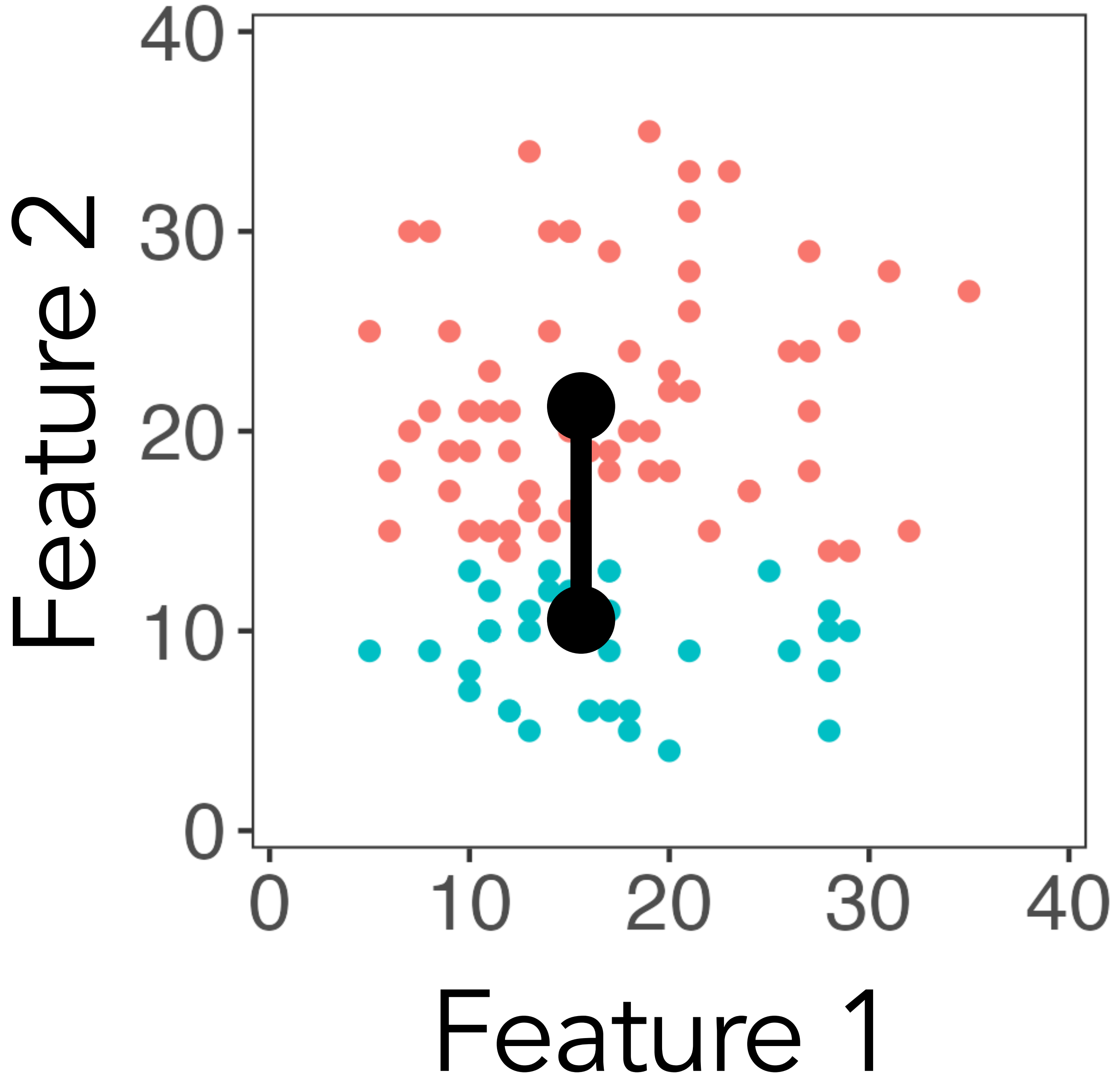
Example 1: using the same data to generate and test a hypothesis



Step 1: cluster the observations.

Generate H_0 :"the expected value of Feature 2 is the same between red observations and the blue observations."

Example 1: using the same data to generate and test a hypothesis



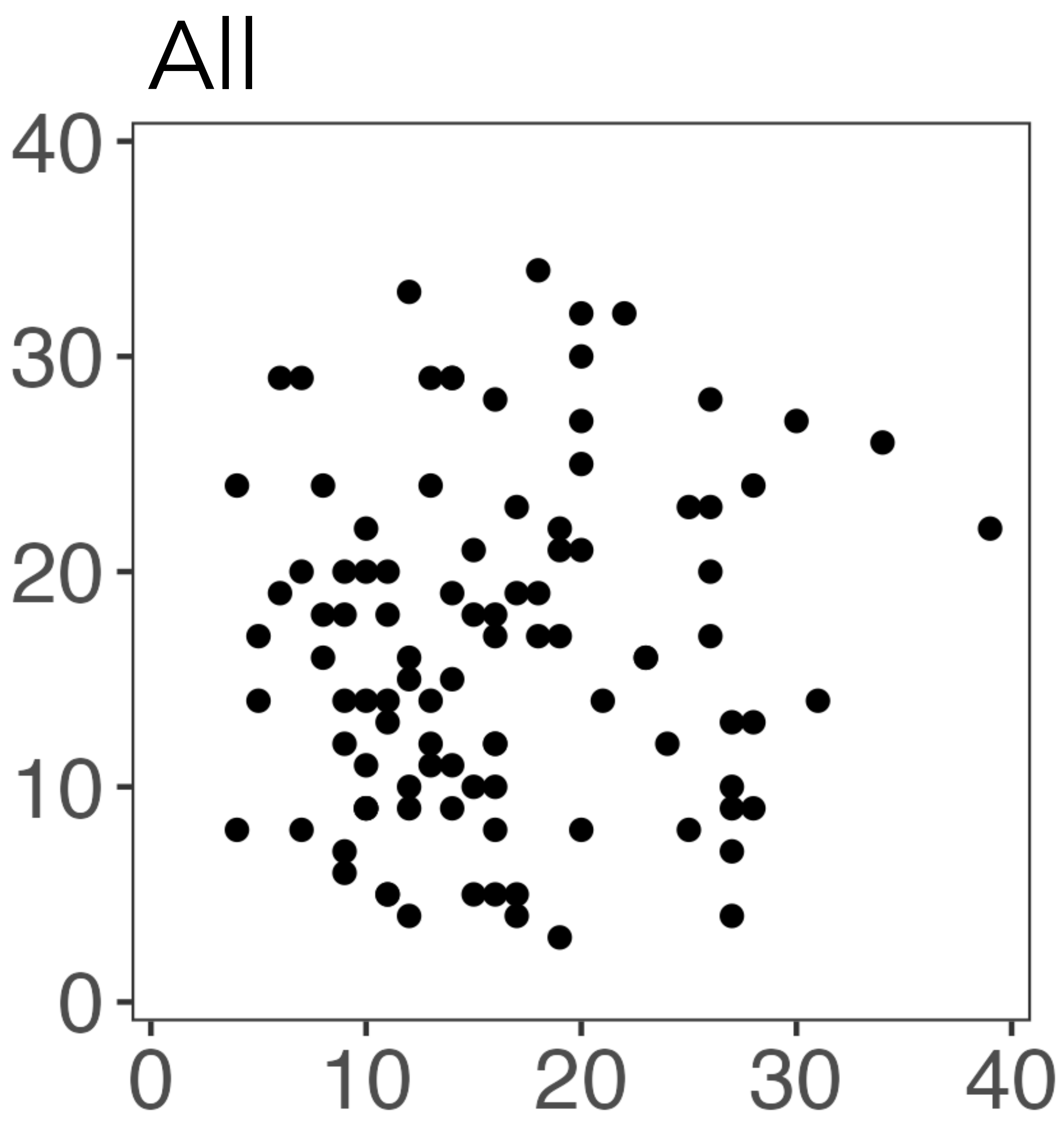
$p < 10^{-10}$ 😱

Step 1: cluster the observations.

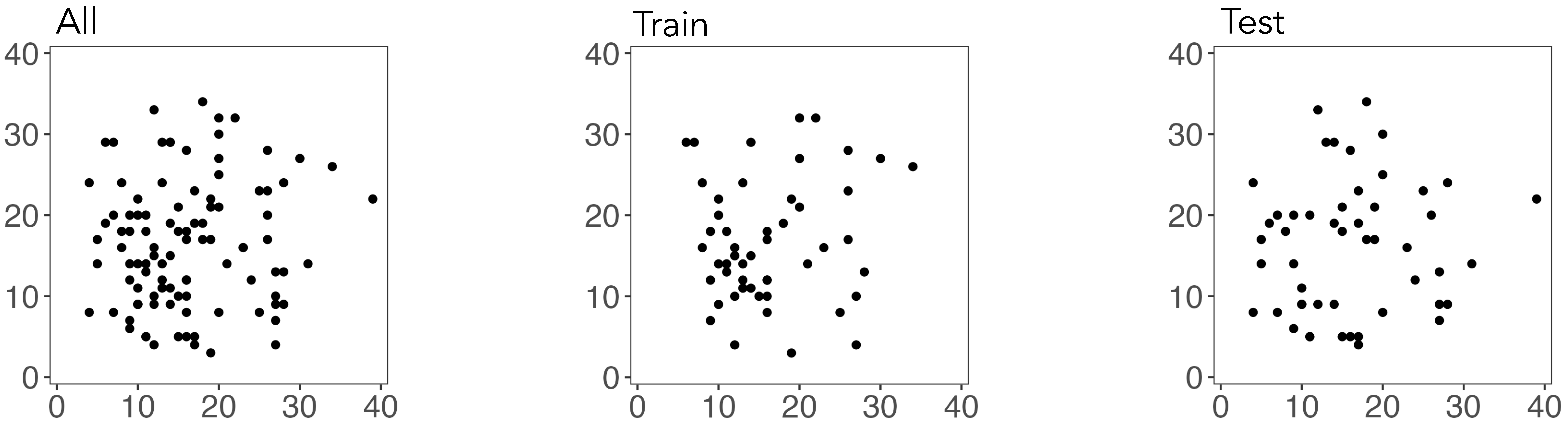
Generate H_0 :"the expected value of Feature 2 is the same between red observations and the blue observations."

Step 2: test H_0 with a t-test.

Sample splitting cannot be used for example 1

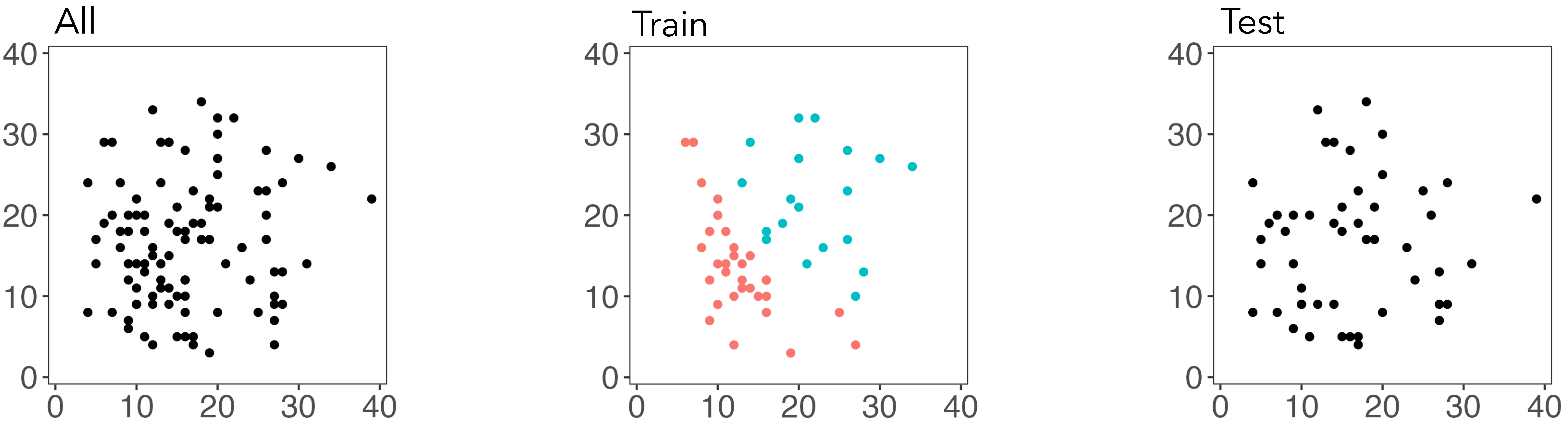


Sample splitting cannot be used for example 1



Step 1: split observations into train/test.

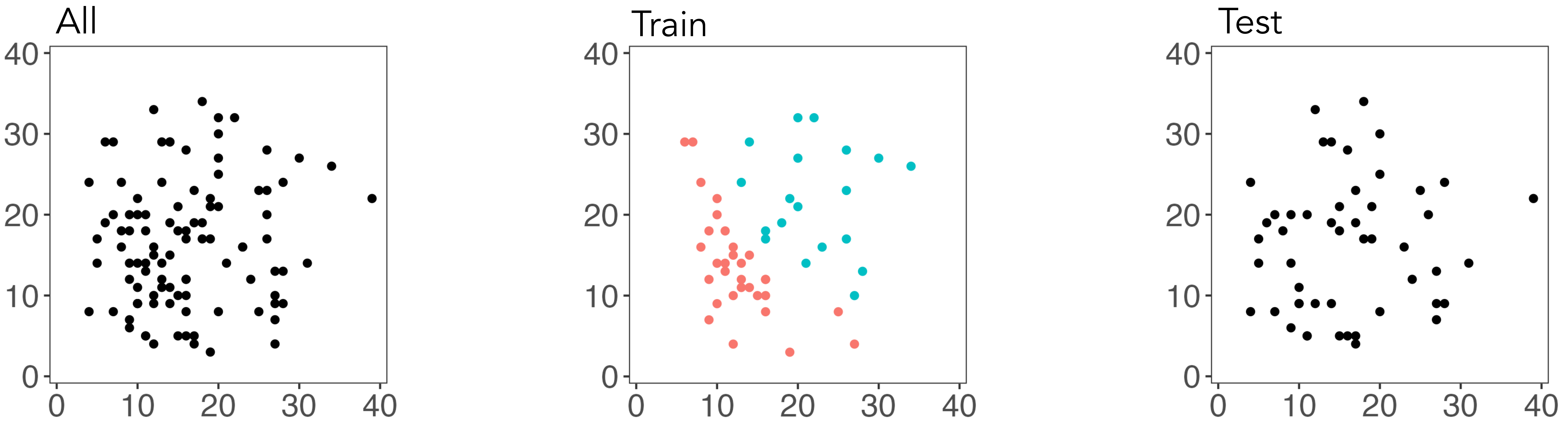
Sample splitting cannot be used for example 1



Step 1: split observations into train/test.

Step 2: cluster the training set.

Sample splitting cannot be used for example 1

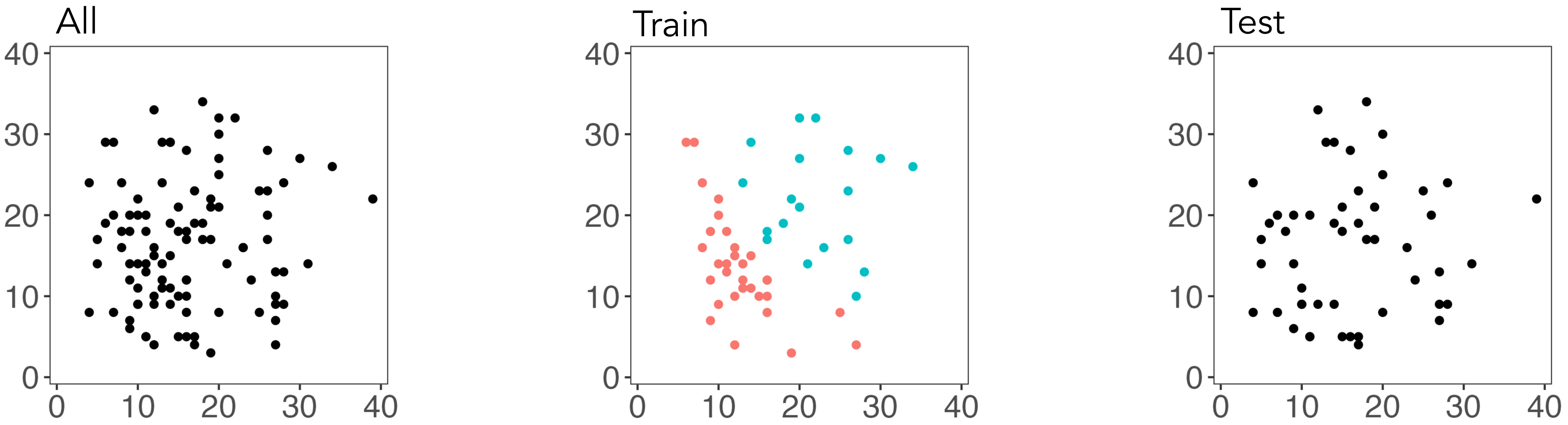


Step 1: split observations into train/test.

Step 2: cluster the training set.

Step 3: test for difference in means using test set.

Sample splitting cannot be used for example 1



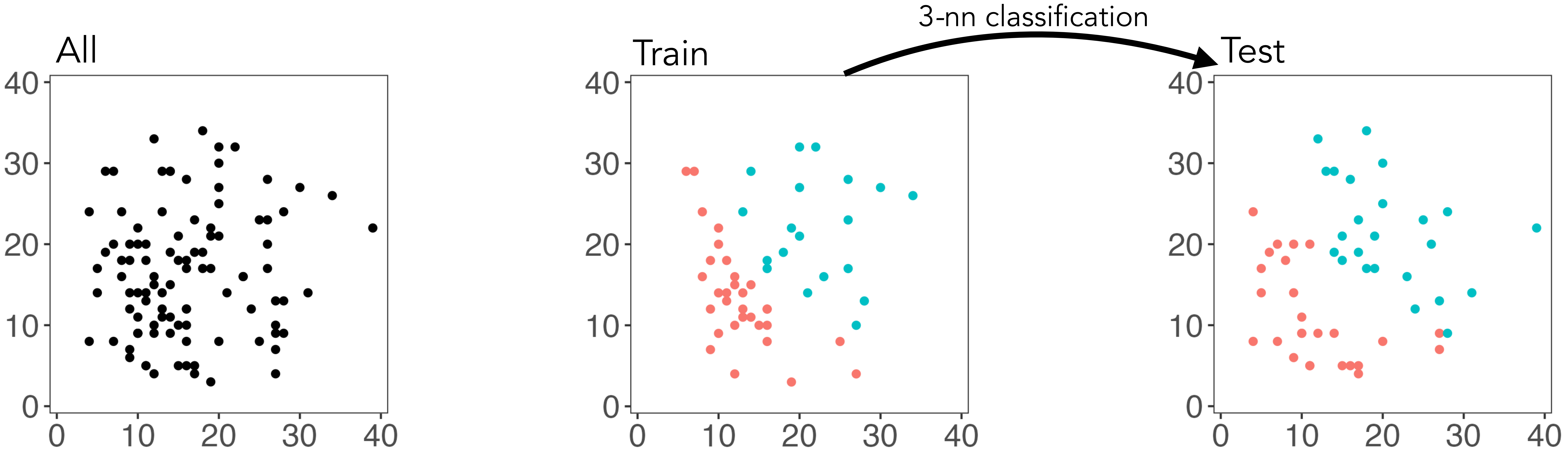
Step 1: split observations into train/test.

Step 2: cluster the training set.

Step 2.5: assign labels to observations in test set.

Step 3: test for difference in means using test set.

Sample splitting cannot be used for example 1



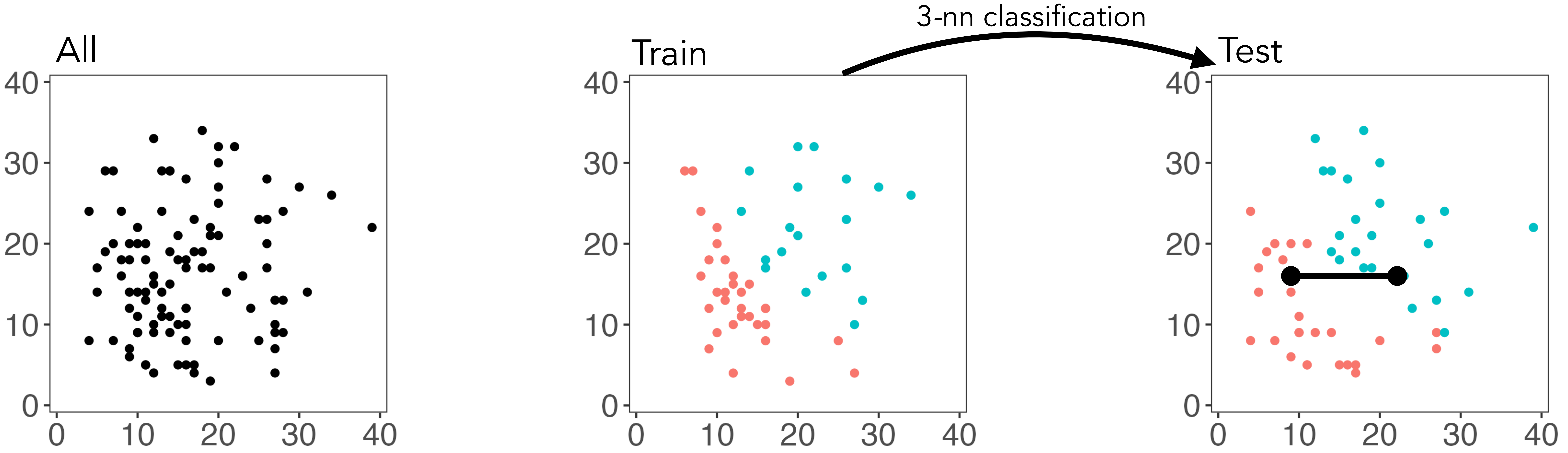
Step 1: split observations into train/test.

Step 2: cluster the training set.

Step 2.5: assign labels to observations in test set.

Step 3: test for difference in means using test set.

Sample splitting cannot be used for example 1



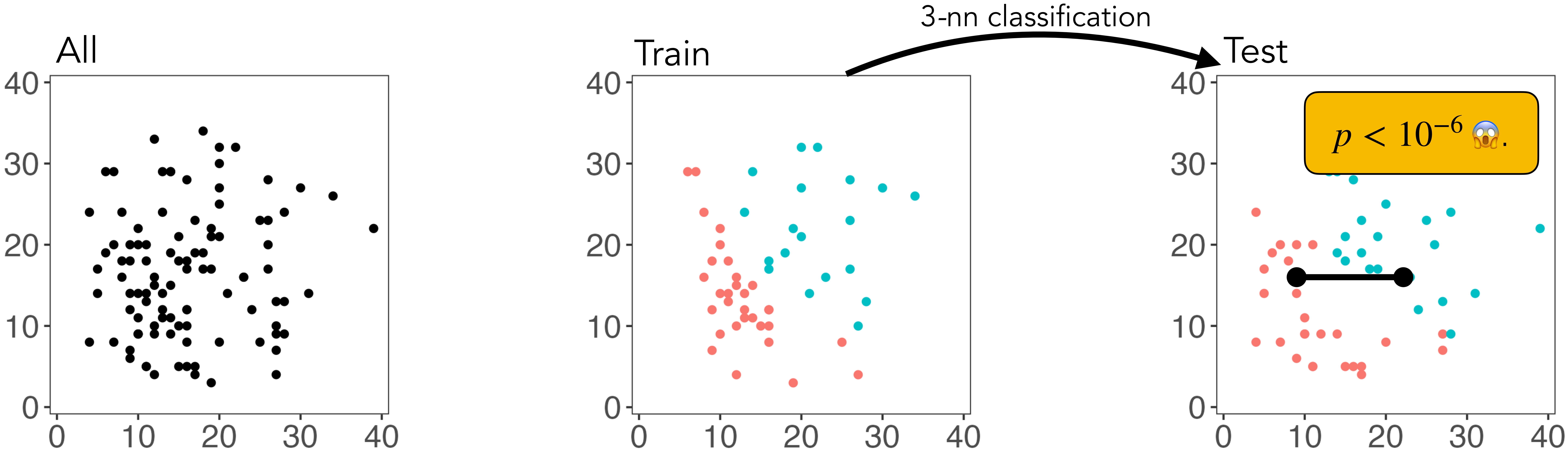
Step 1: split observations into train/test.

Step 2: cluster the training set.

Step 2.5: assign labels to observations in test set.

Step 3: test for difference in means using test set.

Sample splitting cannot be used for example 1



$p < 10^{-6}$ 😱.

Step 1: split observations into train/test.

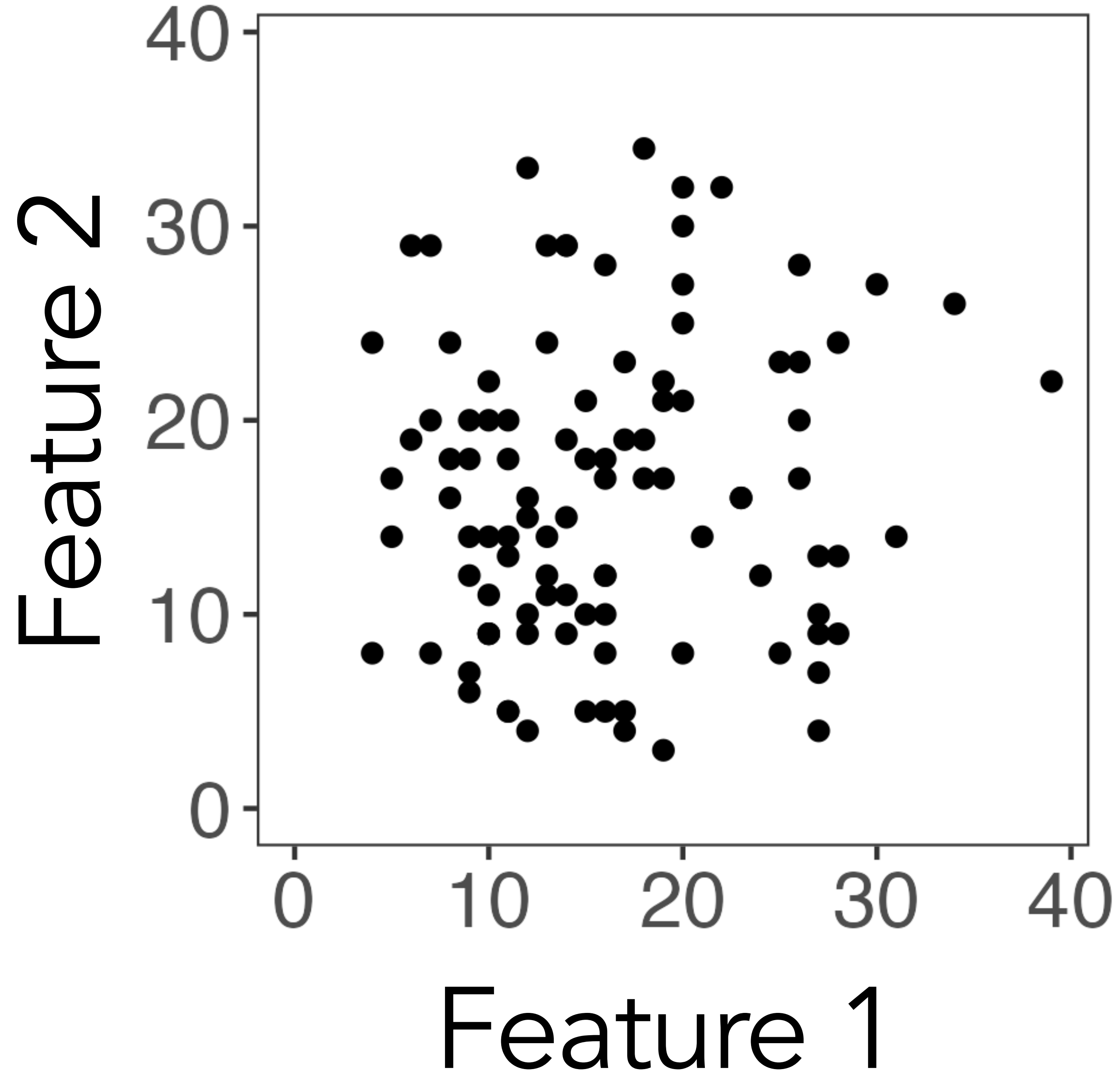
Step 2: cluster the training set.

Step 2.5: assign labels to observations in test set.

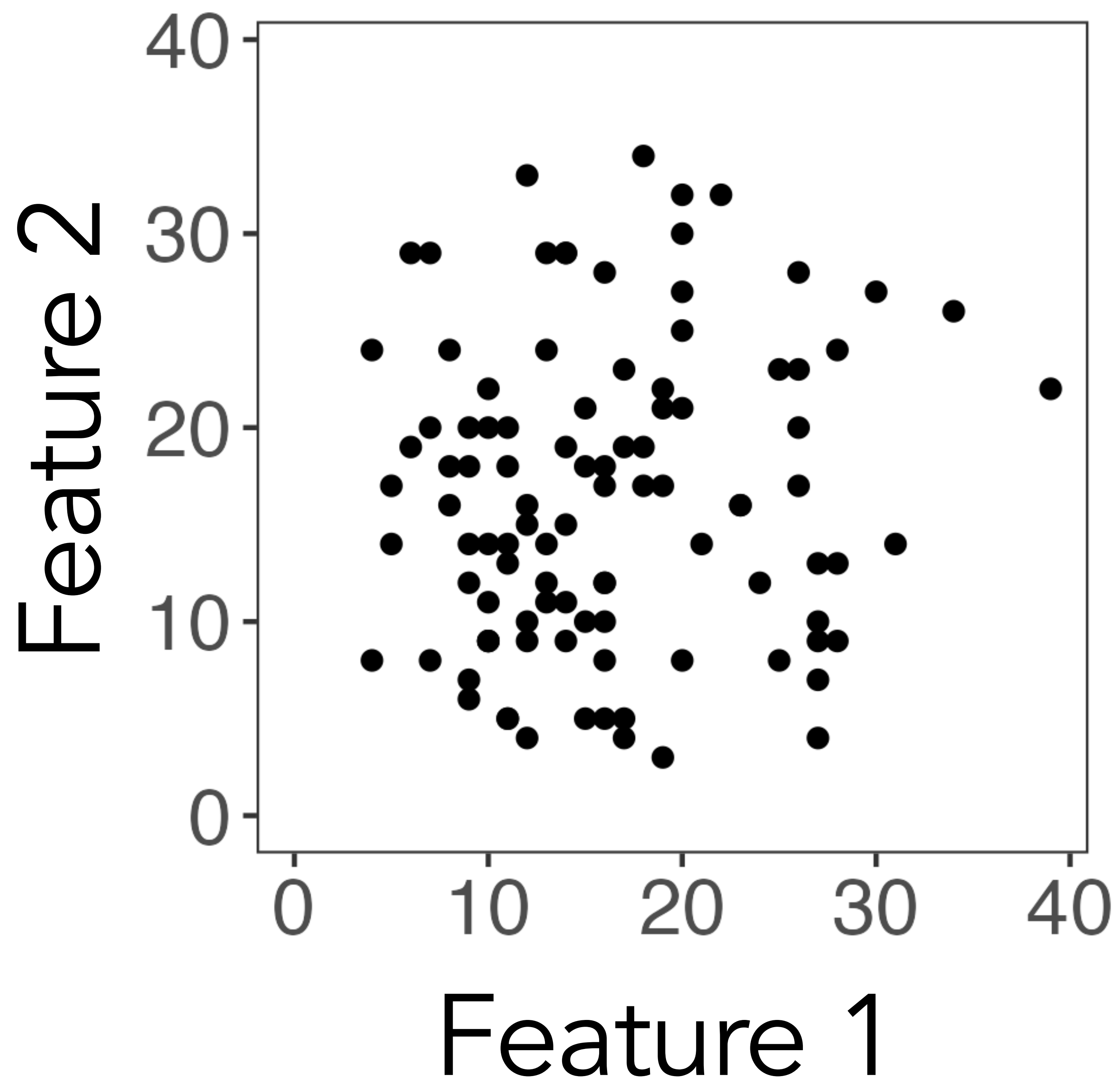
Step 3: test for difference in means using test set.

Gao, Bien, and Witten, 2022 (JASA).

Example 2: using the same data to fit and evaluate a model

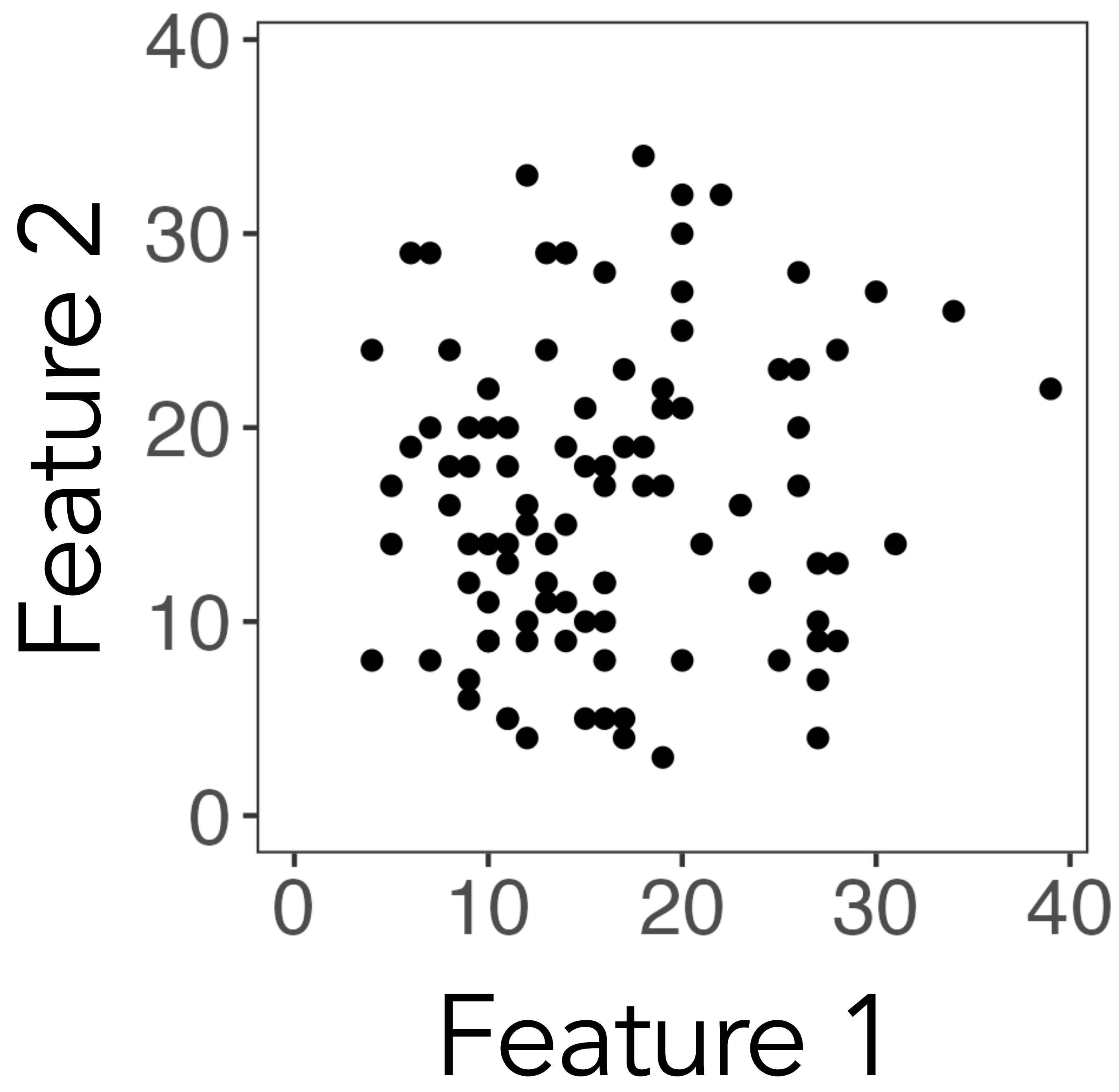


Example 2: using the same data to fit and evaluate a model



Goal: how many clusters are in this data?

Example 2: using the same data to fit and evaluate a model



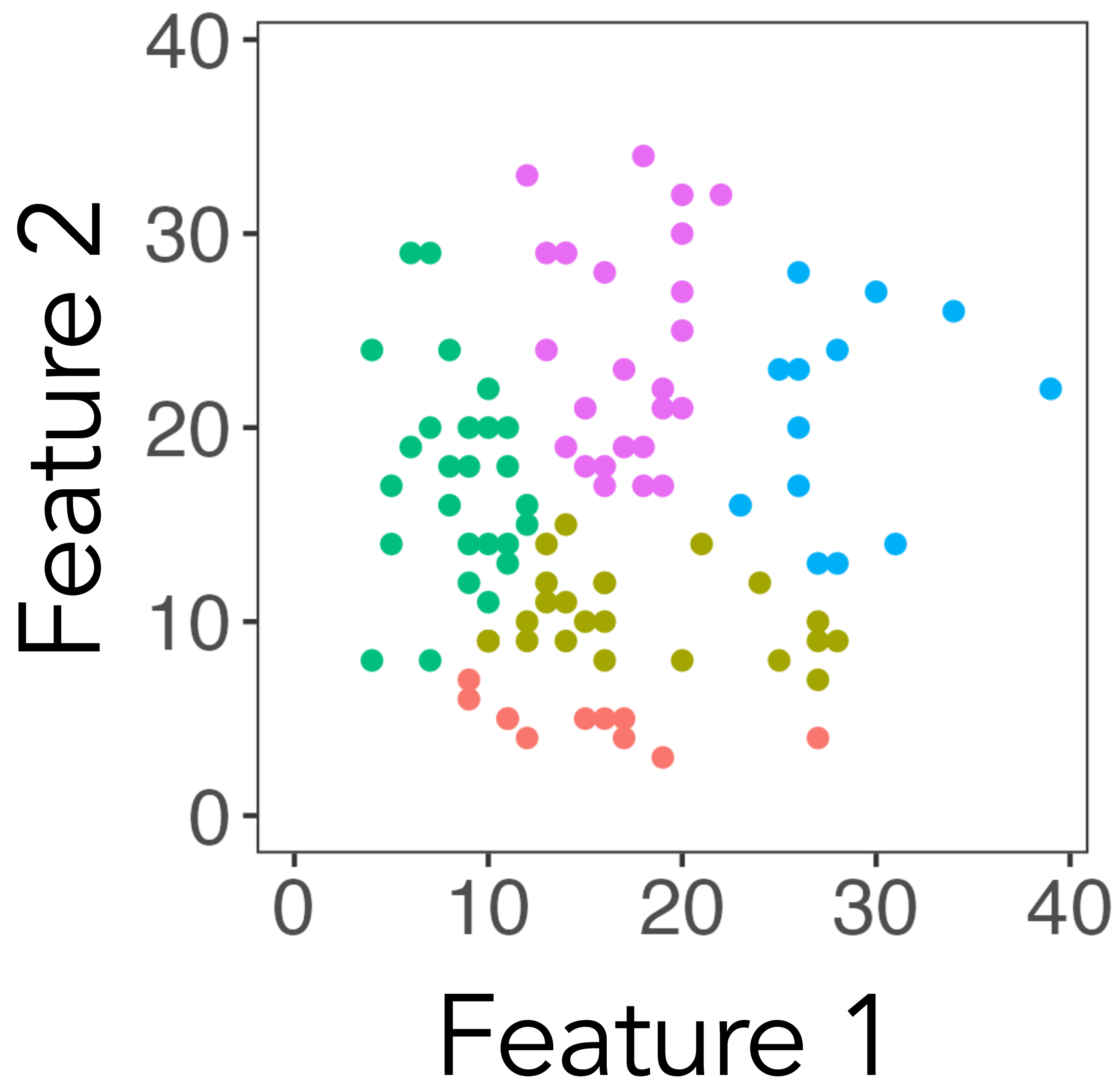
Goal: how many clusters are in this data?

For several values of k:

Step 1: fit a model with k clusters.

Step 2: evaluate model using a loss function.

Example 2: using the same data to fit and evaluate a model



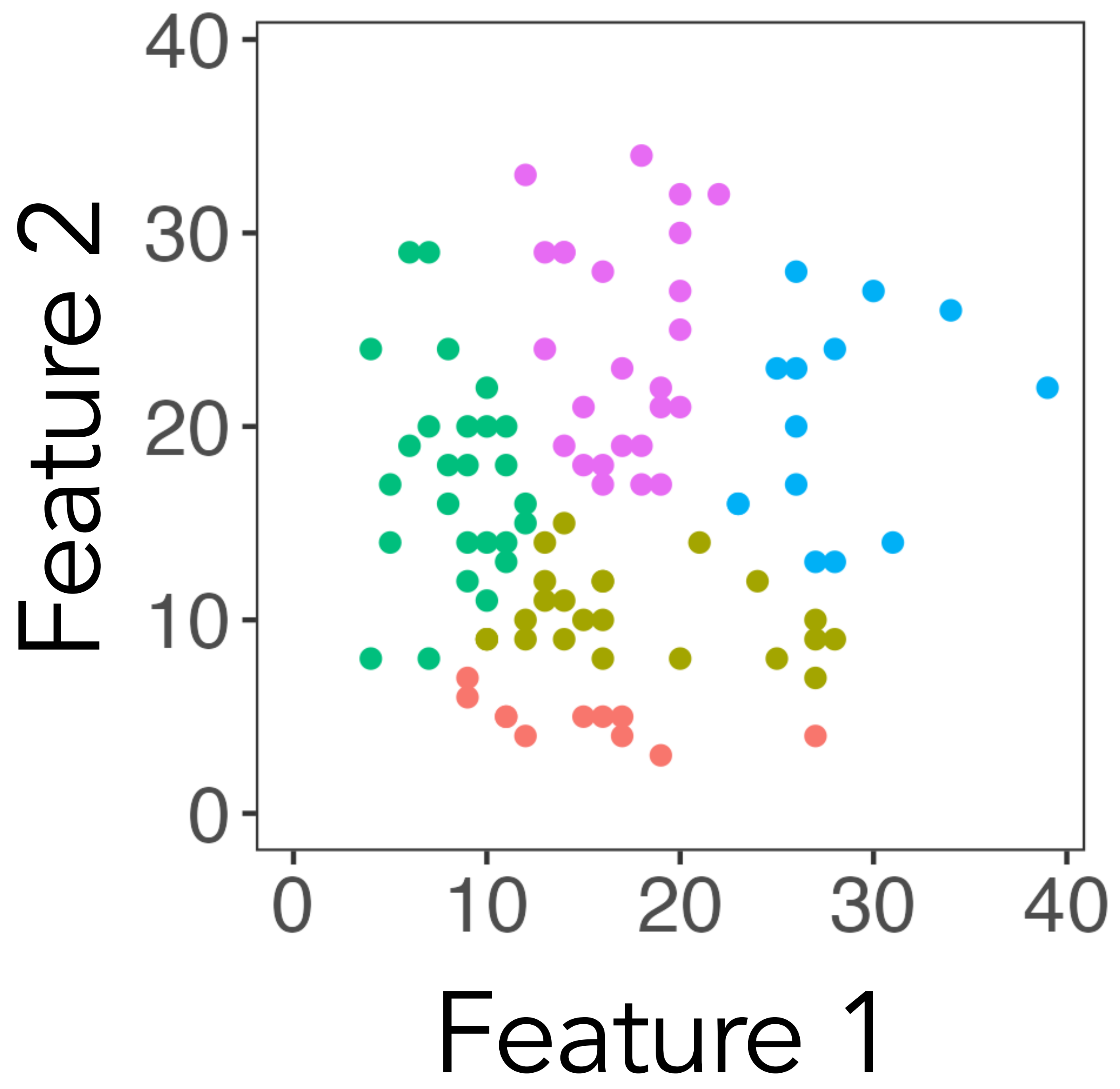
Goal: how many clusters are in this data?

For several values of k:

Step 1: fit a model with k clusters.

Step 2: evaluate model using a loss function.

Example 2: using the same data to fit and evaluate a model

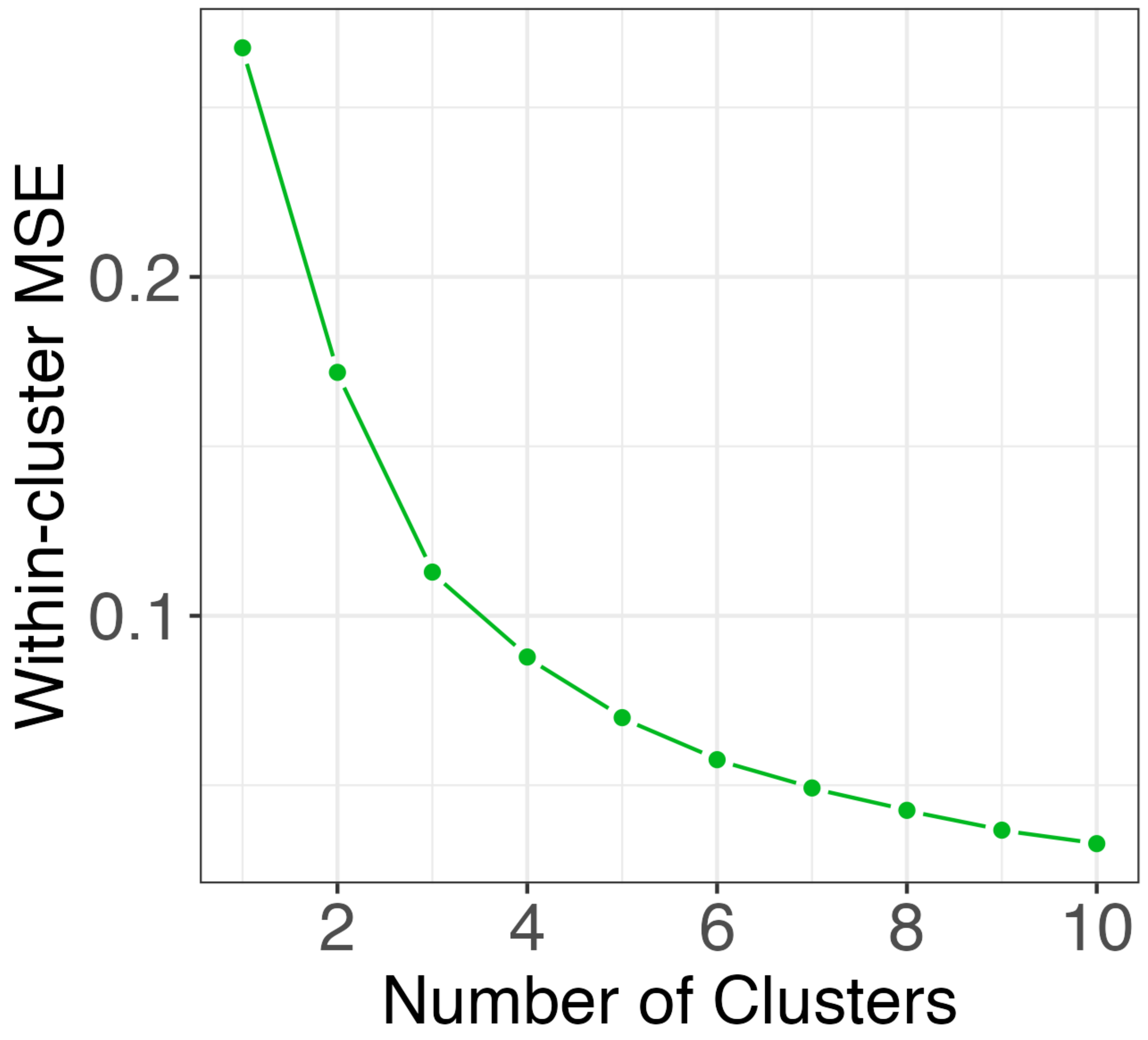


Goal: how many clusters are in this data?

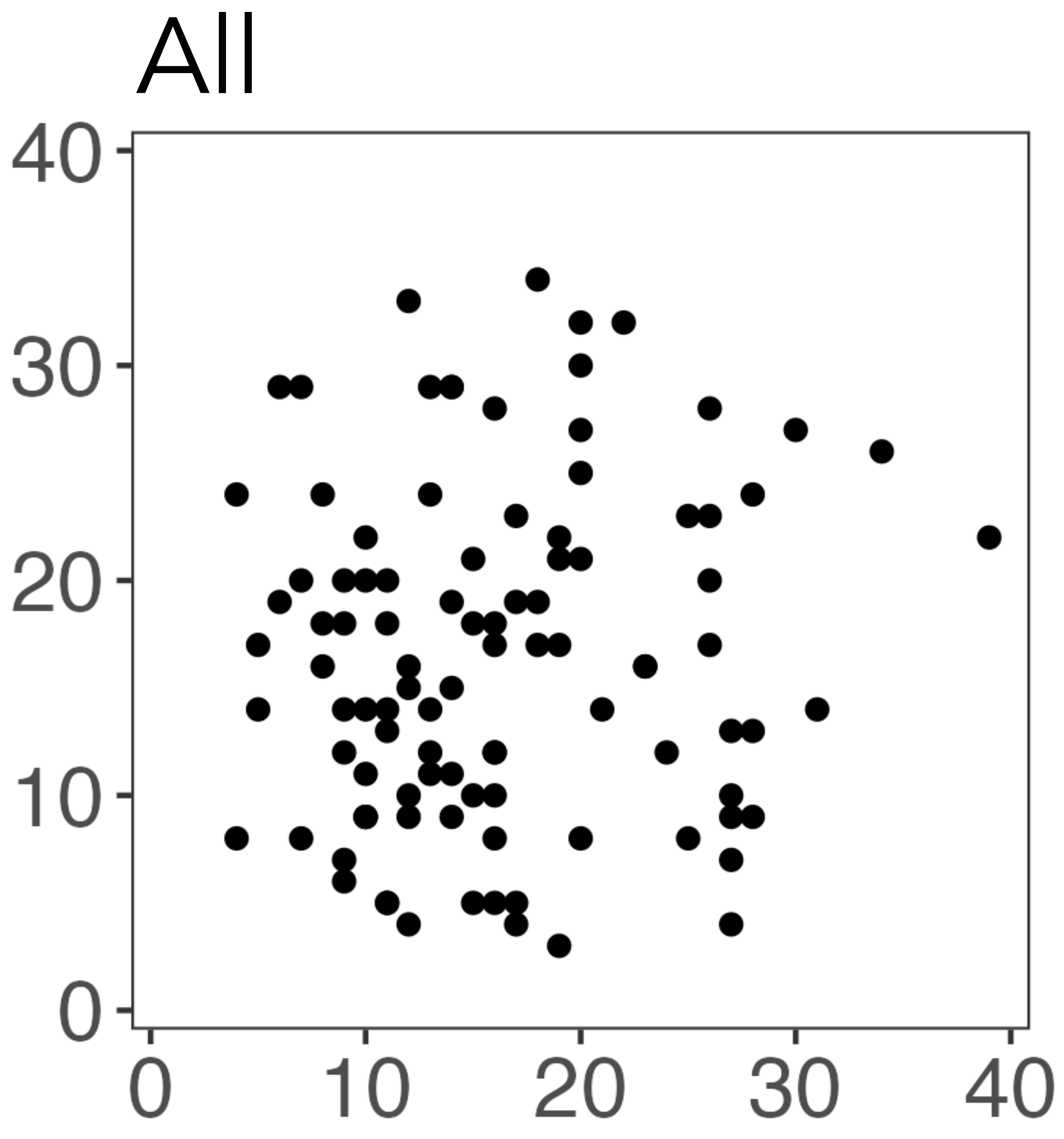
For several values of k:

Step 1: fit a model with k clusters.

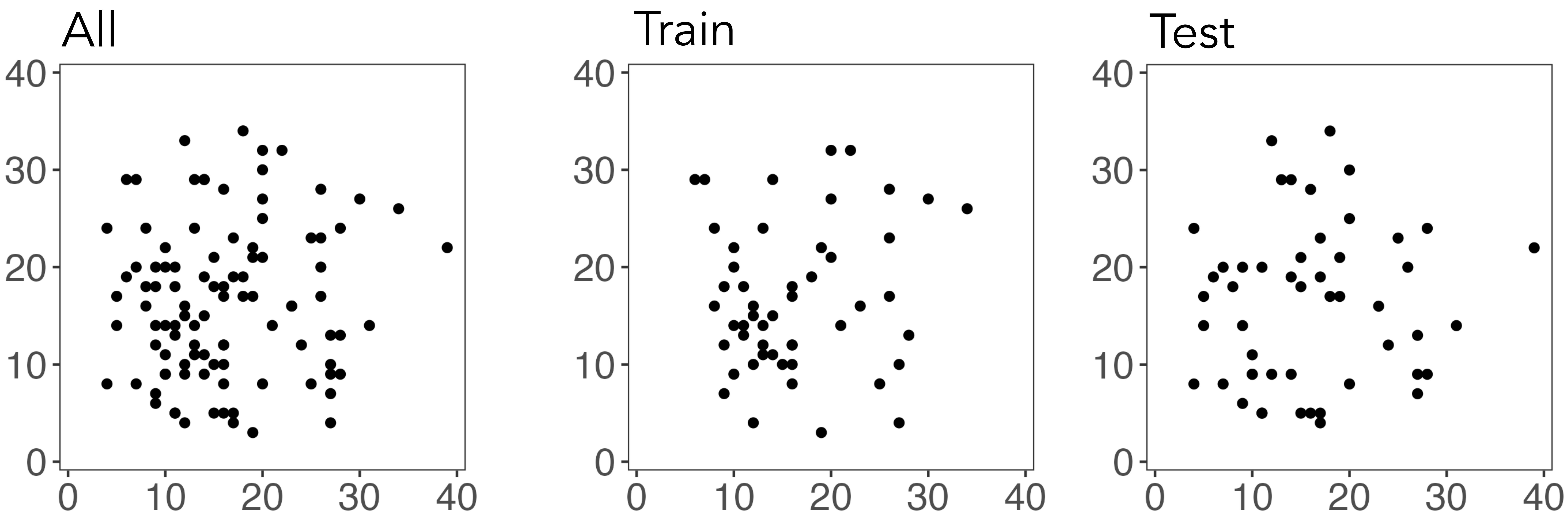
Step 2: evaluate model using a loss function.



Sample splitting cannot be used for example 2

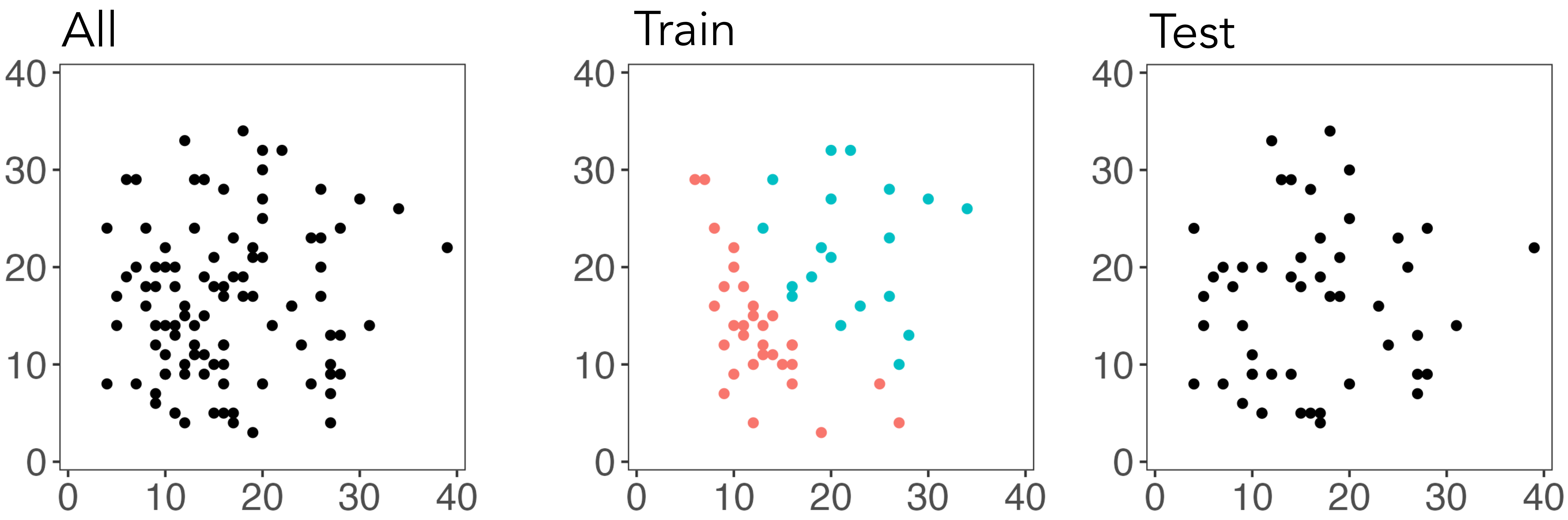


Sample splitting cannot be used for example 2



Step 1: split observations into train/test.

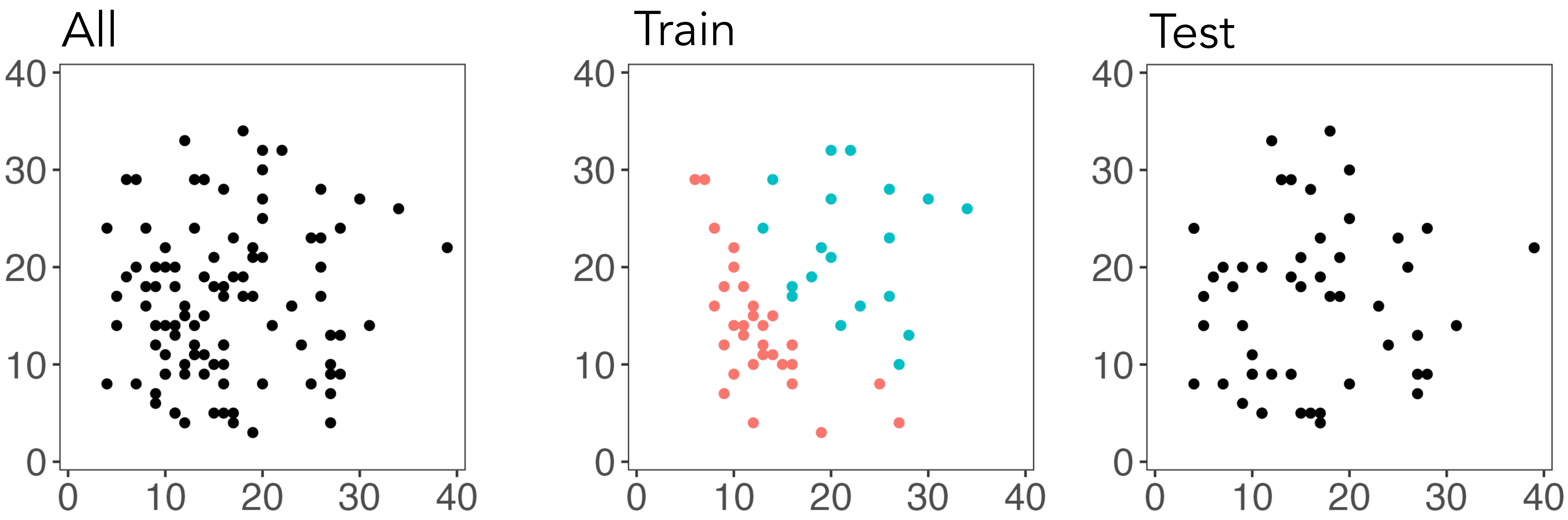
Sample splitting cannot be used for example 2



Step 1: split observations into train/test.

Step 2: cluster the training set.

Sample splitting cannot be used for example 2

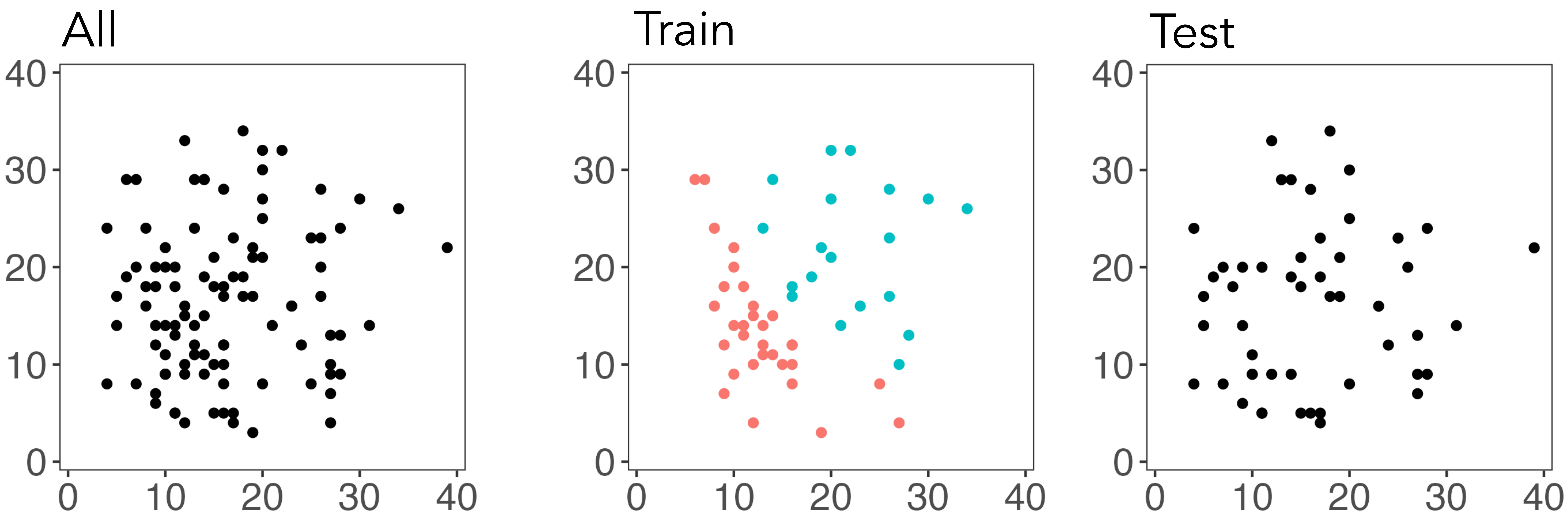


Step 1: split observations into train/test.

Step 2: cluster the training set.

Step 3: evaluate clusters using test set.

Sample splitting cannot be used for example 2



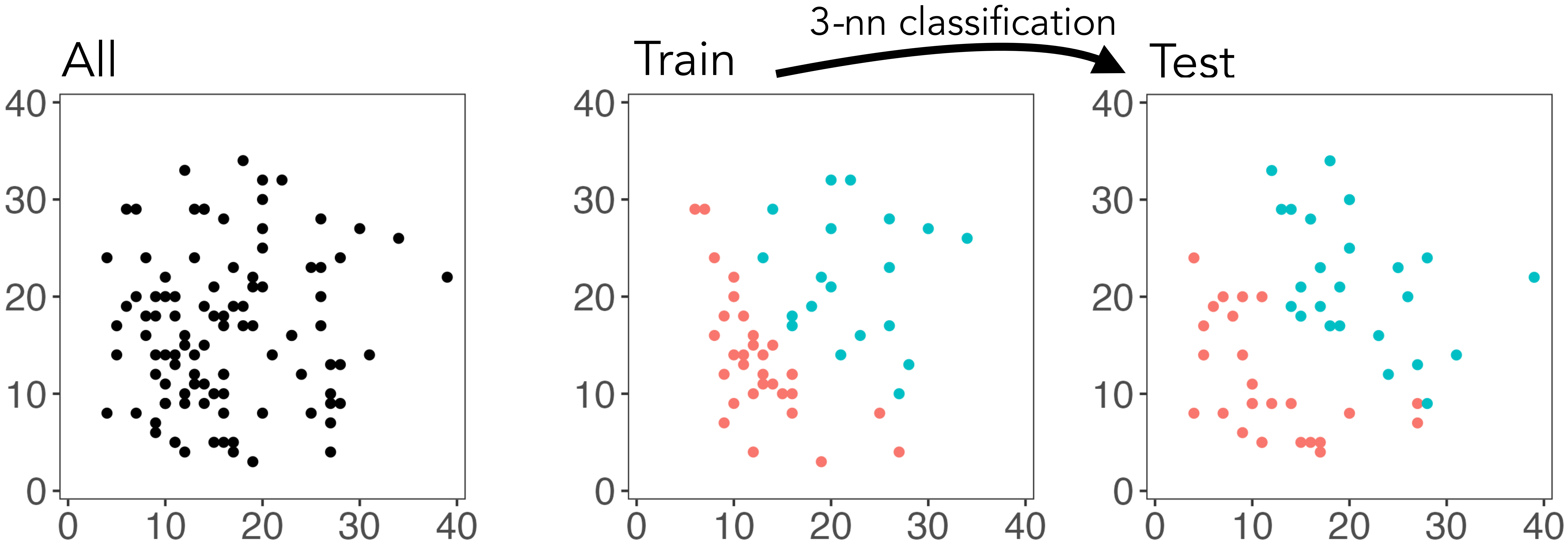
Step 1: split observations into train/test.

Step 2: cluster the training set.

Step 2.5: assign labels to observations in test set.

Step 3: evaluate clusters using test set.

Sample splitting cannot be used for example 2



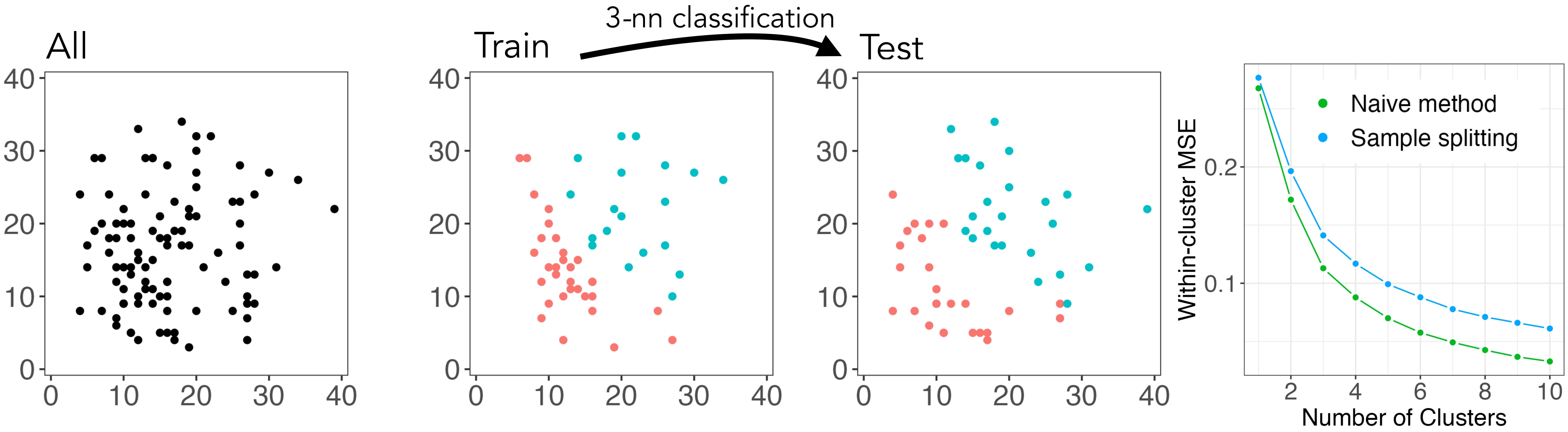
Step 1: split observations into train/test.

Step 2: cluster the training set.

Step 2.5: assign labels to observations in test set.

Step 3: evaluate clusters using test set.

Sample splitting cannot be used for example 2



Step 1: split observations into train/test.

Step 2: cluster the training set.

Step 2.5: assign labels to observations in test set.

Step 3: evaluate clusters using test set.

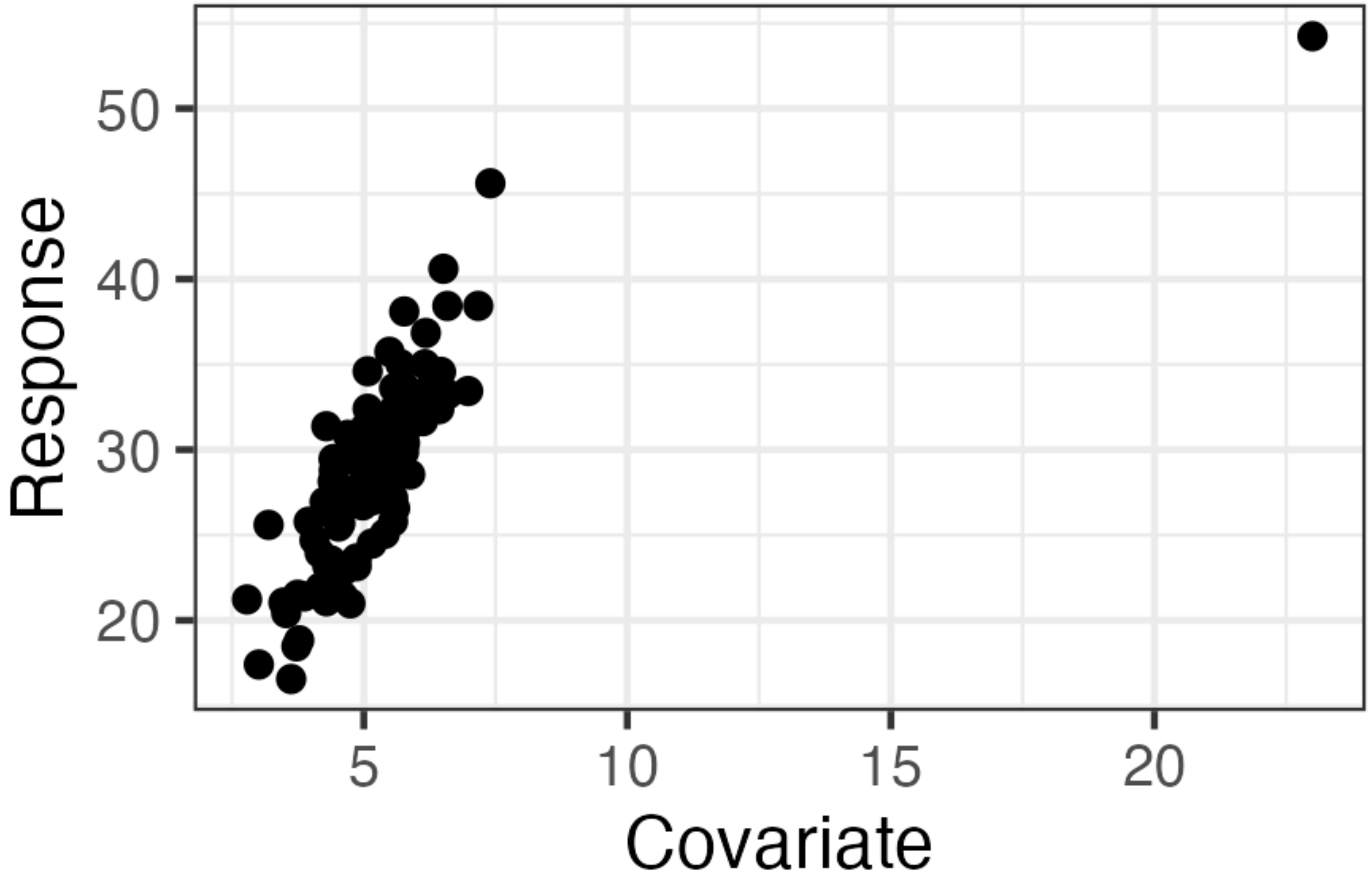
Fu and Perry, 2020 (JCGS).

Other situations in which sample splitting is not a good option

1. Fixed-X regression settings.

2. Non-IID data.

3. Data with outliers or influential points.



Outline

1. Motivation: settings where sample splitting doesn’t work
2. **Poisson thinning**
3. Data thinning
4. Application to single-cell RNA sequencing data
5. Ongoing work

Poisson thinning

X

	Feature 1	Feature 2
Obs. 1	18	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

Poisson thinning

X

	Feature 1	Feature 2
Obs. 1	18	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

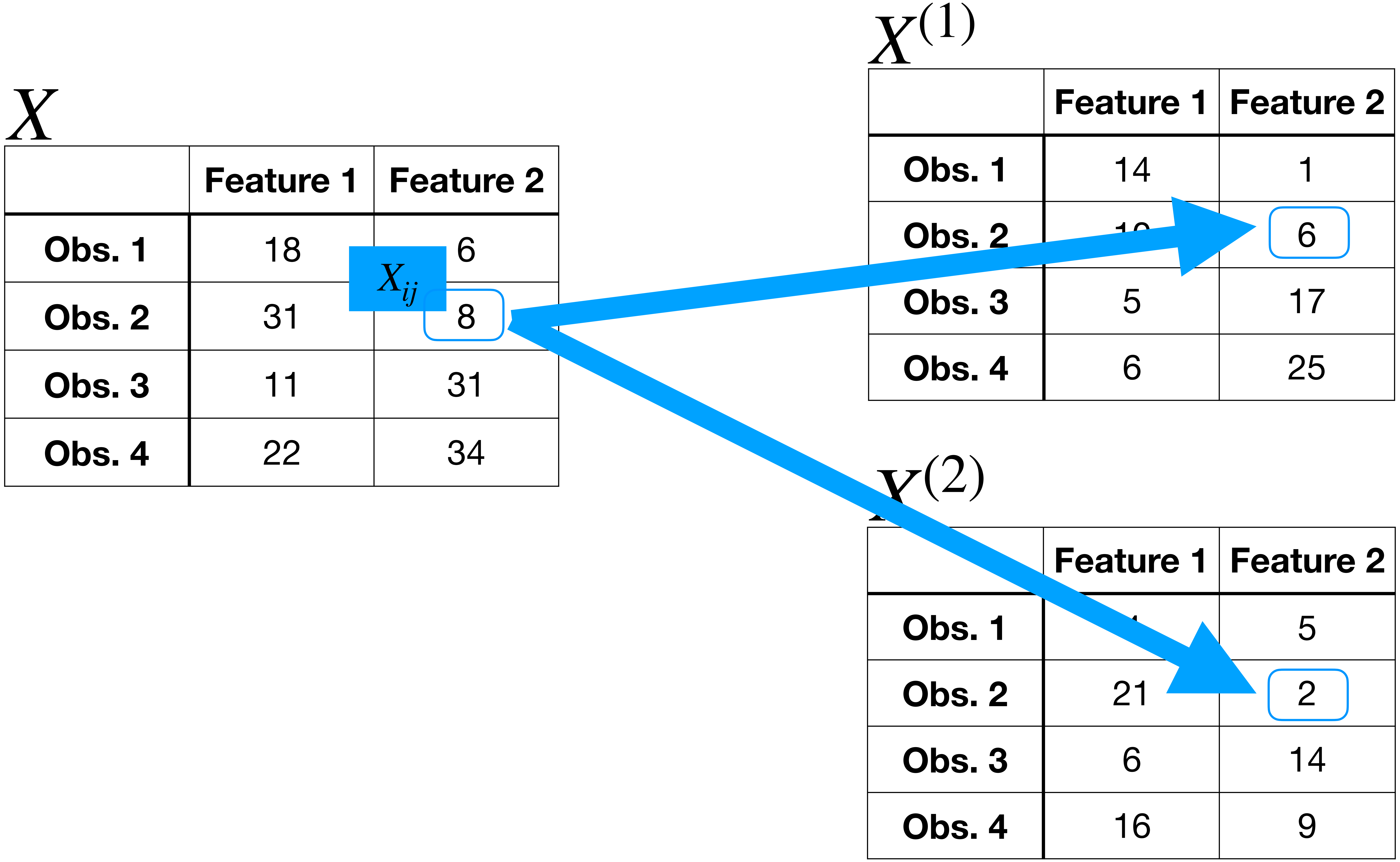
$X^{(1)}$

	Feature 1	Feature 2
Obs. 1	14	1
Obs. 2	10	6
Obs. 3	5	17
Obs. 4	6	25

$X^{(2)}$

	Feature 1	Feature 2
Obs. 1	4	5
Obs. 2	21	2
Obs. 3	6	14
Obs. 4	16	9

Poisson thinning



X

	Feature 1	Feature 2
Obs. 1	18	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

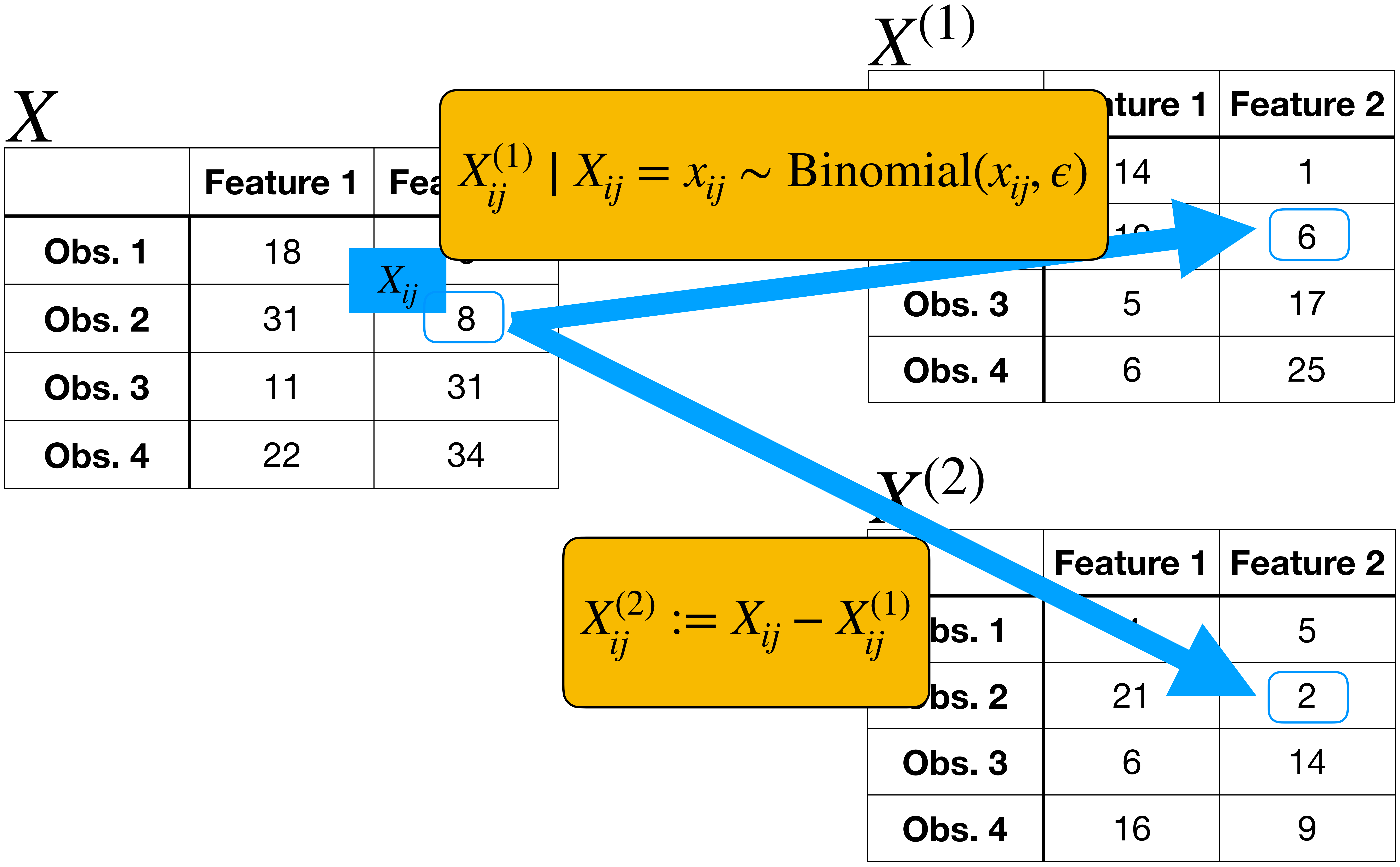
$X^{(1)}$

	Feature 1	Feature 2
Obs. 1	14	1
Obs. 2	10	6
Obs. 3	5	17
Obs. 4	6	25

$X^{(2)}$

	Feature 1	Feature 2
Obs. 1	4	5
Obs. 2	21	2
Obs. 3	6	14
Obs. 4	16	9

Poisson thinning



X

	Feature 1	Feature 2
Obs. 1	18	6
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

X_{ij}

$$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, \epsilon)$$

$X^{(1)}$

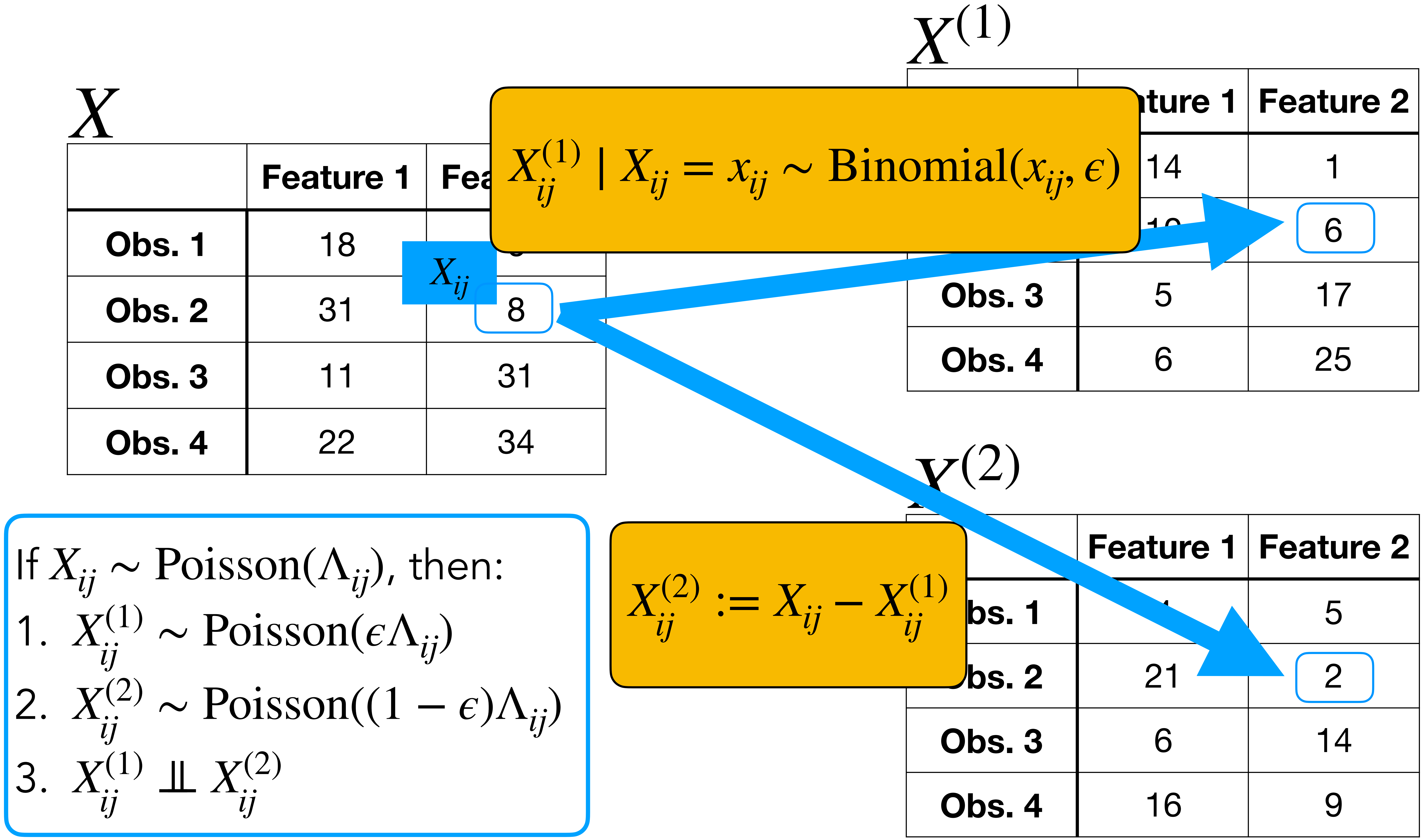
	Feature 1	Feature 2
Obs. 1	14	1
Obs. 2	10	6
Obs. 3	5	17
Obs. 4	6	25

$$X_{ij}^{(2)} := X_{ij} - X_{ij}^{(1)}$$

$X^{(2)}$

	Feature 1	Feature 2
Obs. 1	4	5
Obs. 2	21	2
Obs. 3	6	14
Obs. 4	16	9

Poisson thinning



X

	Feature 1	Fea
Obs. 1	18	[illegible]
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, \epsilon)$

$X^{(1)}$

	ture 1	Feature 2
	14	1
	[illegible]	6
Obs. 3	5	17
Obs. 4	6	25

$X_{ij}^{(2)} := X_{ij} - X_{ij}^{(1)}$

$X^{(2)}$

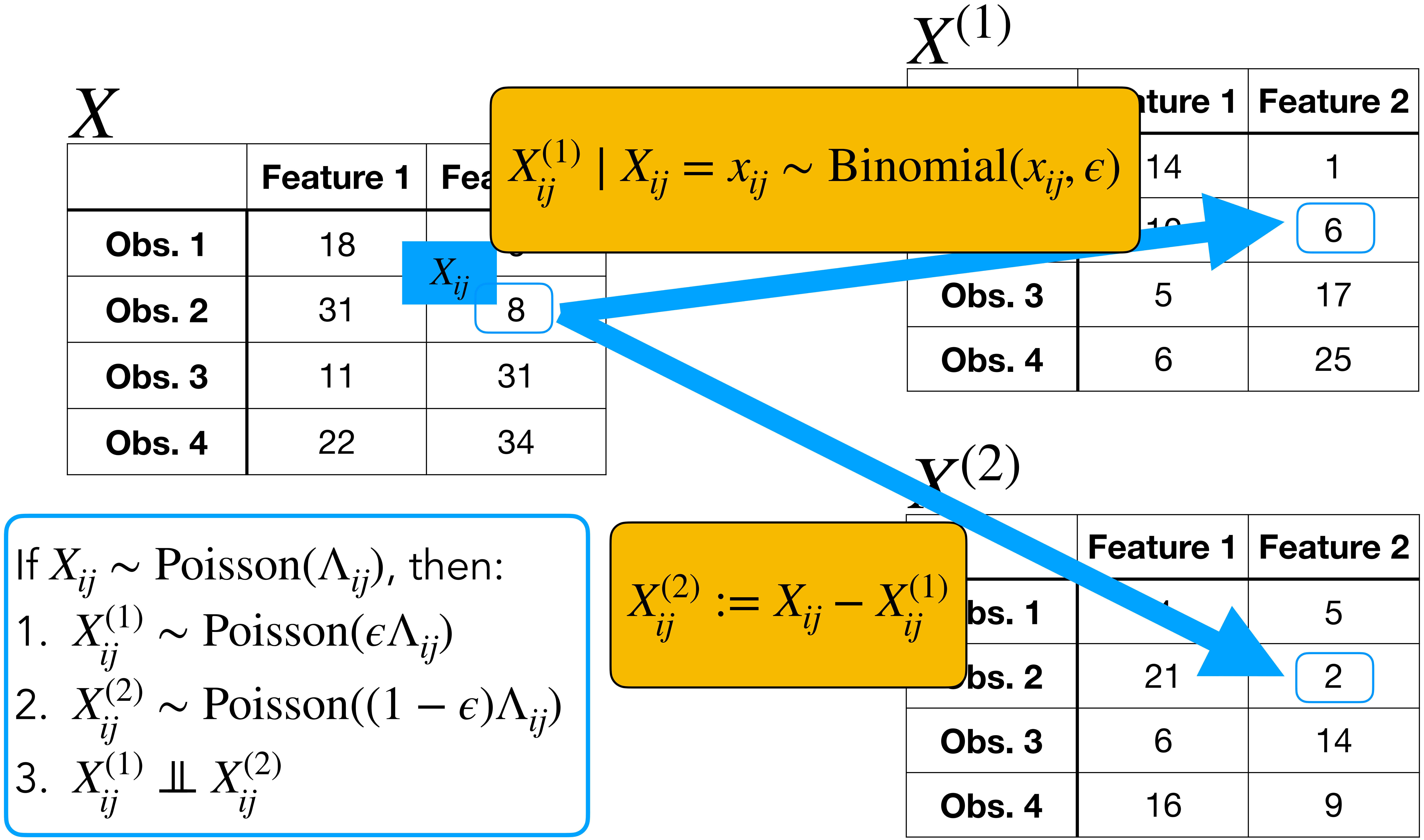
	Feature 1	Feature 2
bs. 1	[illegible]	5
bs. 2	21	2
Obs. 3	6	14
Obs. 4	16	9

If $X_{ij} \sim \text{Poisson}(\Lambda_{ij})$, then:

1. $X_{ij}^{(1)} \sim \text{Poisson}(\epsilon\Lambda_{ij})$
2. $X_{ij}^{(2)} \sim \text{Poisson}((1-\epsilon)\Lambda_{ij})$
3. $X_{ij}^{(1)} \perp\!\!\!\perp X_{ij}^{(2)}$

A very well-known result.

Poisson thinning



X

	Feature 1	Fea
Obs. 1	18	[illegible]
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

$X^{(1)}_{ij} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, \epsilon)$

$X^{(1)}$

	ture 1	Feature 2
	14	1
	[illegible]	6
Obs. 3	5	17
Obs. 4	6	25

Select hypothesis.

$X^{(2)}_{ij} := X_{ij} - X^{(1)}_{ij}$

$X^{(2)}$

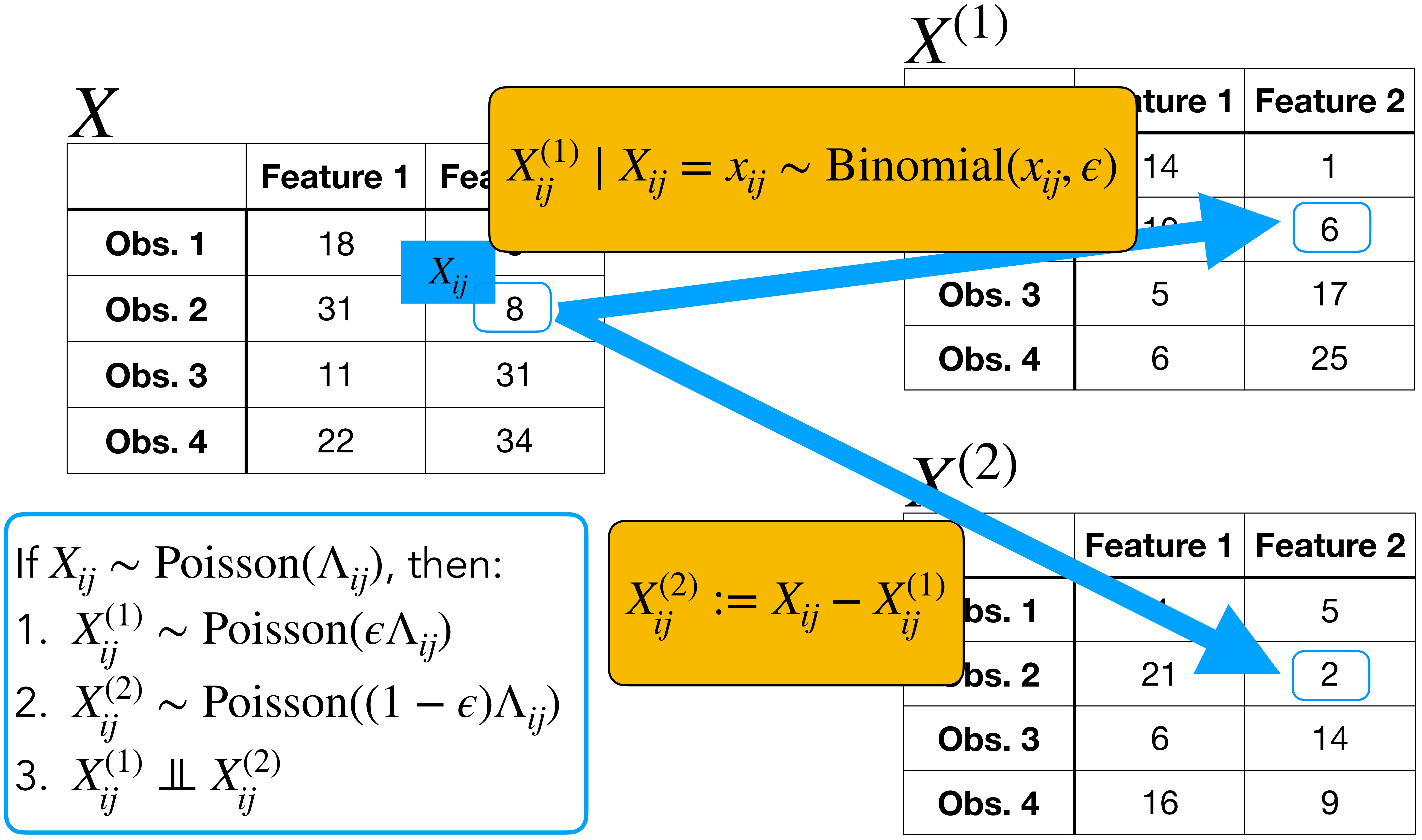
	Feature 1	Feature 2
bs. 1	[illegible]	5
bs. 2	21	2
Obs. 3	6	14
Obs. 4	16	9

If $X_{ij} \sim \text{Poisson}(\Lambda_{ij})$, then:

1. $X^{(1)}_{ij} \sim \text{Poisson}(\epsilon\Lambda_{ij})$
2. $X^{(2)}_{ij} \sim \text{Poisson}((1-\epsilon)\Lambda_{ij})$
3. $X^{(1)}_{ij} \perp\!\!\!\perp X^{(2)}_{ij}$

A very well-known result.

Poisson thinning



X

	Feature 1	Fea
Obs. 1	18	[illegible]
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

$$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, \epsilon)$$

$X^{(1)}$

[illegible]	ture 1	Feature 2
[illegible]	14	1
[illegible]	[illegible]	6
Obs. 3	5	17
Obs. 4	6	25

Select hypothesis.

$$X_{ij}^{(2)} := X_{ij} - X_{ij}^{(1)}$$

$X^{(2)}$

	Feature 1	Feature 2
bs. 1	[illegible]	5
bs. 2	21	2
Obs. 3	6	14
Obs. 4	16	9

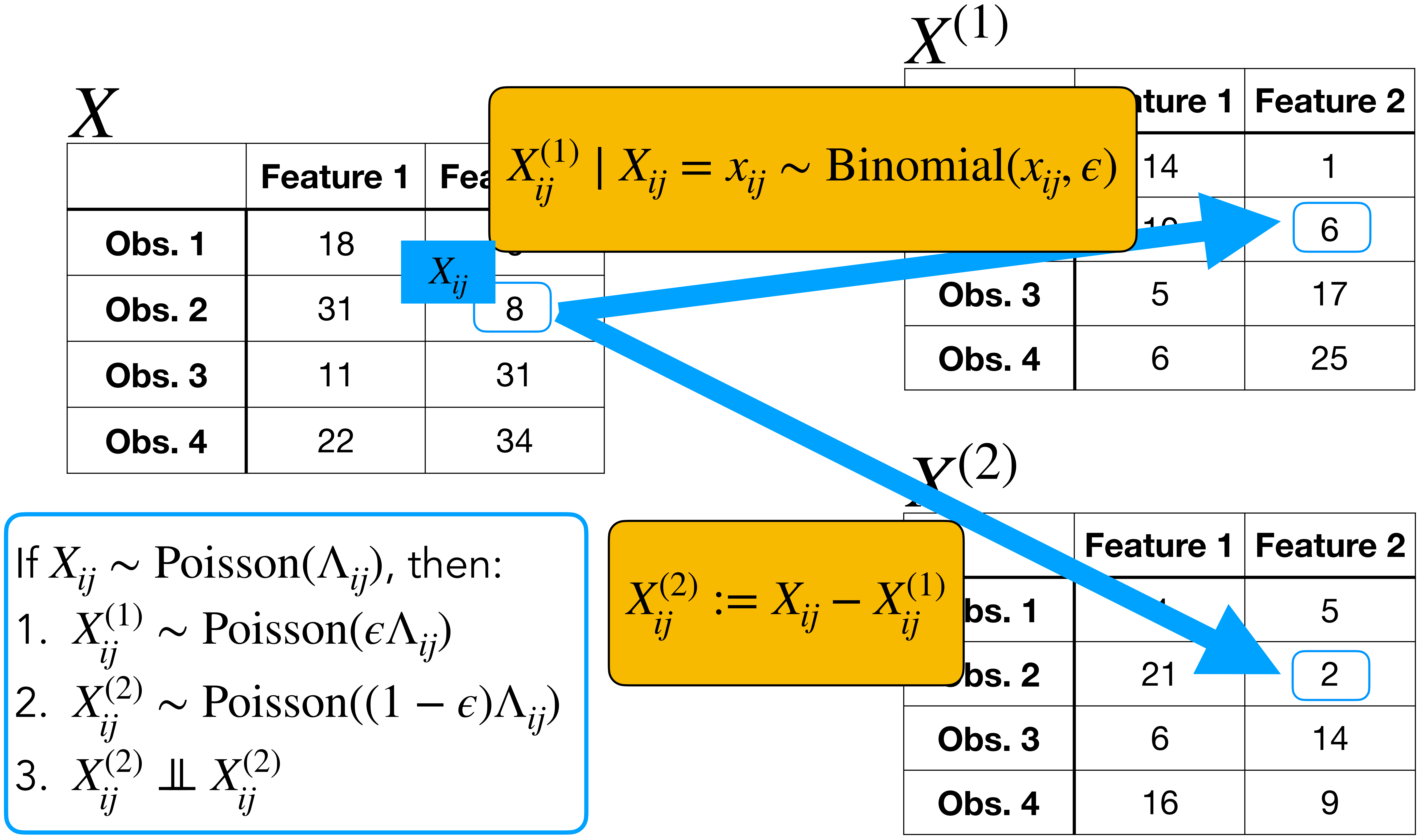
Test hypothesis.

If $X_{ij} \sim \text{Poisson}(\Lambda_{ij})$, then:

1. $X_{ij}^{(1)} \sim \text{Poisson}(\epsilon \Lambda_{ij})$
2. $X_{ij}^{(2)} \sim \text{Poisson}((1-\epsilon)\Lambda_{ij})$
3. $X_{ij}^{(1)} \perp\!\!\!\perp X_{ij}^{(2)}$

A very well-known result.

Poisson thinning



X

	Feature 1	Fea
Obs. 1	18	[illegible]
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

X_{ij}

$X^{(1)}_{ij} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, \epsilon)$

$X^{(1)}$

	ture 1	Feature 2
[illegible]	14	1
[illegible]	[illegible]	6
Obs. 3	5	17
Obs. 4	6	25

Fit model.

$X^{(2)}_{ij} := X_{ij} - X^{(1)}_{ij}$

$X^{(2)}$

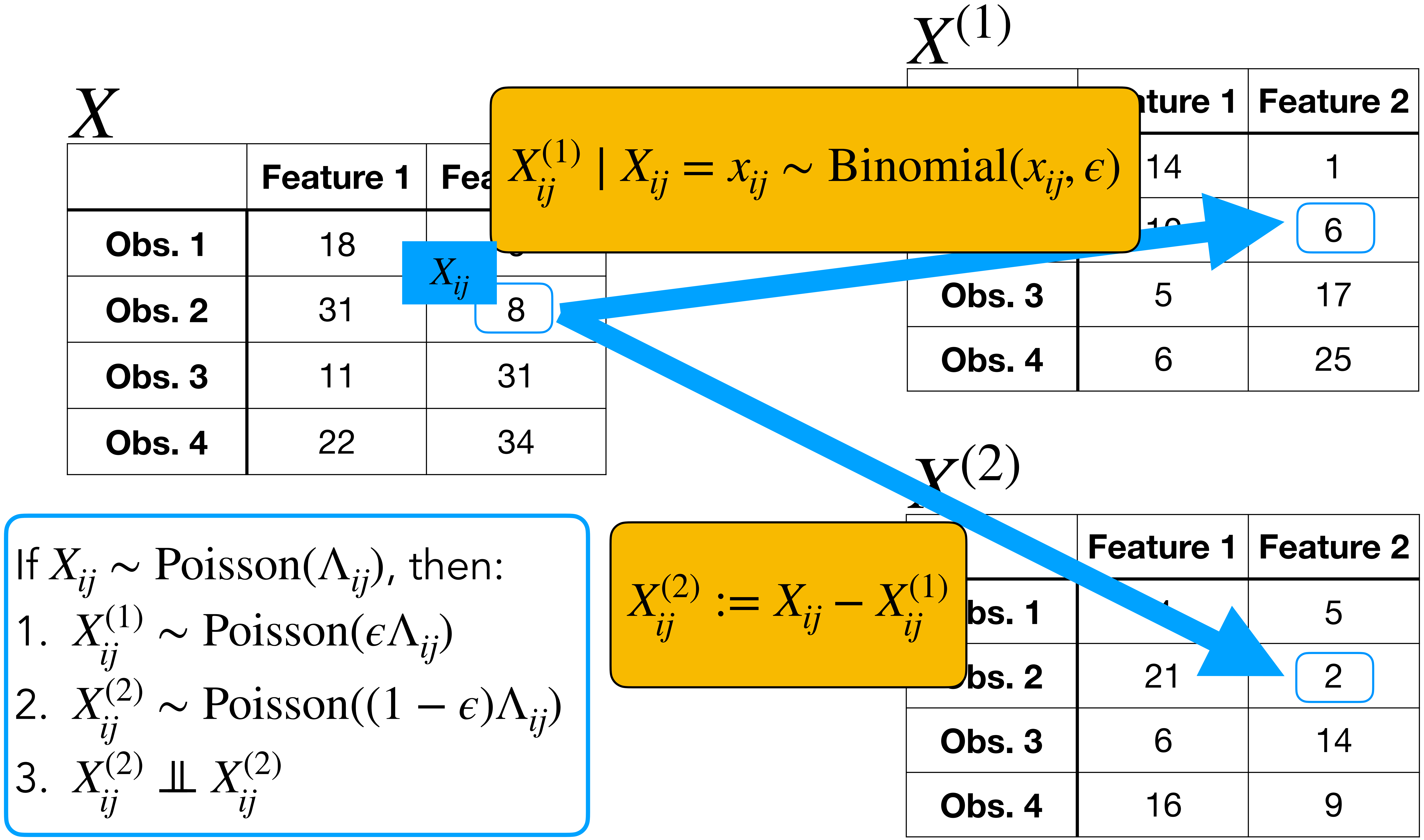
	Feature 1	Feature 2
bs. 1	[illegible]	5
bs. 2	21	2
Obs. 3	6	14
Obs. 4	16	9

If $X_{ij} \sim \text{Poisson}(\Lambda_{ij})$, then:

1. $X^{(1)}_{ij} \sim \text{Poisson}(\epsilon\Lambda_{ij})$
2. $X^{(2)}_{ij} \sim \text{Poisson}((1-\epsilon)\Lambda_{ij})$
3. $X^{(2)}_{ij} \perp\!\!\!\perp X^{(2)}_{ij}$

A very well-known result.

Poisson thinning



X

	Feature 1	Fea
Obs. 1	18	
Obs. 2	31	8
Obs. 3	11	31
Obs. 4	22	34

X_{ij}

$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, \epsilon)$

$X^{(1)}$

	ture 1	Feature 2
	14	1
		6
Obs. 3	5	17
Obs. 4	6	25

Fit model.

$X_{ij}^{(2)} := X_{ij} - X_{ij}^{(1)}$

$X^{(2)}$

	Feature 1	Feature 2
bs. 1		5
bs. 2	21	2
Obs. 3	6	14
Obs. 4	16	9

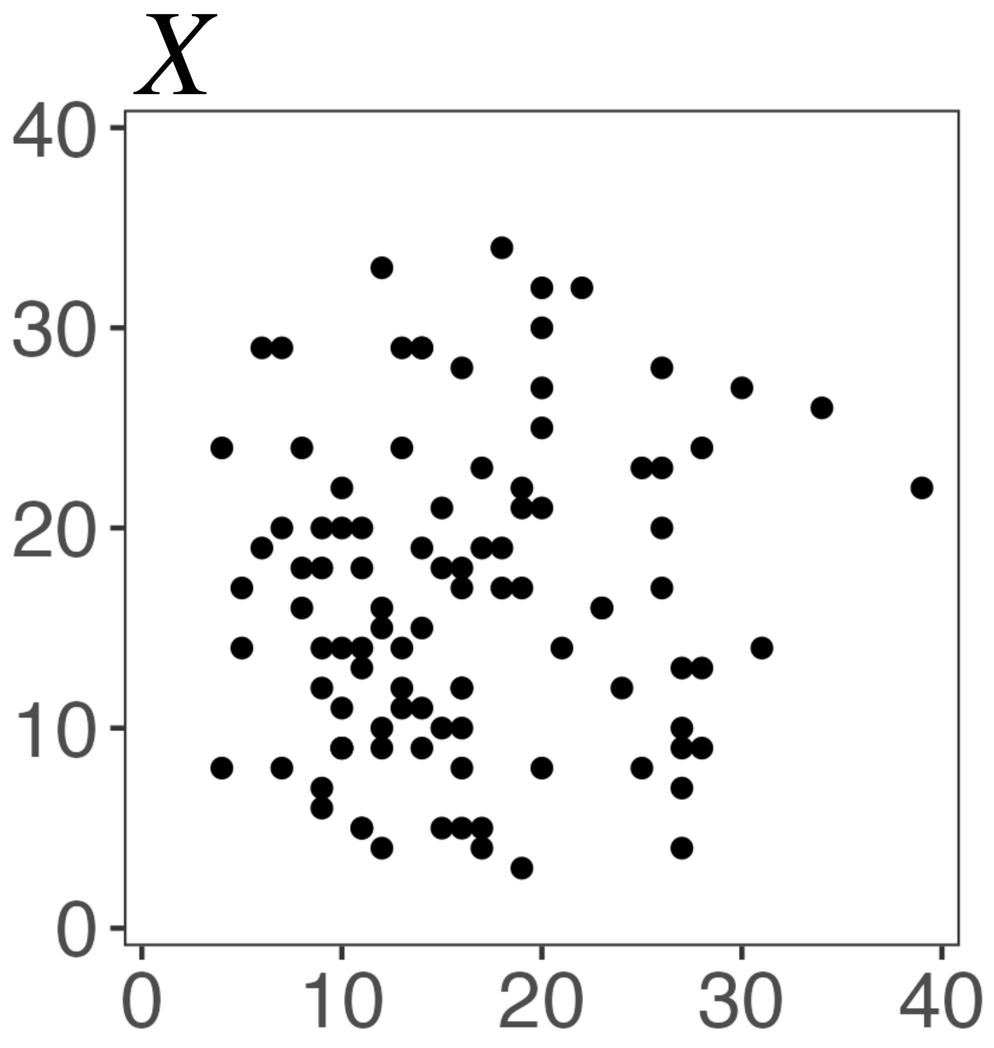
Evaluate model.

If $X_{ij} \sim \text{Poisson}(\Lambda_{ij})$, then:

1. $X_{ij}^{(1)} \sim \text{Poisson}(\epsilon\Lambda_{ij})$
2. $X_{ij}^{(2)} \sim \text{Poisson}((1-\epsilon)\Lambda_{ij})$
3. $X_{ij}^{(2)} \perp\!\!\!\perp X_{ij}^{(2)}$

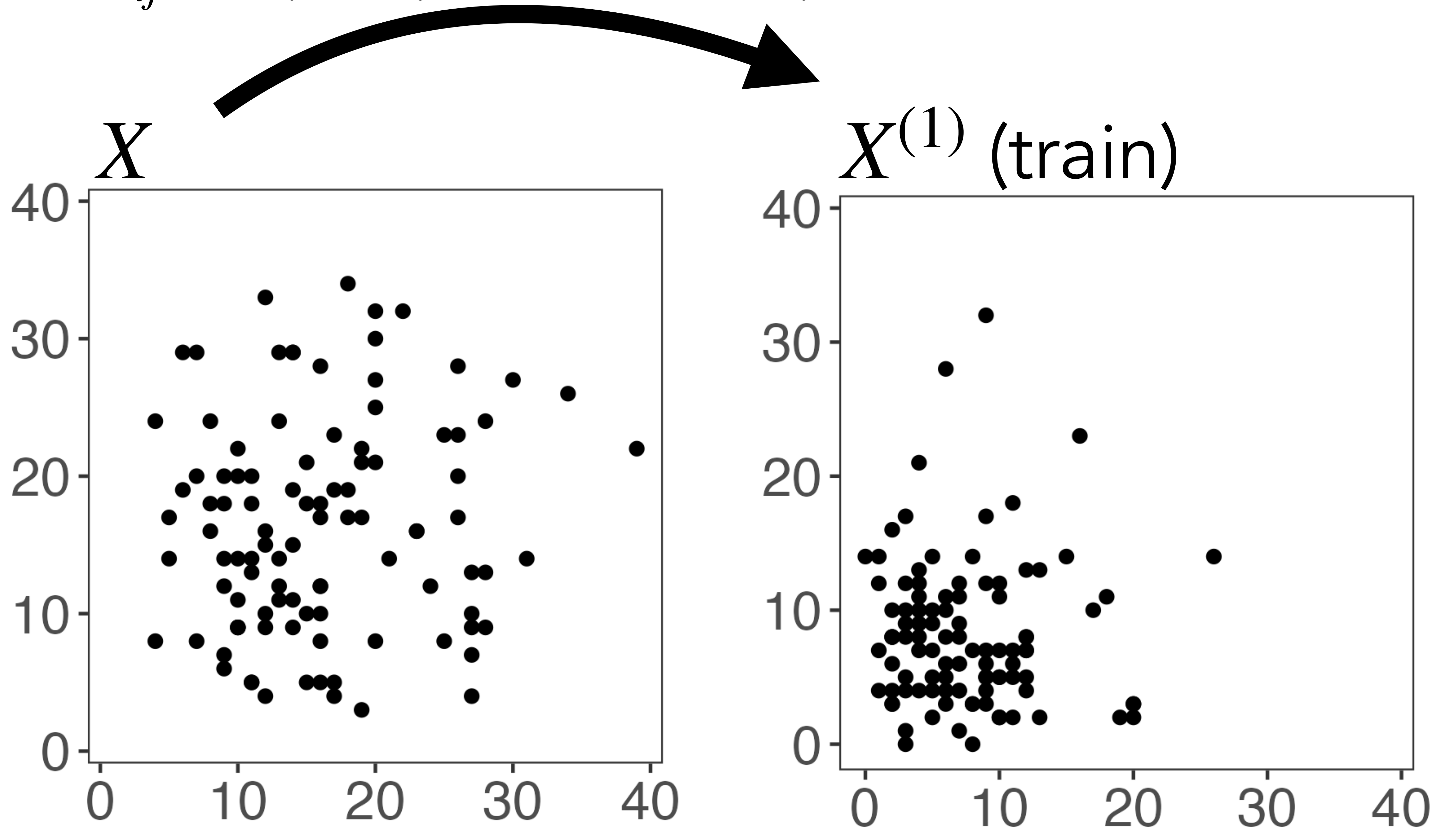
A very well-known result.

Thinning avoids the pitfall of sample splitting on our motivating examples



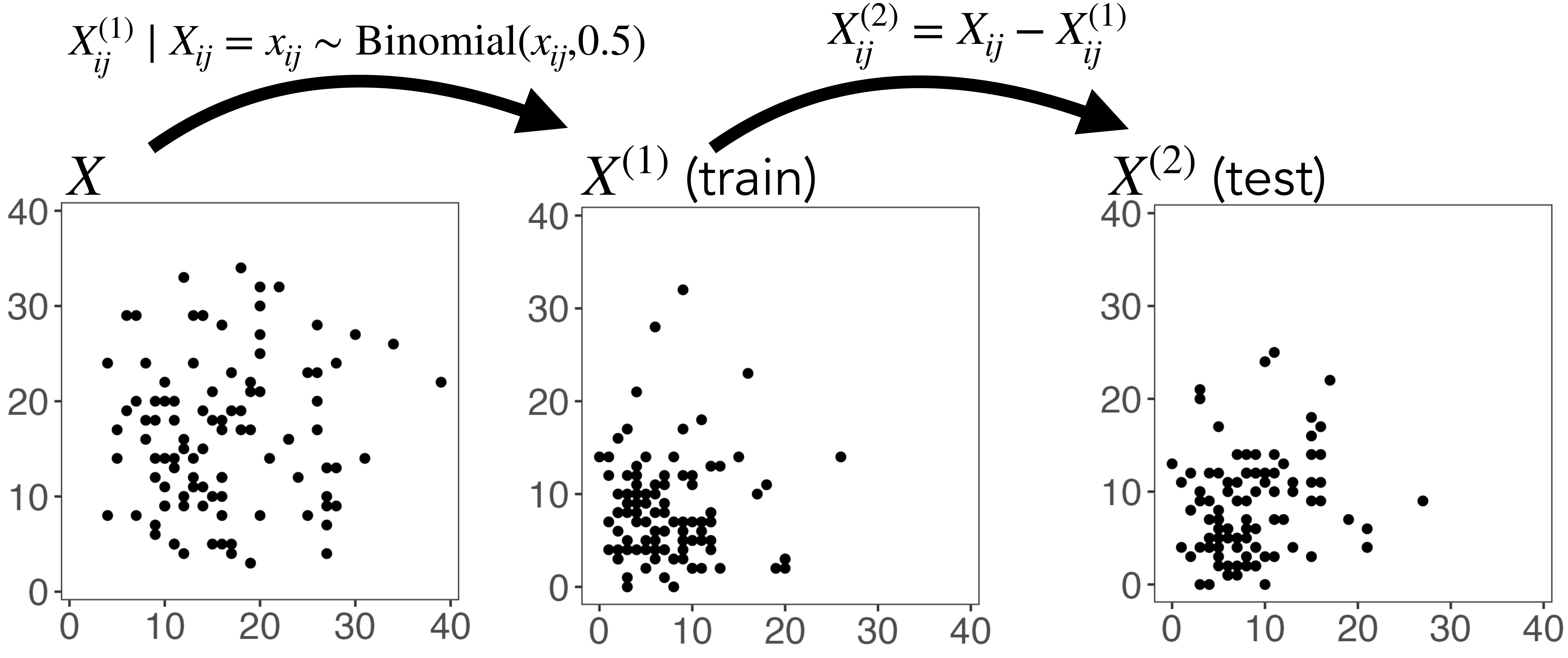
Thinning avoids the pitfall of sample splitting on our motivating examples

$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$



Step 1: thin observations into train/test.

Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

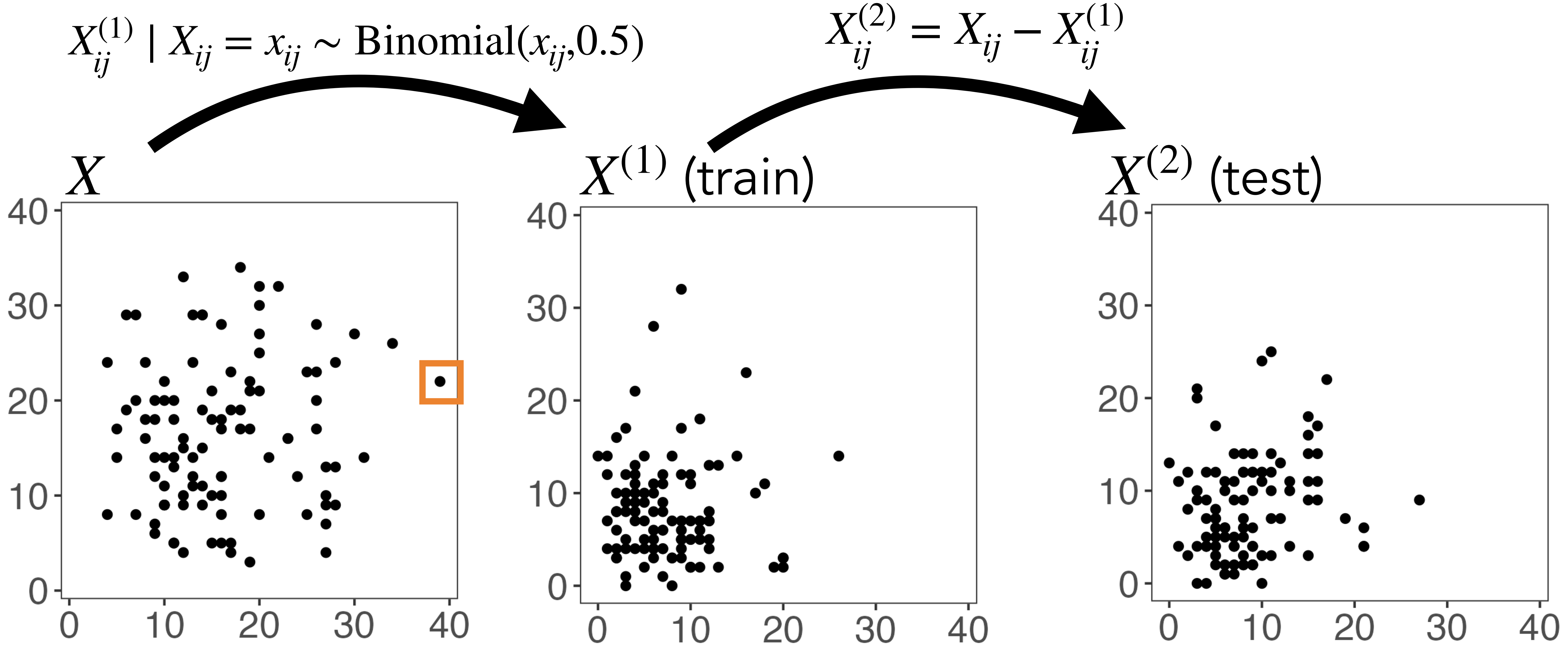
X

$X^{(1)}$ (train)

$X^{(2)}$ (test)

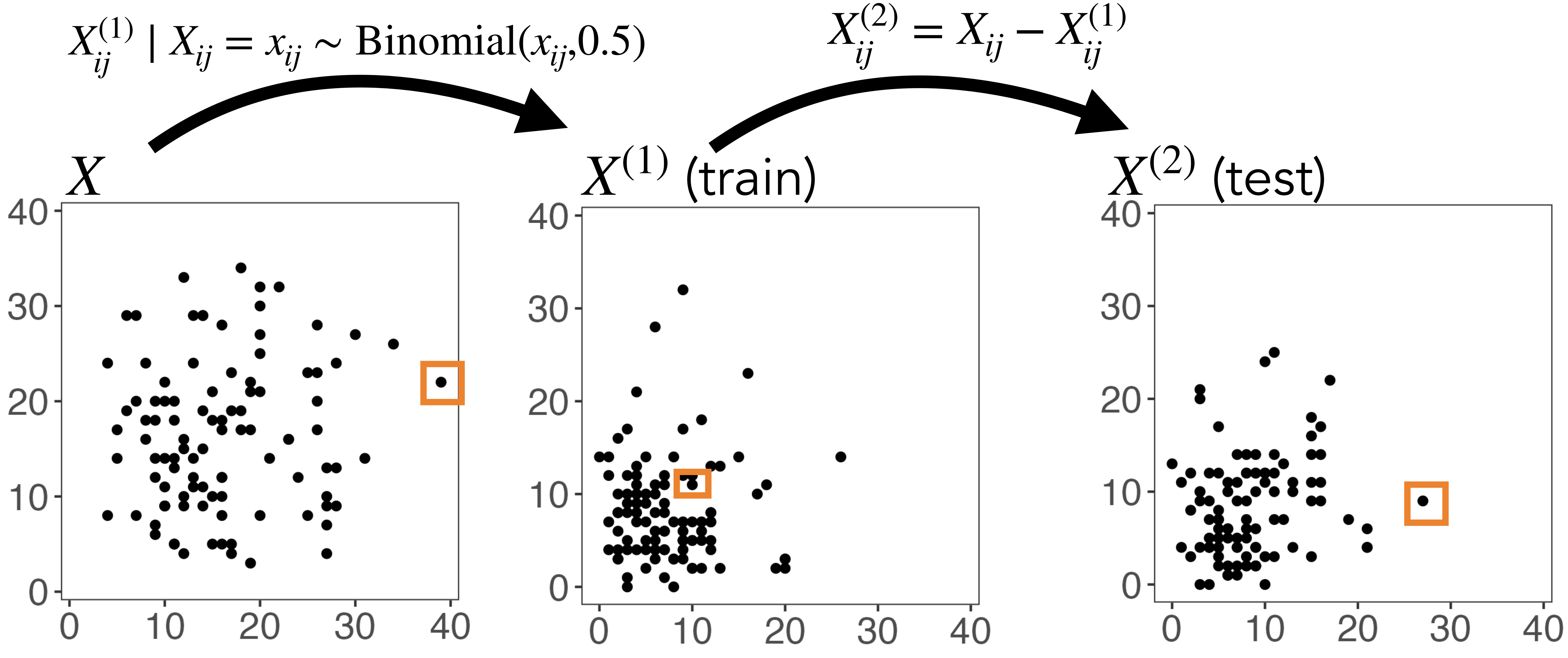
Step 1: thin observations into train/test.

Thinning avoids the pitfall of sample splitting on our motivating examples



Step 1: thin observations into train/test.

Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

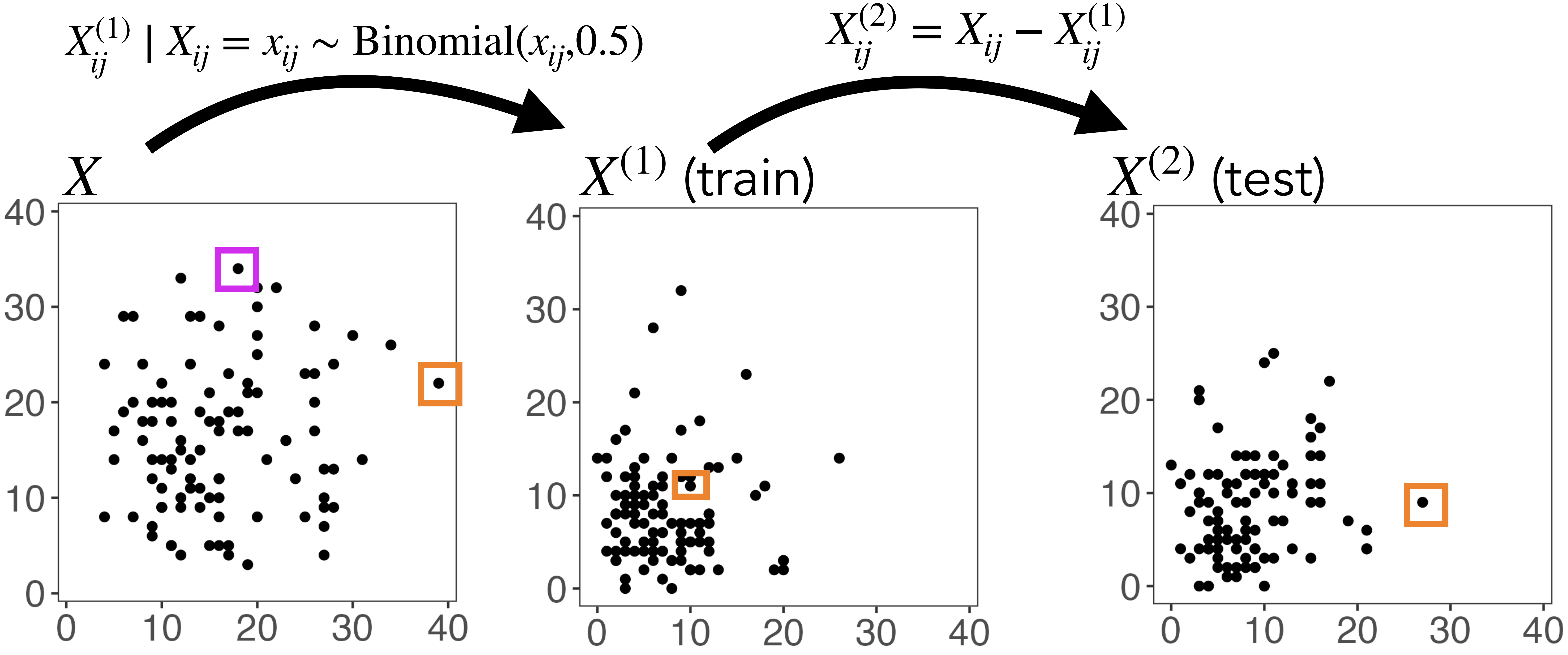
X

$X^{(1)}$ (train)

$X^{(2)}$ (test)

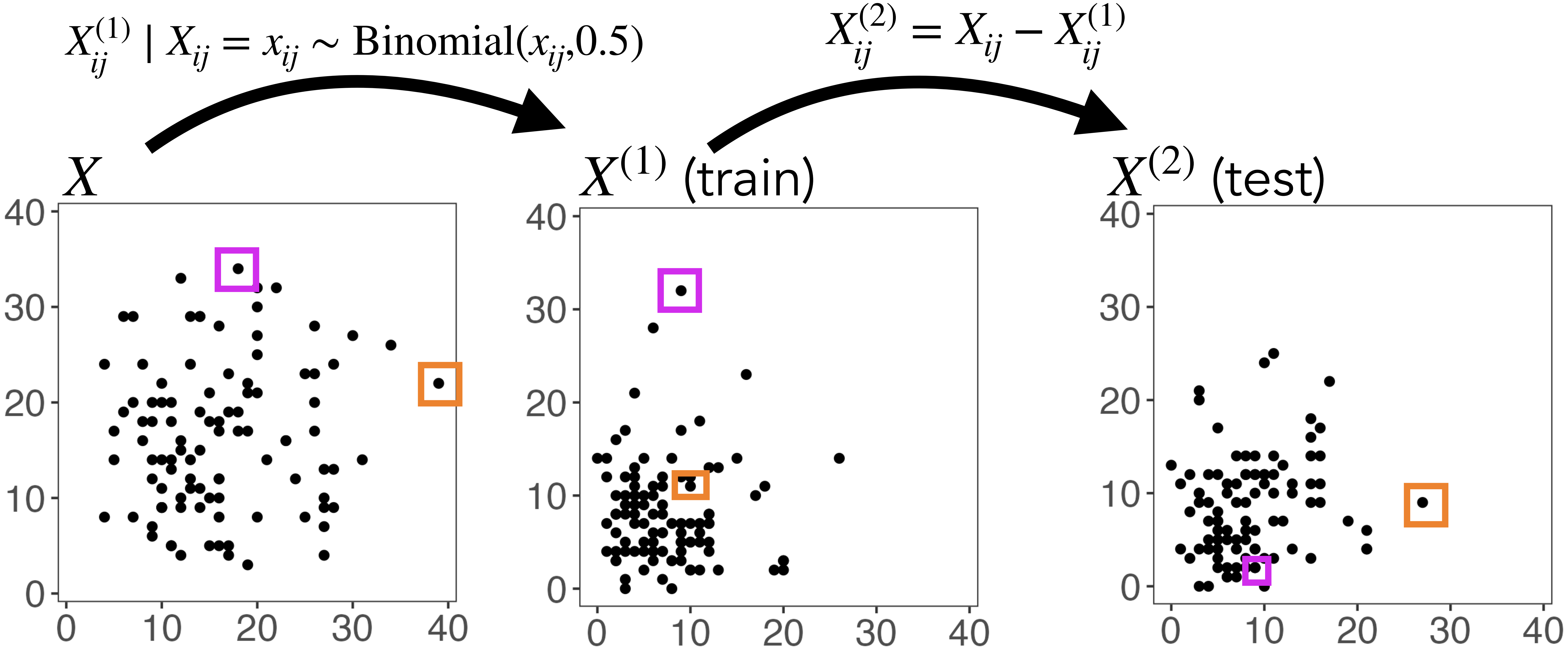
Step 1: thin observations into train/test.

Thinning avoids the pitfall of sample splitting on our motivating examples



Step 1: thin observations into train/test.

Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

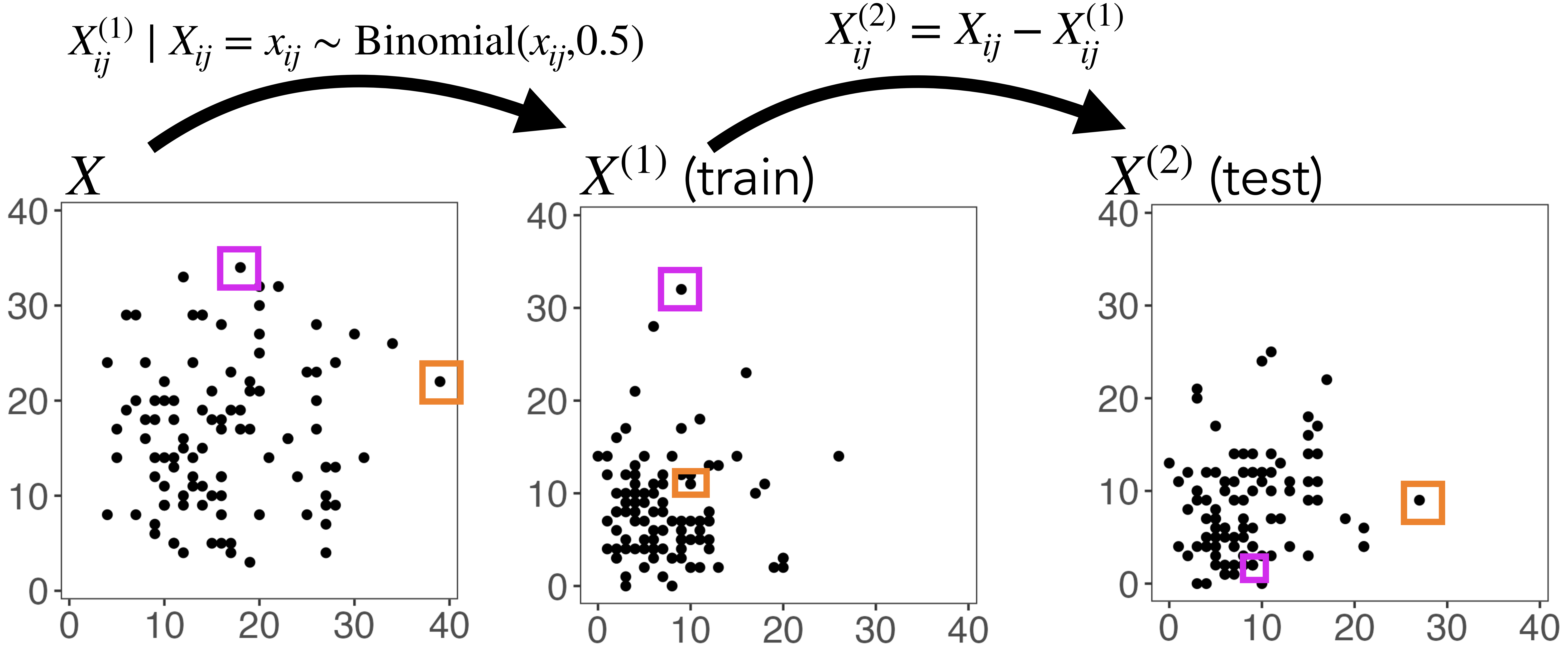
X

$X^{(1)}$ (train)

$X^{(2)}$ (test)

Step 1: thin observations into train/test.

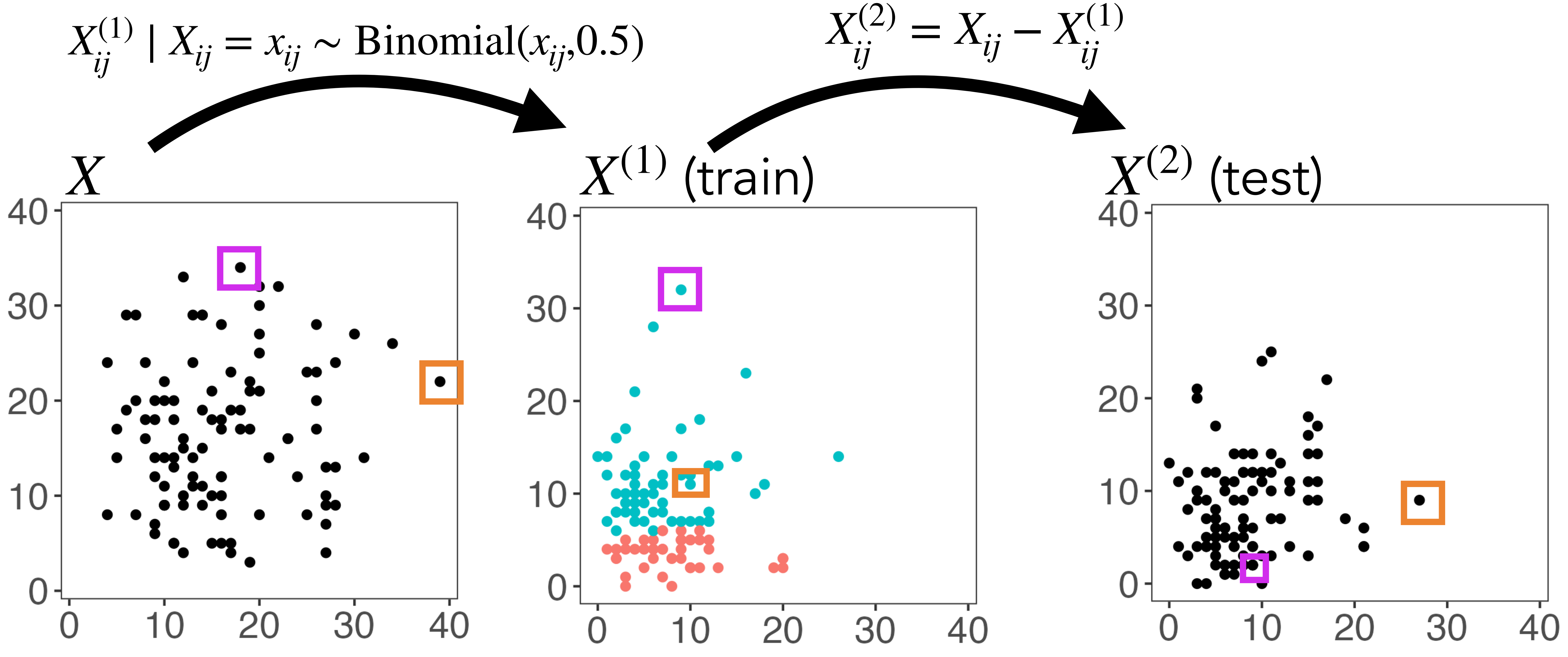
Thinning avoids the pitfall of sample splitting on our motivating examples



Step 1: thin observations into train/test.

Step 2: cluster the training set.

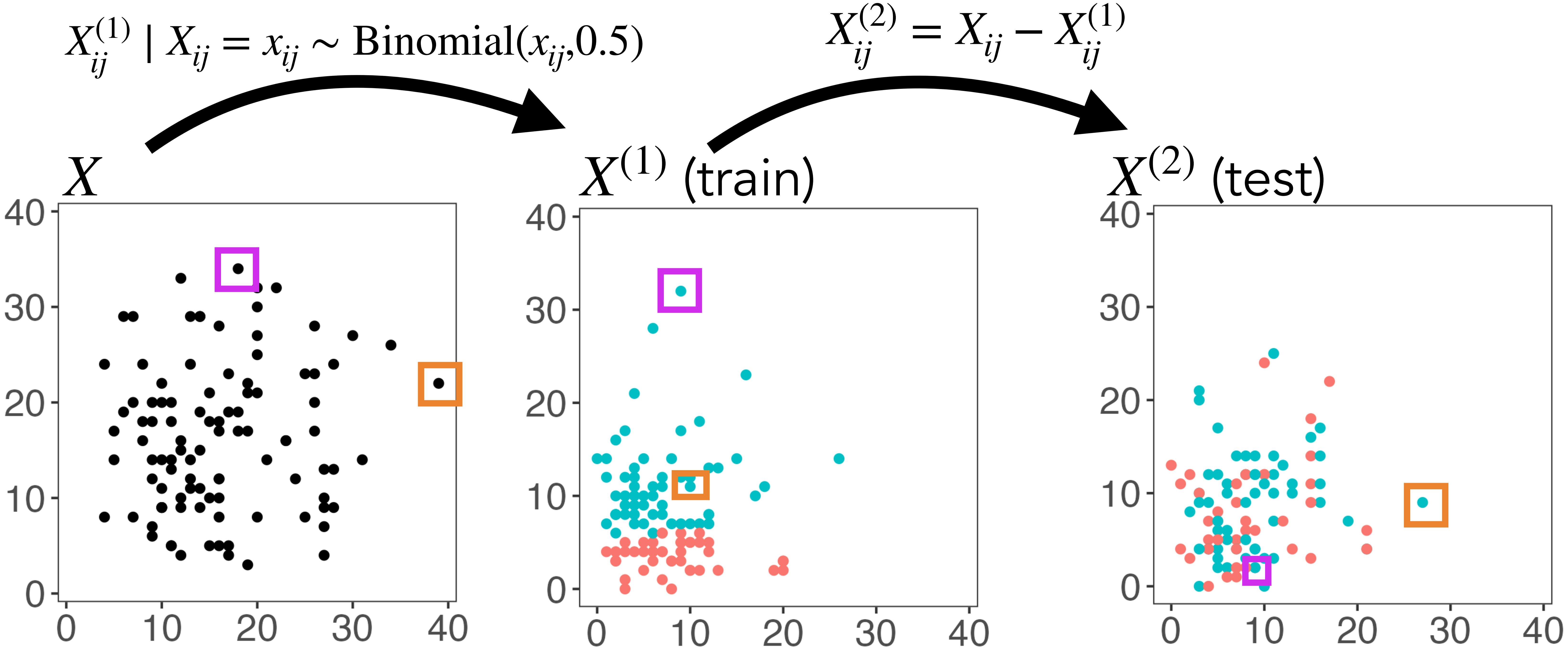
Thinning avoids the pitfall of sample splitting on our motivating examples



Step 1: thin observations into train/test.

Step 2: cluster the training set.

Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

X

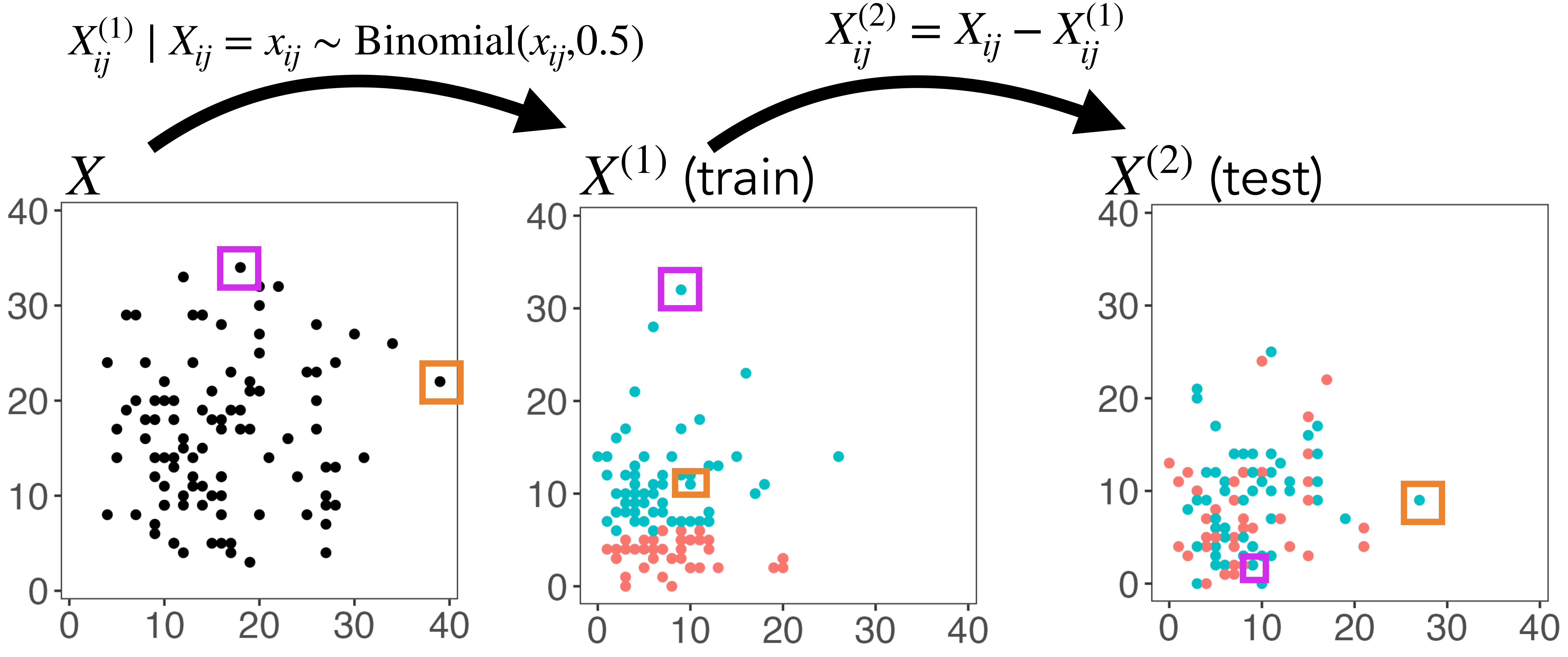
$X^{(1)}$ (train)

$X^{(2)}$ (test)

Step 1: thin observations into train/test.

Step 2: cluster the training set.

Thinning avoids the pitfall of sample splitting on our motivating examples

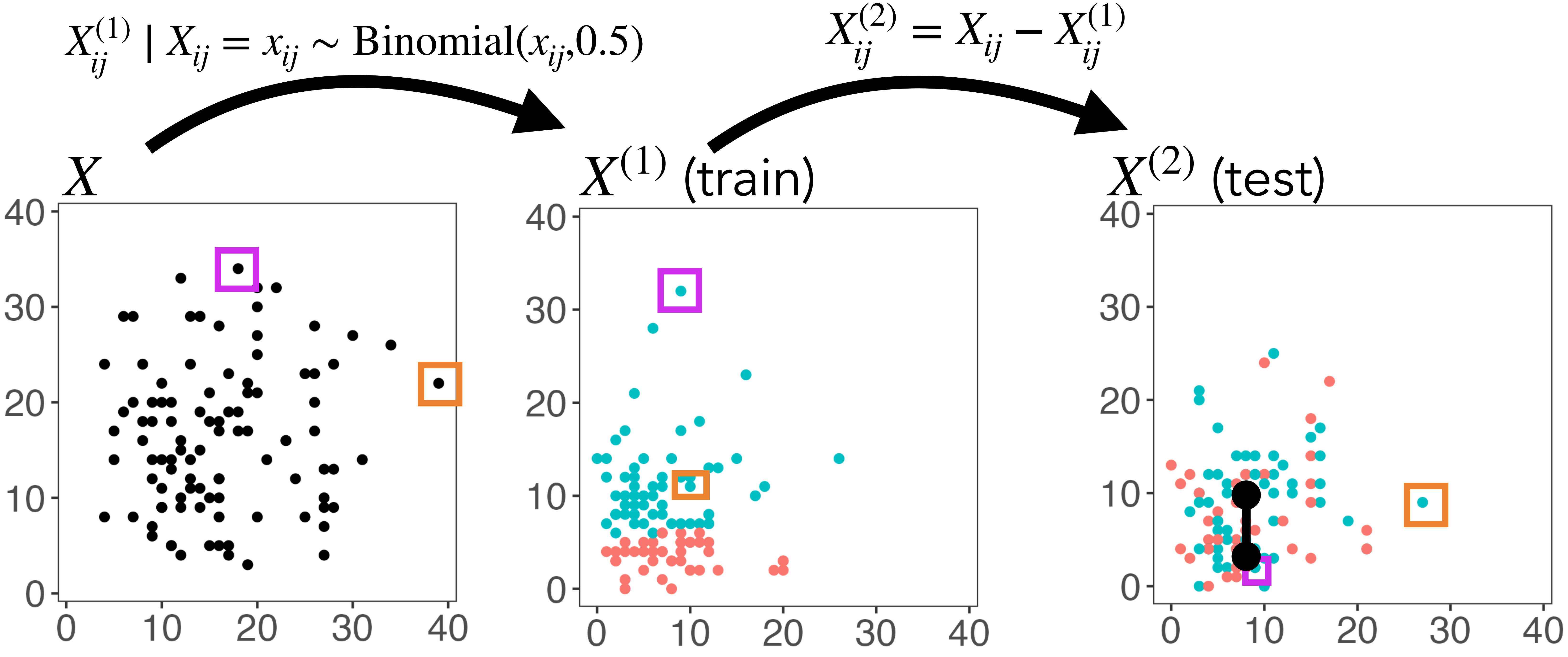


Step 1: thin observations into train/test.

Step 2: cluster the training set.

Step 3: test for difference in means or evaluate clusters on test set.

Thinning avoids the pitfall of sample splitting on our motivating examples

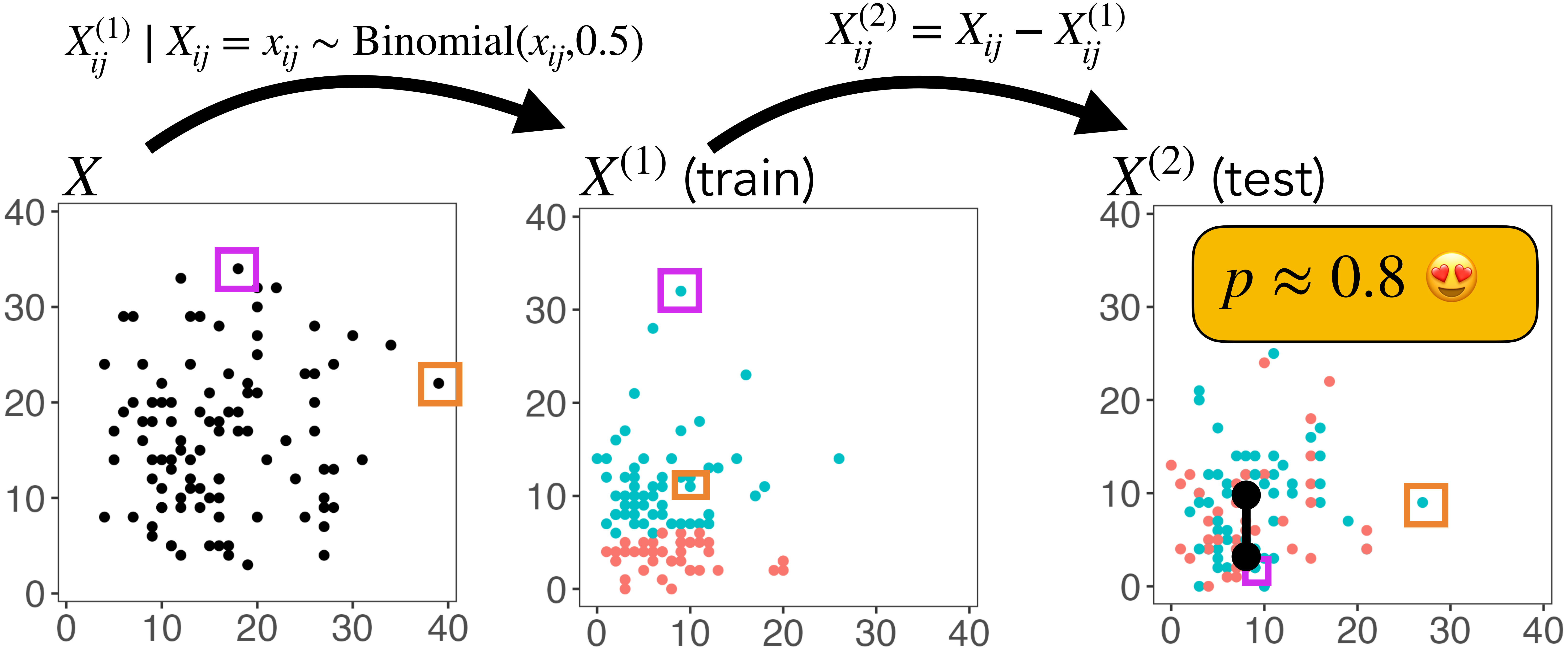


Step 1: thin observations into train/test.

Step 2: cluster the training set.

Step 3: test for difference in means or evaluate clusters on test set.

Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

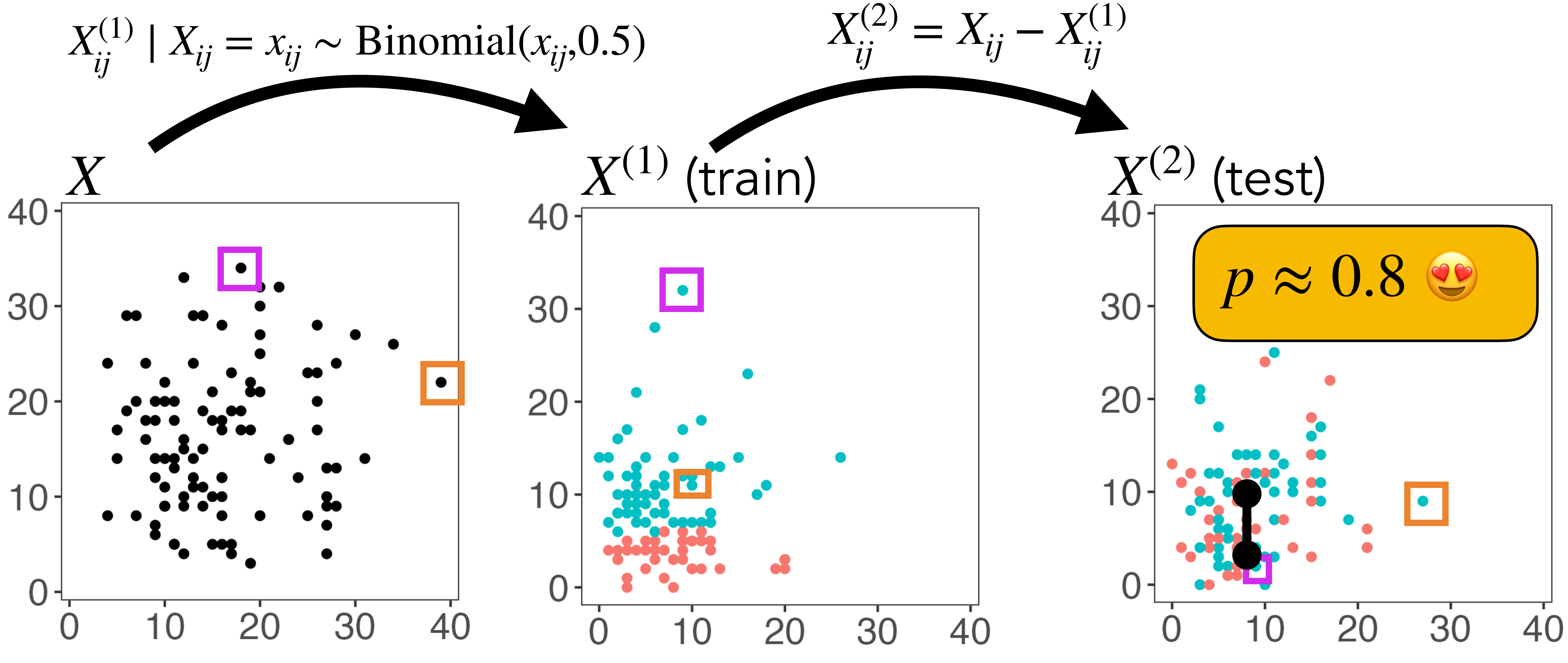
$p \approx 0.8$ 😍

Step 1: thin observations into train/test.

Step 2: cluster the training set.

Step 3: test for difference in means or evaluate clusters on test set.

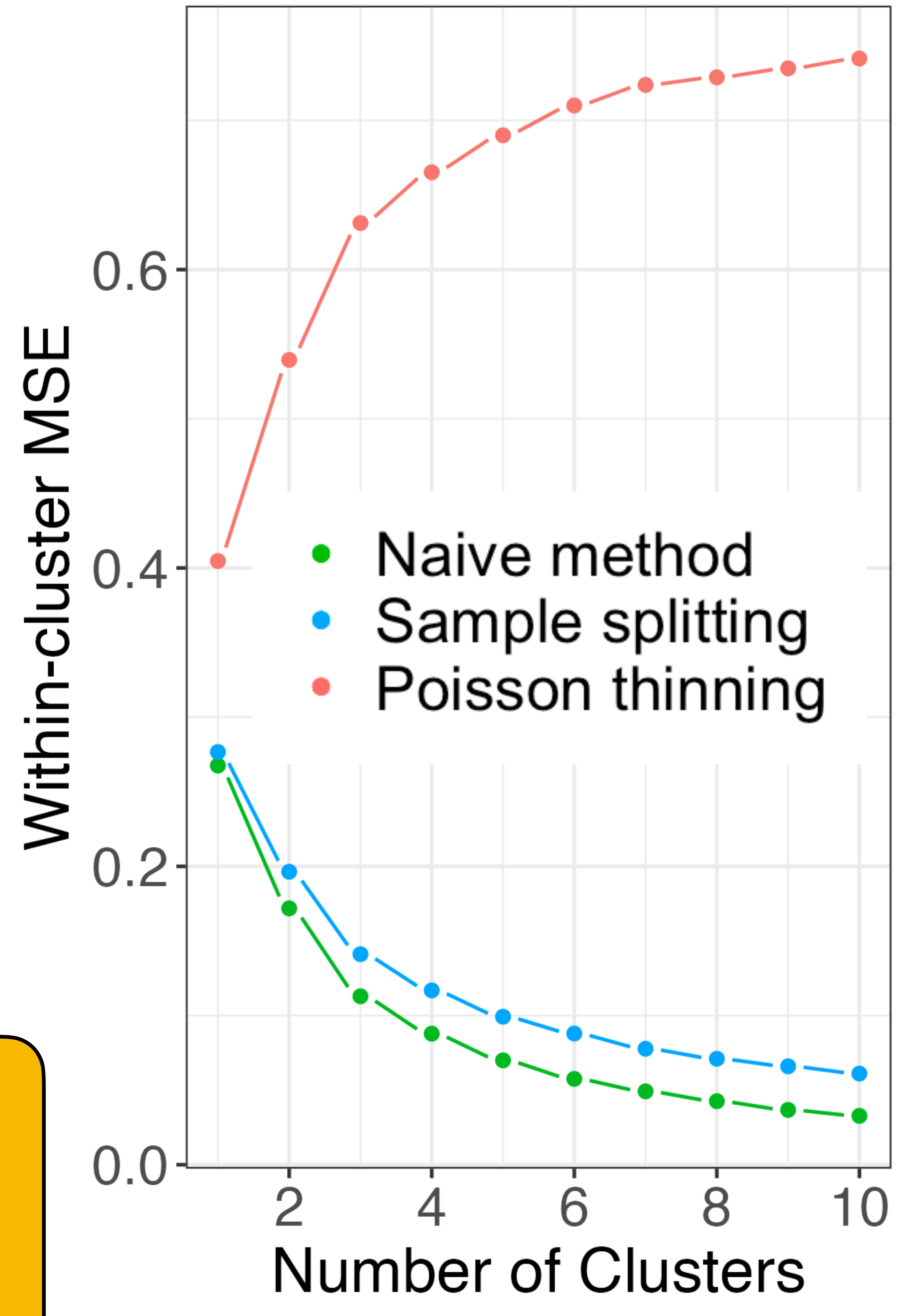
Thinning avoids the pitfall of sample splitting on our motivating examples



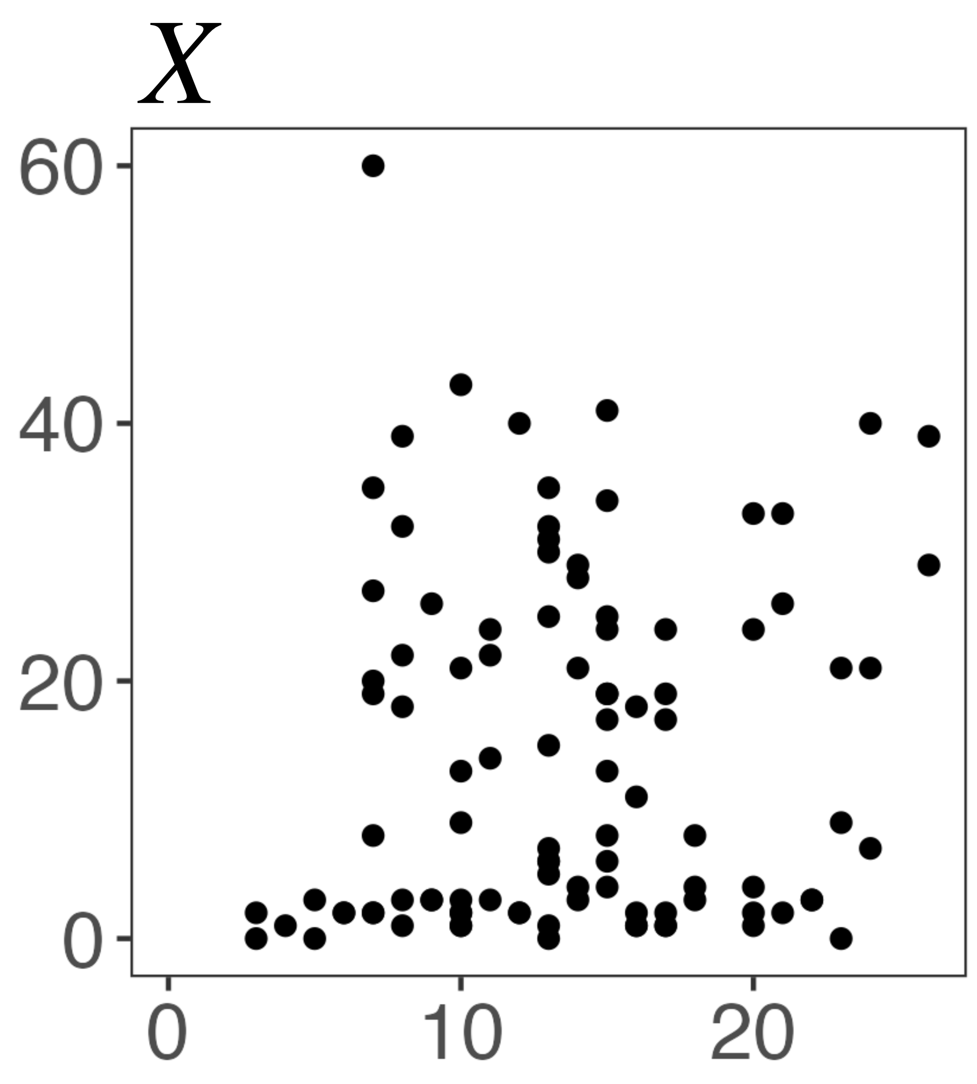
Step 1: thin observations into train/test.

Step 2: cluster the training set.

Step 3: test for difference in means or evaluate clusters on test set.



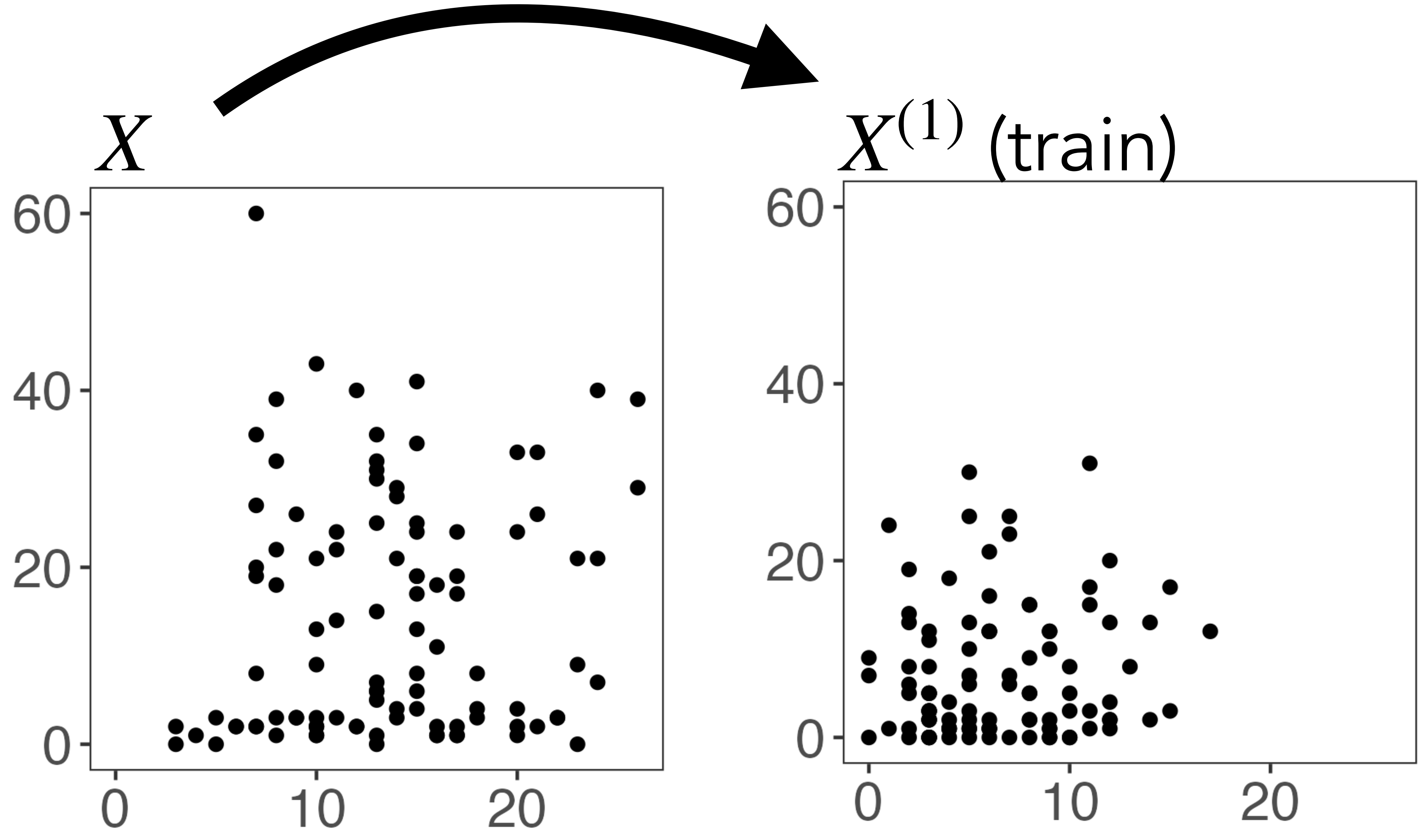
Thinning avoids the pitfall of sample splitting on our motivating examples



$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

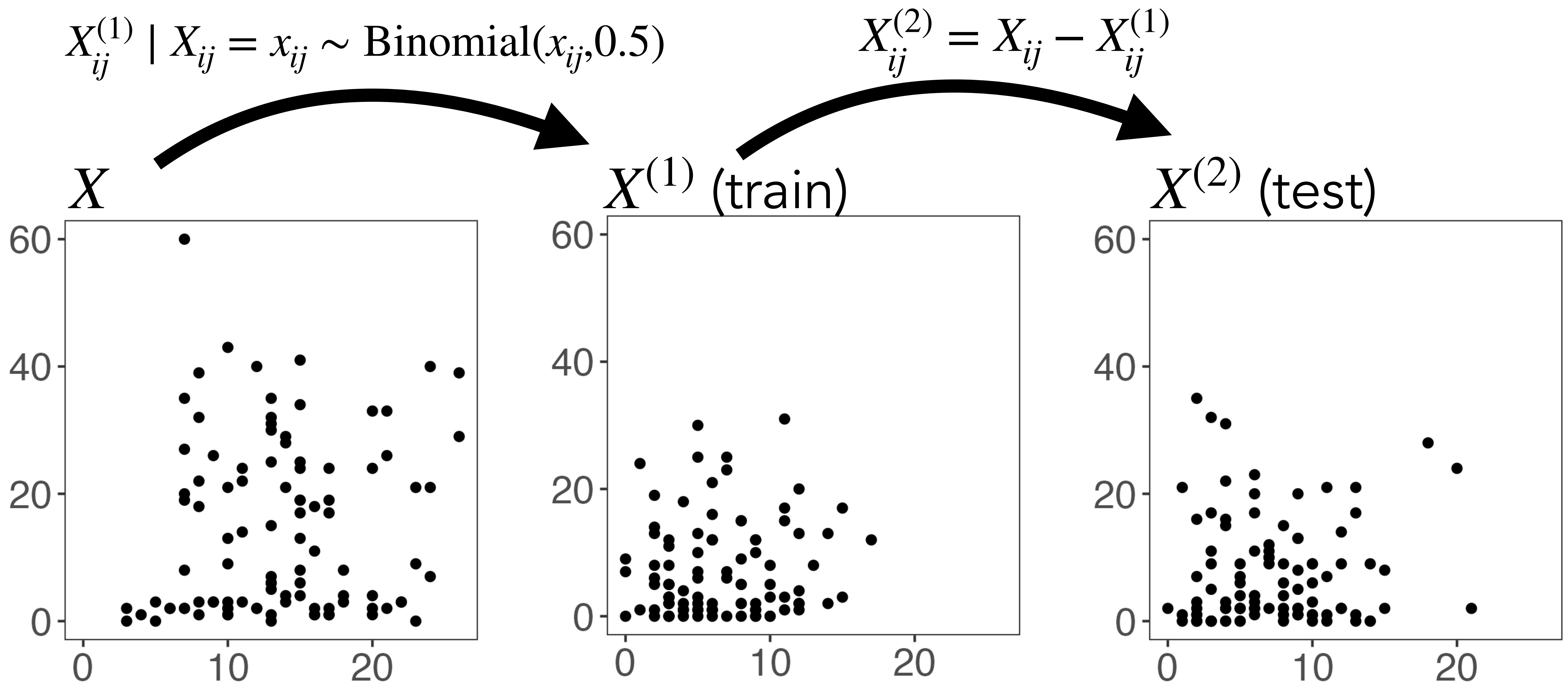
Thinning avoids the pitfall of sample splitting on our motivating examples

$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$



$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

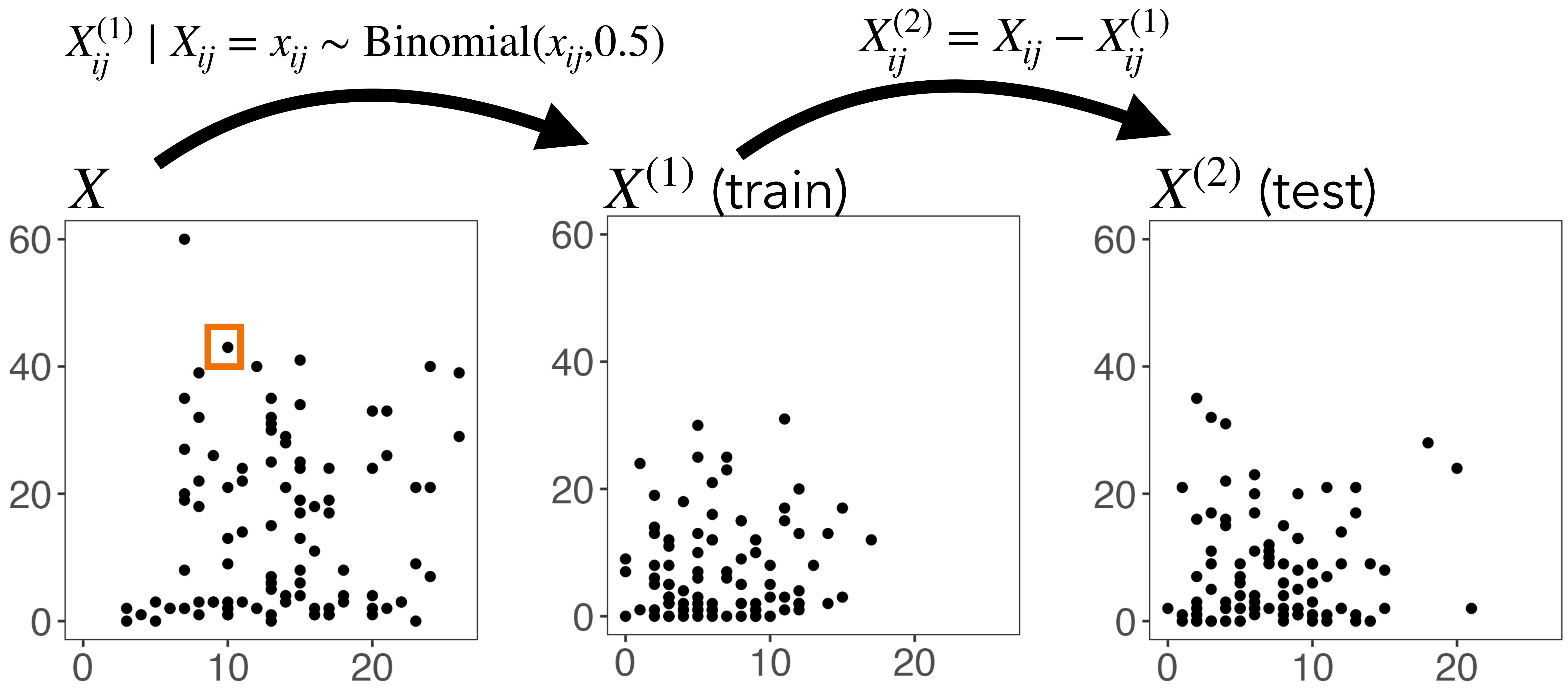
X

$X^{(1)}$ (train)

$X^{(2)}$ (test)

$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

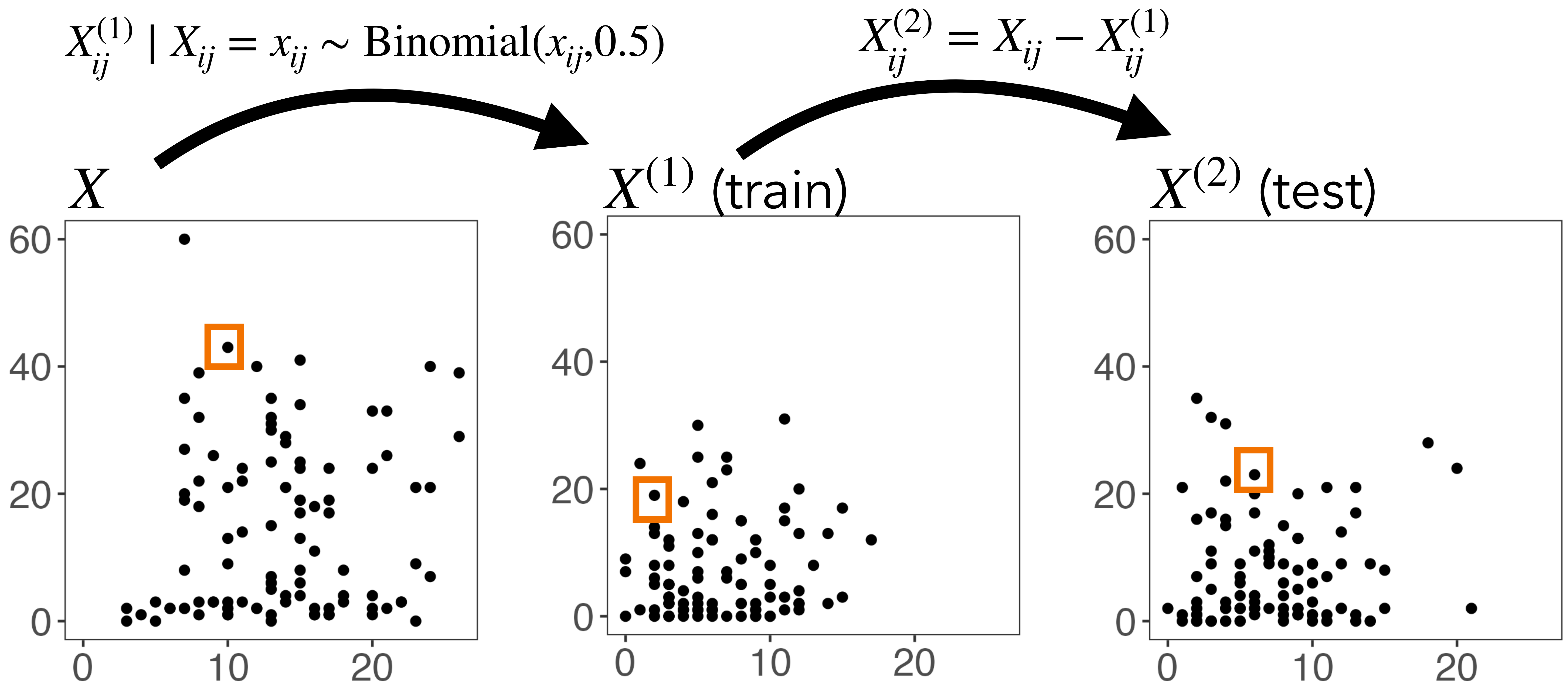
X

$X^{(1)}$ (train)

$X^{(2)}$ (test)

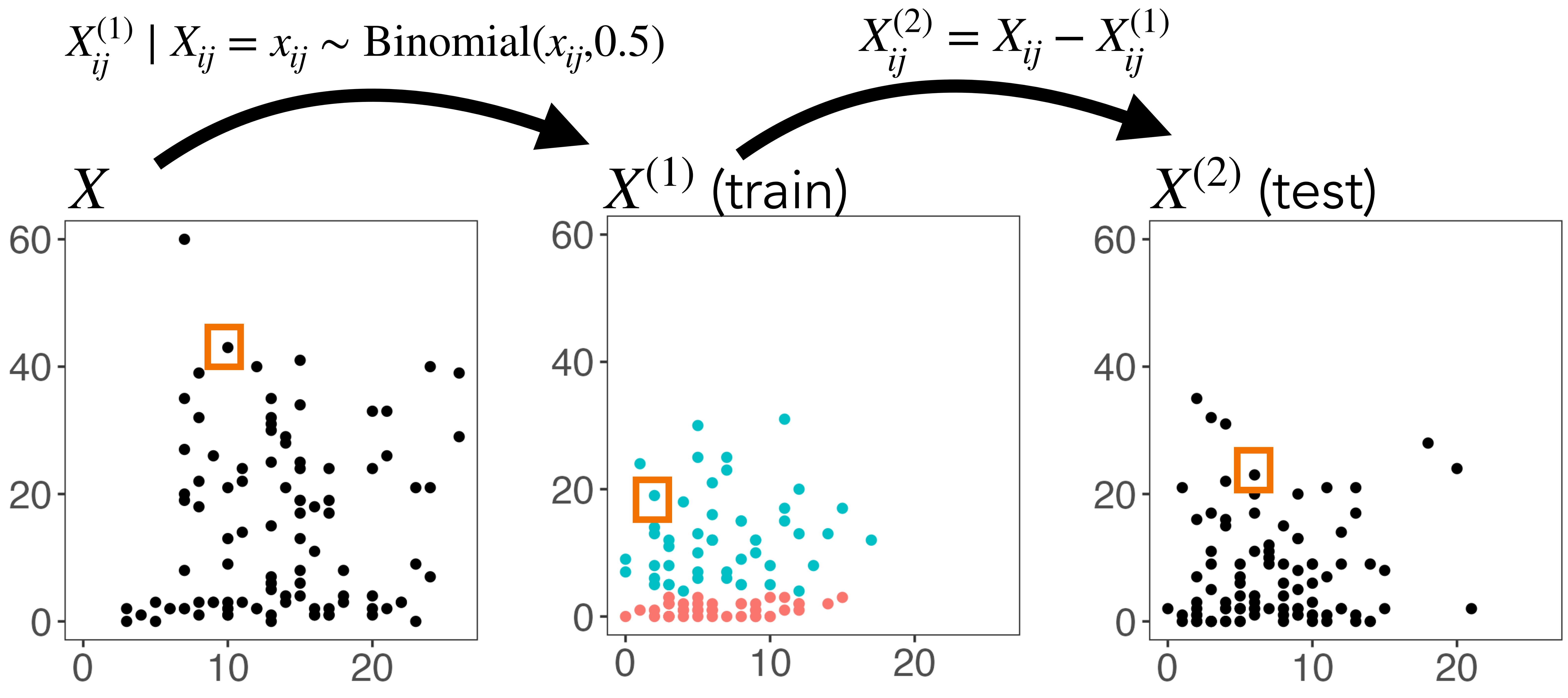
$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

Thinning avoids the pitfall of sample splitting on our motivating examples



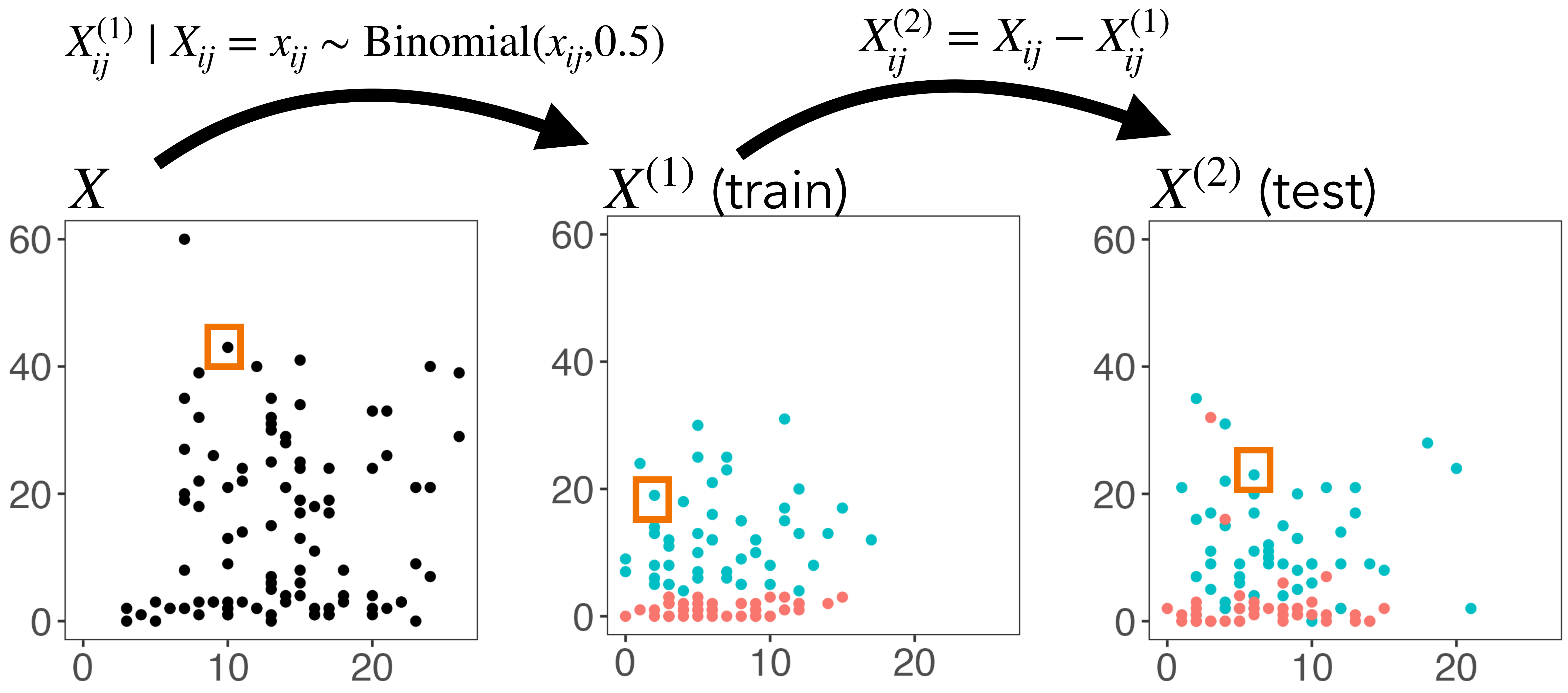
$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

Thinning avoids the pitfall of sample splitting on our motivating examples



$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

X

$X^{(1)}$ (train)

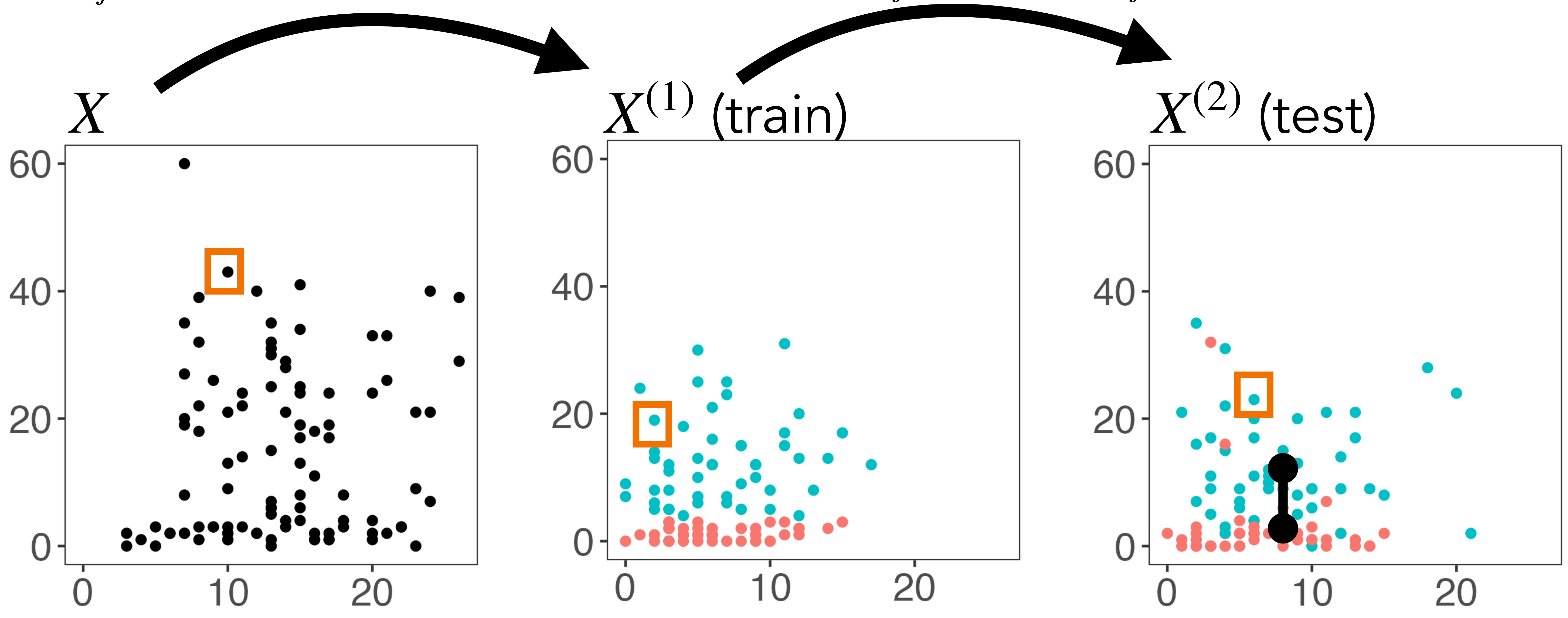
$X^{(2)}$ (test)

$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

Thinning avoids the pitfall of sample splitting on our motivating examples

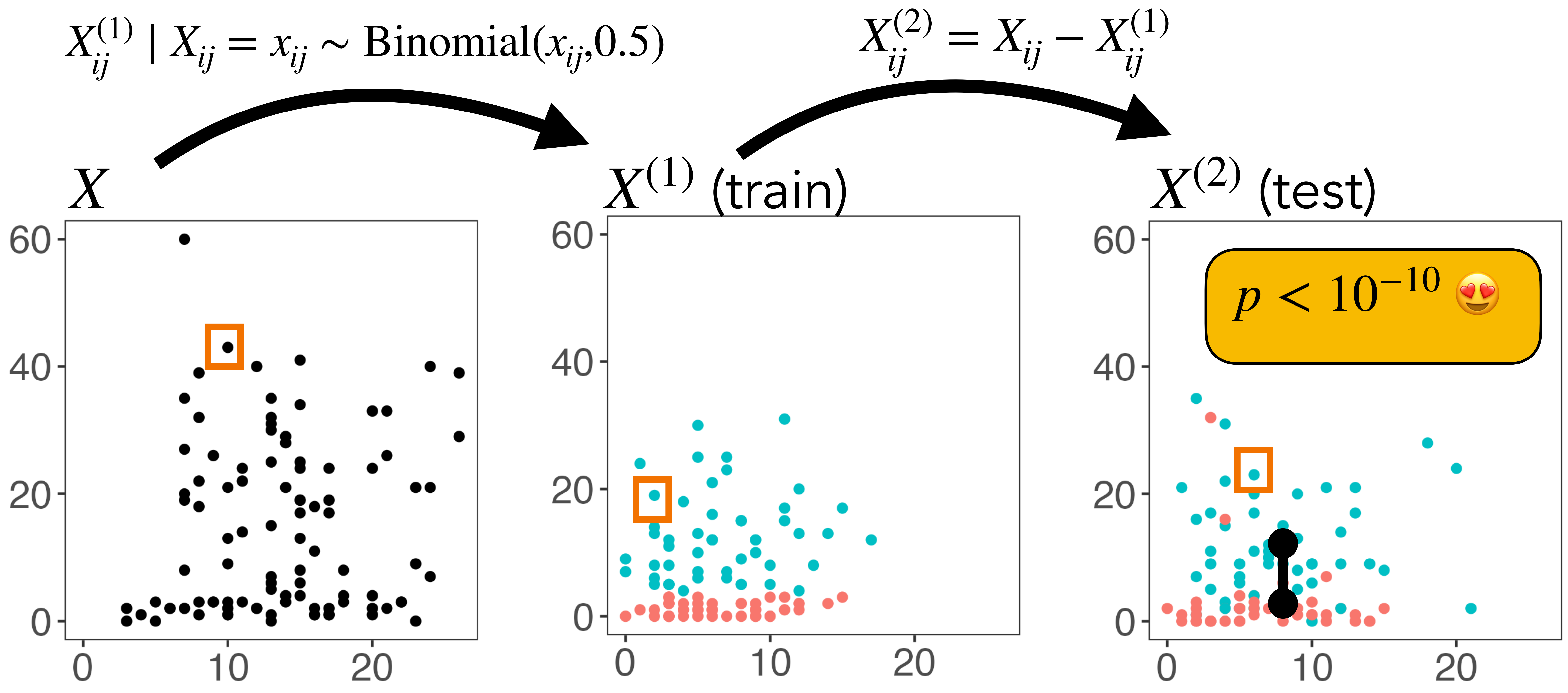
$$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$$

$$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$$



$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

Thinning avoids the pitfall of sample splitting on our motivating examples



$X^{(1)}_{ij} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

$X^{(2)}_{ij} = X_{ij} - X^{(1)}_{ij}$

X

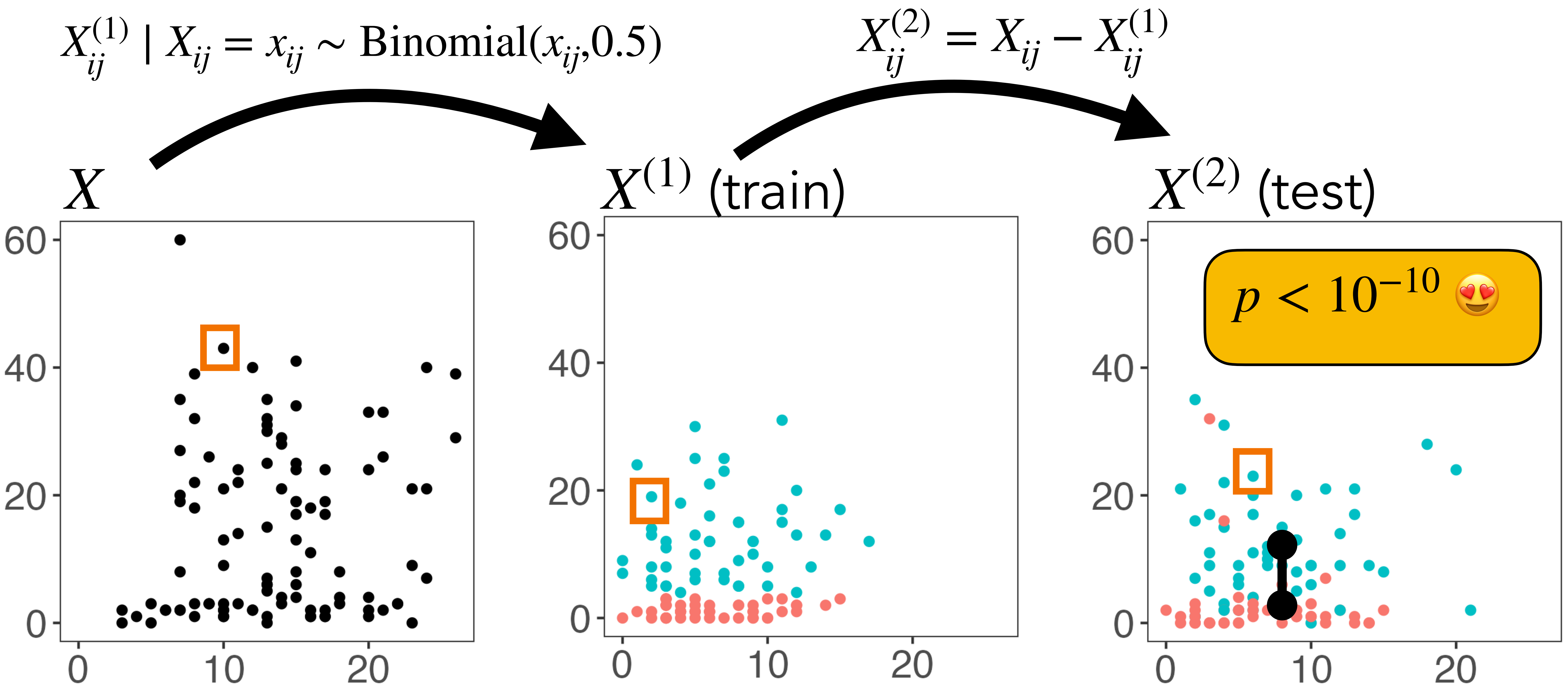
$X^{(1)}$ (train)

$X^{(2)}$ (test)

$p < 10^{-10}$ 😍

$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

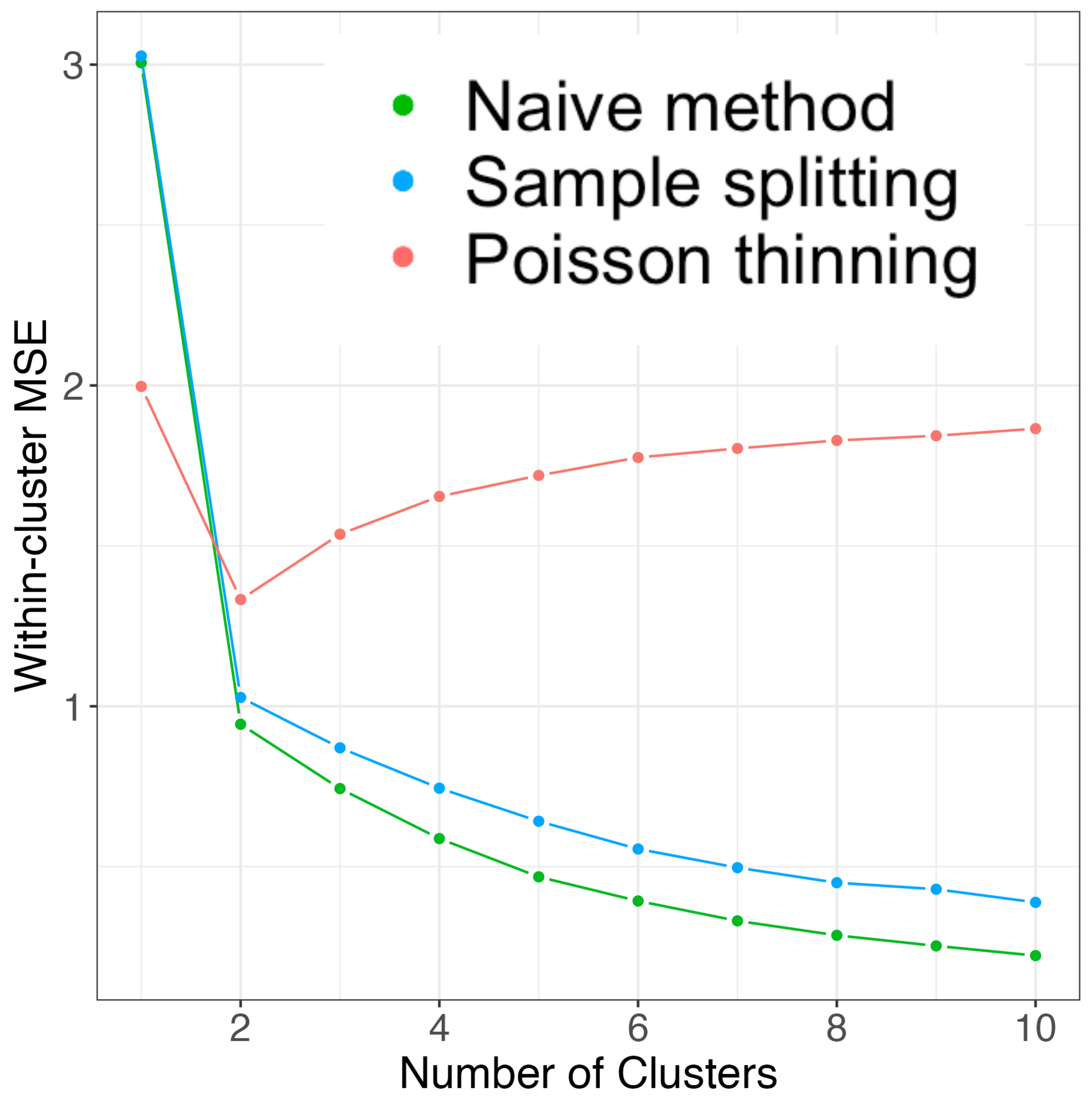
Thinning avoids the pitfall of sample splitting on our motivating examples



$X_{ij}^{(1)} \mid X_{ij} = x_{ij} \sim \text{Binomial}(x_{ij}, 0.5)$

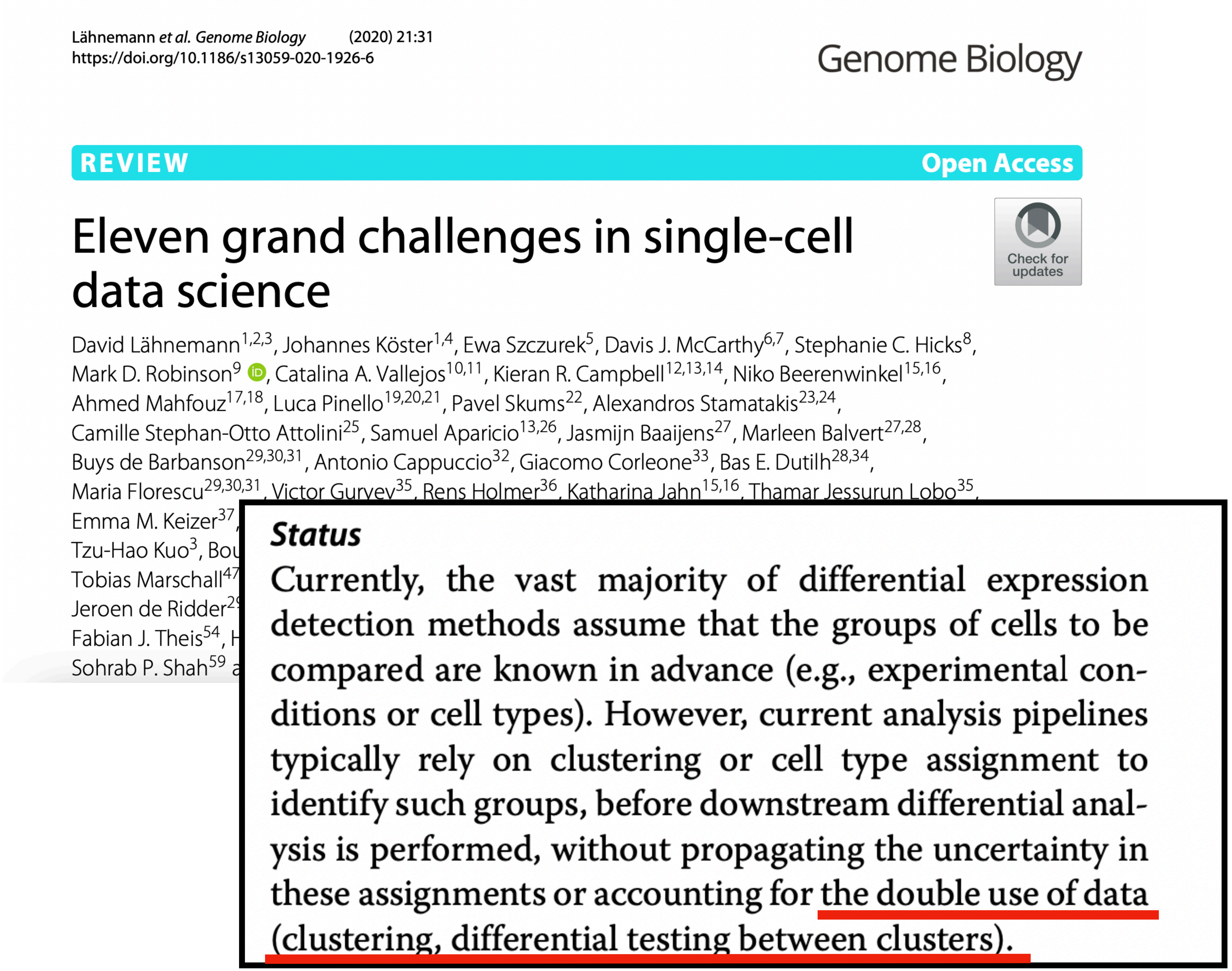
$X_{ij}^{(2)} = X_{ij} - X_{ij}^{(1)}$

$p < 10^{-10}$ 😍



$$X_{i2} \sim \begin{cases} \text{Poisson}(3) & \text{if } i \leq 50 \\ \text{Poisson}(25) & \text{if } i > 50 \end{cases}$$

Poisson thinning is useful in the analysis of single-cell RNA sequencing data



Lähnemann *et al. Genome Biology* (2020) 21:31
https://doi.org/10.1186/s13059-020-1926-6

Genome Biology

REVIEW **Open Access**

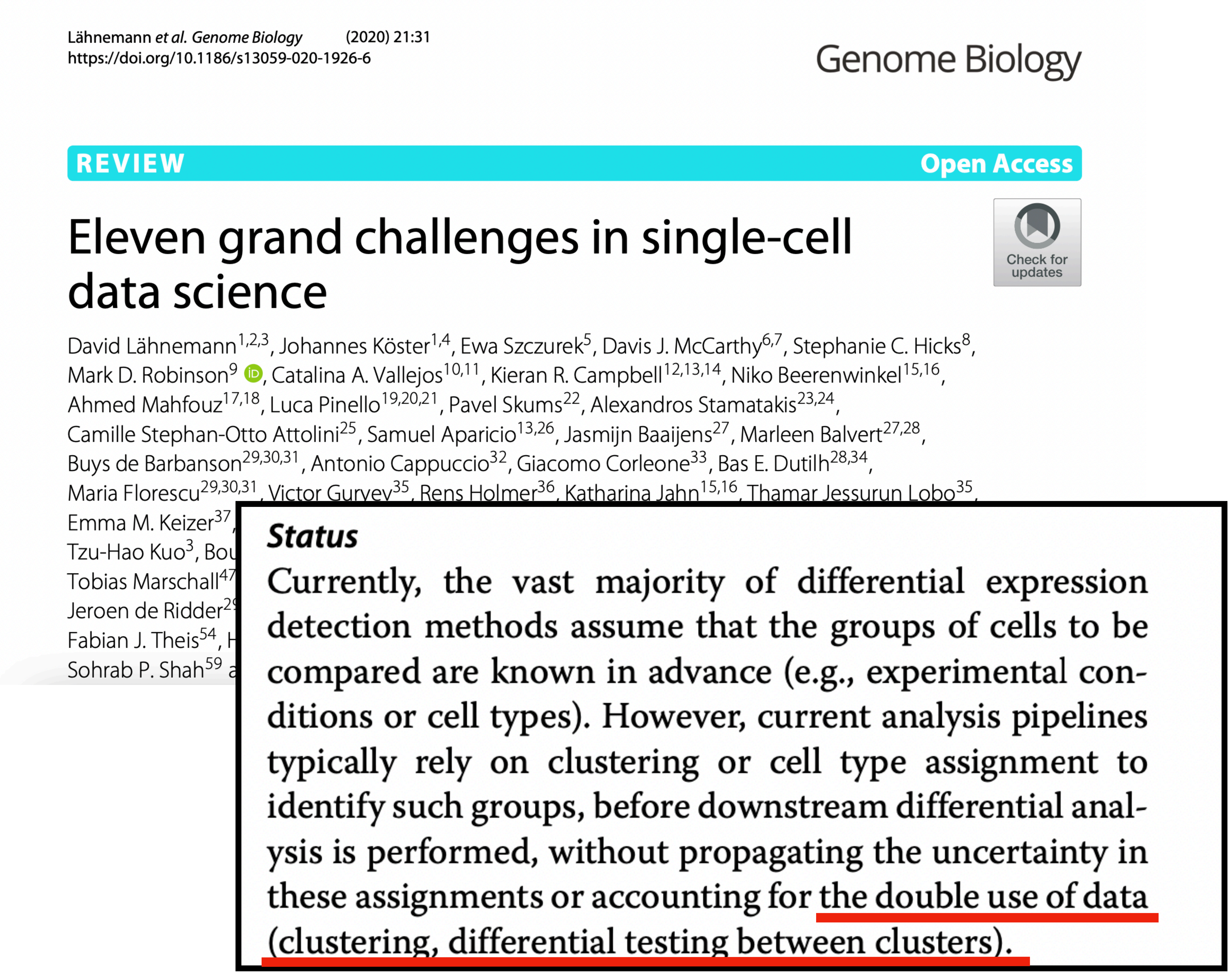
Eleven grand challenges in single-cell data science

David Lähnemann[1,2,3], Johannes Köster[1,4], Ewa Szczurek[5], Davis J. McCarthy[6,7], Stephanie C. Hicks[8], Mark D. Robinson[9], Catalina A. Vallejos[10,11], Kieran R. Campbell[12,13,14], Niko Beerenwinkel[15,16], Ahmed Mahfouz[17,18], Luca Pinello[19,20,21], Pavel Skums[22], Alexandros Stamatakis[23,24], Camille Stephan-Otto Attolini[25], Samuel Aparicio[13,26], Jasmijn Baaijens[27], Marleen Balvert[27,28], Buys de Barbanson[29,30,31], Antonio Cappuccio[32], Giacomo Corleone[33], Bas E. Dutilh[28,34], Maria Florescu[29,30,31], Victor Guryev[35], Rens Holmer[36], Katharina Jahn[15,16], Thamar Jessurun Lobo[35],
Emma M. Keizer[37],
Tzu-Hao Kuo[3], Bou
Tobias Marschall[47]
Jeroen de Ridder[29
Fabian J. Theis[54], H
Sohrab P. Shah[59] a

Status

Currently, the vast majority of differential expression detection methods assume that the groups of cells to be compared are known in advance (e.g., experimental conditions or cell types). However, current analysis pipelines typically rely on clustering or cell type assignment to identify such groups, before downstream differential analysis is performed, without propagating the uncertainty in these assignments or accounting for the double use of data (clustering, differential testing between clusters).

Poisson thinning is useful in the analysis of single-cell RNA sequencing data



Lähnemann *et al. Genome Biology* (2020) 21:31
https://doi.org/10.1186/s13059-020-1926-6

Genome Biology

REVIEW **Open Access**

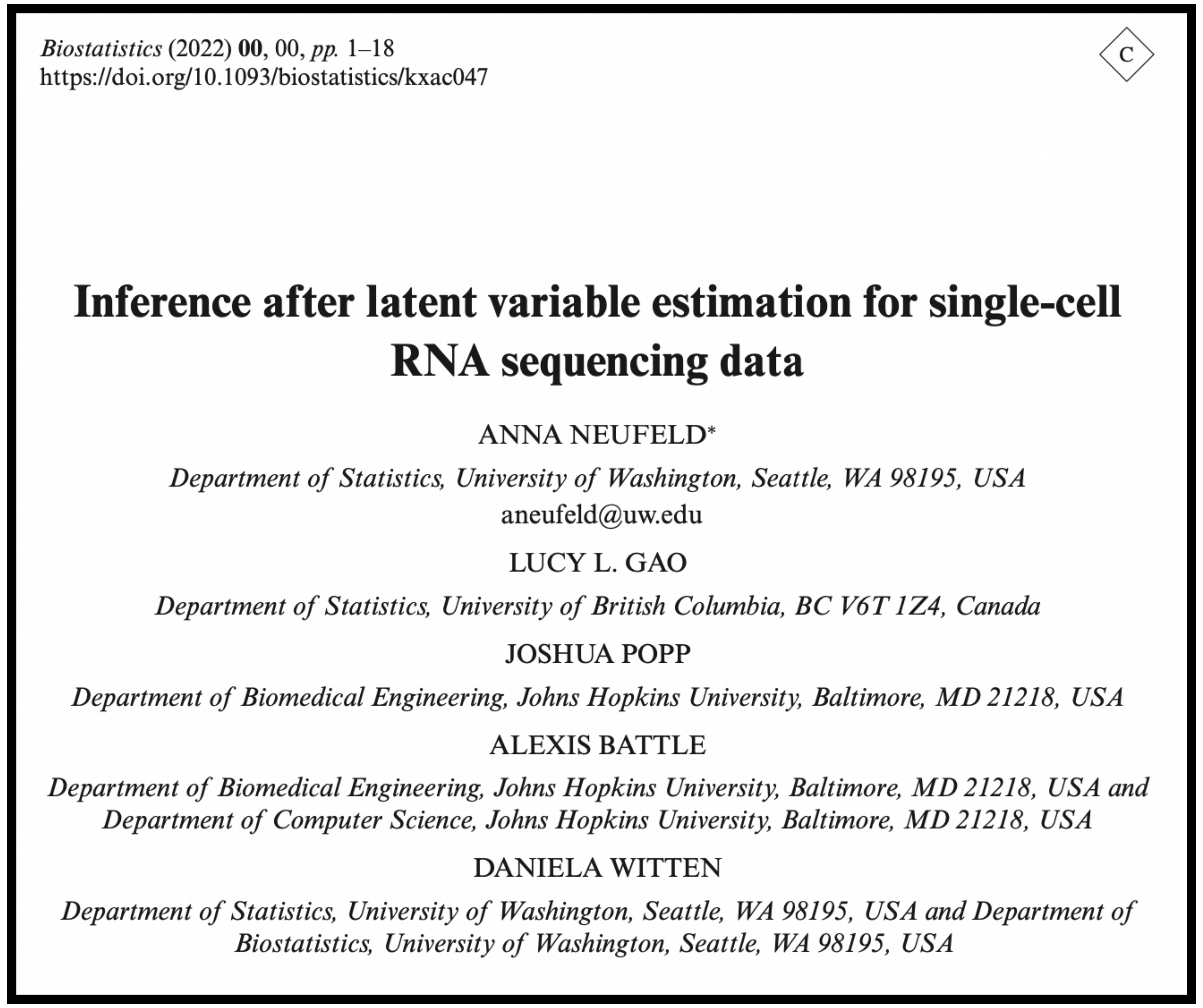
Eleven grand challenges in single-cell data science

David Lähnemann[1,2,3], Johannes Köster[1,4], Ewa Szczurek[5], Davis J. McCarthy[6,7], Stephanie C. Hicks[8], Mark D. Robinson[9], Catalina A. Vallejos[10,11], Kieran R. Campbell[12,13,14], Niko Beerenwinkel[15,16], Ahmed Mahfouz[17,18], Luca Pinello[19,20,21], Pavel Skums[22], Alexandros Stamatakis[23,24], Camille Stephan-Otto Attolini[25], Samuel Aparicio[13,26], Jasmijn Baaijens[27], Marleen Balvert[27,28], Buys de Barbanson[29,30,31], Antonio Cappuccio[32], Giacomo Corleone[33], Bas E. Dutilh[28,34], Maria Florescu[29,30,31], Victor Guryev[35], Rens Holmer[36], Katharina Jahn[15,16], Thamar Jessurun Lobo[35], Emma M. Keizer[37], Tzu-Hao Kuo[3], Bou Tobias Marschall[47] Jeroen de Ridder[2 Fabian J. Theis[54], H Sohrab P. Shah[59] a

Status

Currently, the vast majority of differential expression detection methods assume that the groups of cells to be compared are known in advance (e.g., experimental conditions or cell types). However, current analysis pipelines typically rely on clustering or cell type assignment to identify such groups, before downstream differential analysis is performed, without propagating the uncertainty in these assignments or accounting for the double use of data (clustering, differential testing between clusters).

Project 2



Biostatistics (2022) **00**, 00, *pp.* 1–18
https://doi.org/10.1093/biostatistics/kxac047

Inference after latent variable estimation for single-cell RNA sequencing data

ANNA NEUFELD*
Department of Statistics, University of Washington, Seattle, WA 98195, USA
aneufeld@uw.edu

LUCY L. GAO
Department of Statistics, University of British Columbia, BC V6T 1Z4, Canada

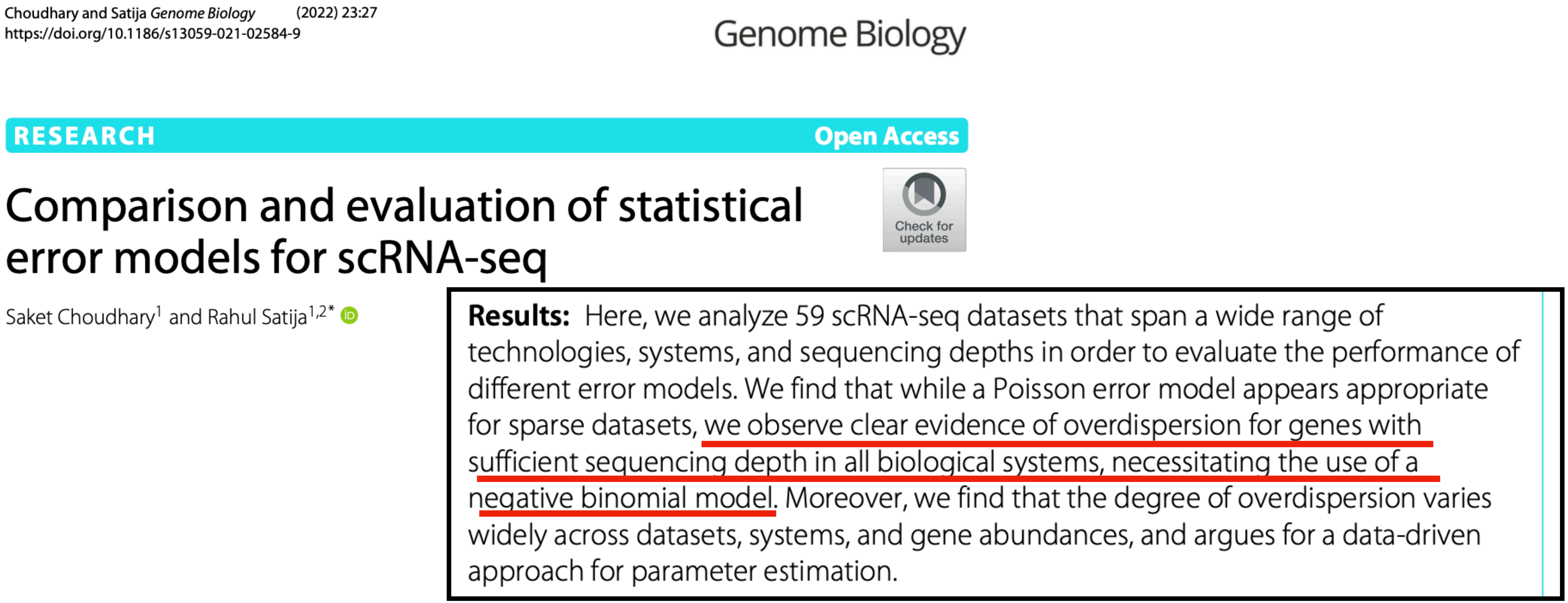
JOSHUA POPP
Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD 21218, USA

ALEXIS BATTLE
Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD 21218, USA and Department of Computer Science, Johns Hopkins University, Baltimore, MD 21218, USA

DANIELA WITTEN
Department of Statistics, University of Washington, Seattle, WA 98195, USA and Department of Biostatistics, University of Washington, Seattle, WA 98195, USA

R package and tutorials:
https://anna-neufeld.github.io/countsplit/

But generalizations of Poisson thinning are needed



Choudhary and Satija *Genome Biology* (2022) 23:27
https://doi.org/10.1186/s13059-021-02584-9

Genome Biology

RESEARCH Open Access

Comparison and evaluation of statistical error models for scRNA-seq

Saket Choudhary[1] and Rahul Satija[1,2*]

Results: Here, we analyze 59 scRNA-seq datasets that span a wide range of technologies, systems, and sequencing depths in order to evaluate the performance of different error models. We find that while a Poisson error model appears appropriate for sparse datasets, we observe clear evidence of overdispersion for genes with sufficient sequencing depth in all biological systems, necessitating the use of a negative binomial model. Moreover, we find that the degree of overdispersion varies widely across datasets, systems, and gene abundances, and argues for a data-driven approach for parameter estimation.

Outline

1. Motivation: settings where sample splitting doesn’t work

2. Poisson thinning

3. **Data thinning**

4. Application to single-cell RNA sequencing data

5. Ongoing work

What did we like about Poisson thinning?

We split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

(1) $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.

(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

What did we like about Poisson thinning?

We split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

(1) $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.

(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

Can we achieve these same properties when X is not Poisson?

Data thinning

Goal: split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

(1) $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.

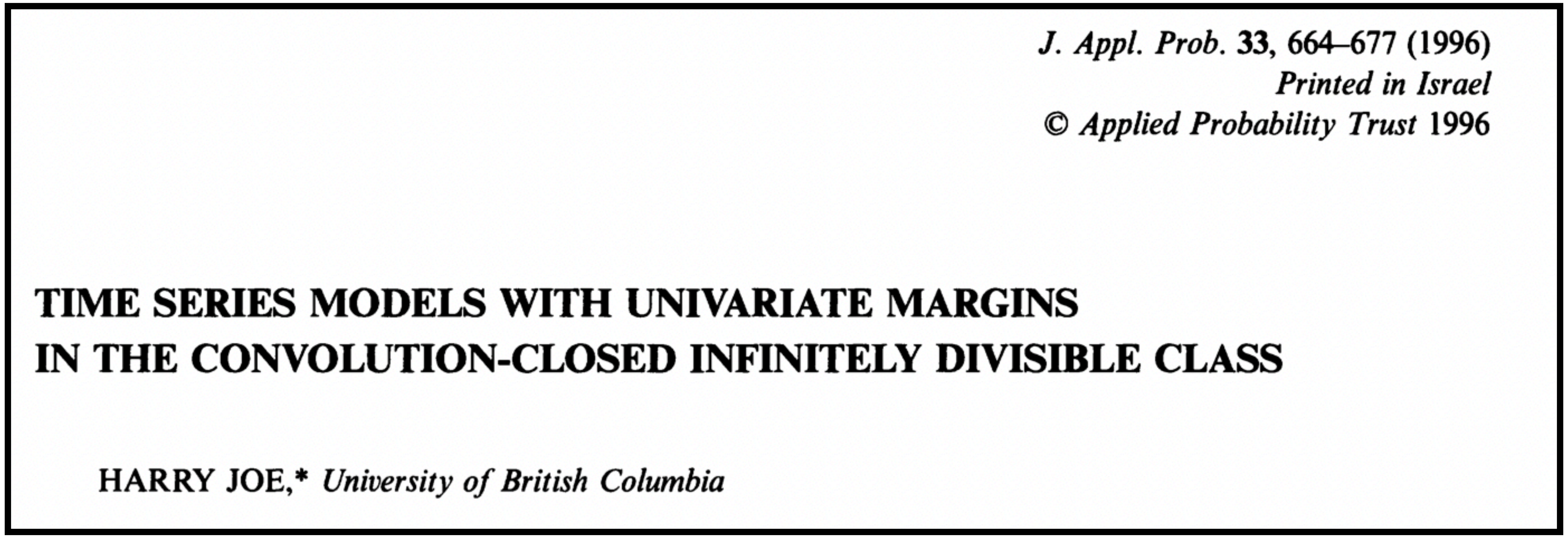
(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

Data thinning

Goal: split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

(1) $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.

(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.



J. Appl. Prob. **33**, 664–677 (1996)
Printed in Israel
© *Applied Probability Trust* 1996

TIME SERIES MODELS WITH UNIVARIATE MARGINS IN THE CONVOLUTION-CLOSED INFINITELY DIVISIBLE CLASS

HARRY JOE,* *University of British Columbia*

Convolution-closed distributions

A family of distributions F_λ is "convolution-closed" in parameter λ if

- $X' \sim F_{\lambda_1}$
- $X'' \sim F_{\lambda_2}$
- $X' \perp\!\!\!\perp X''$

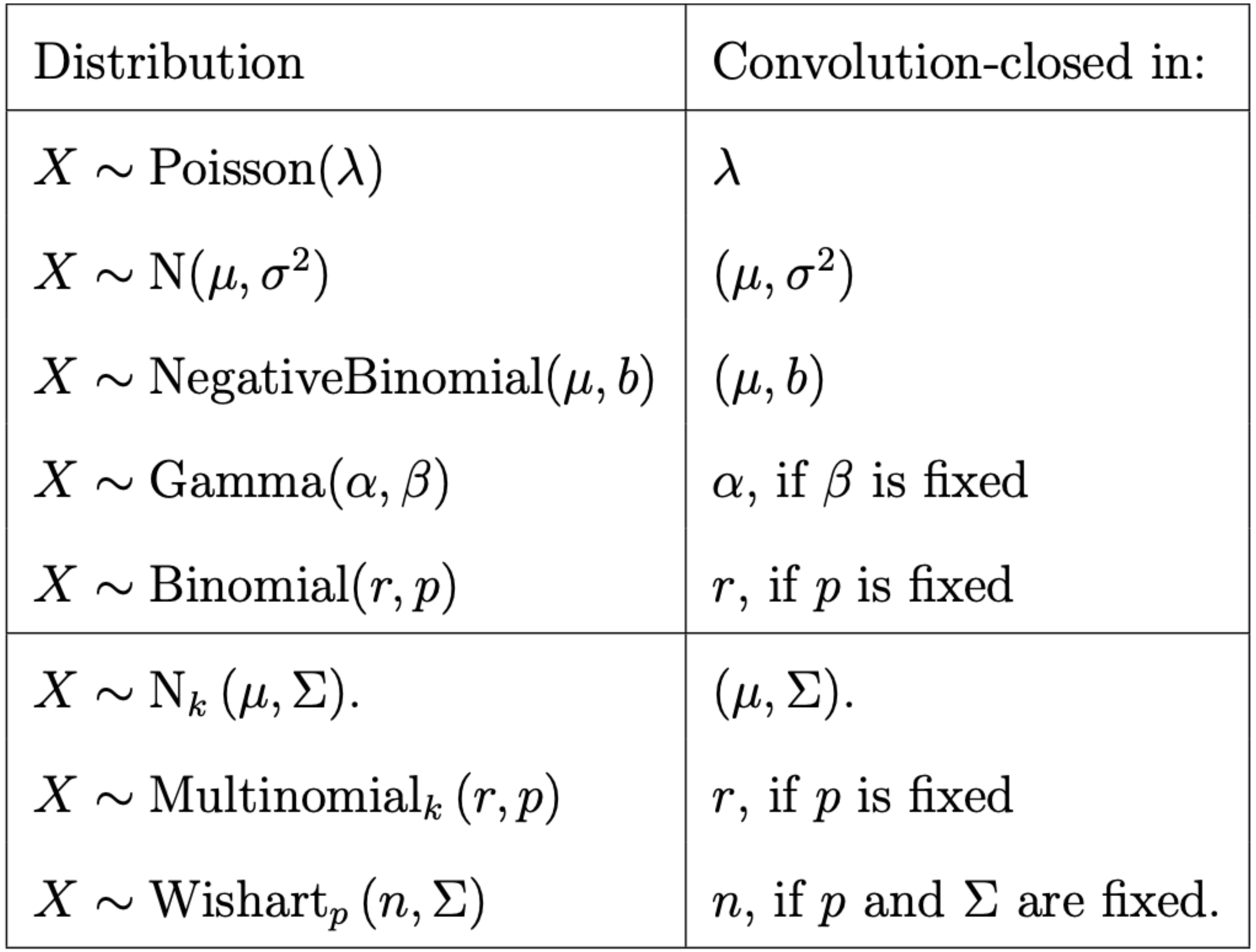
together imply that $X' + X'' \sim F_{\lambda_1+\lambda_2}$.

Convolution-closed distributions

A family of distributions F_λ is "convolution-closed" in parameter λ if

- $X' \sim F_{\lambda_1}$
- $X'' \sim F_{\lambda_2}$
- $X' \perp\!\!\!\perp X''$

together imply that $X' + X'' \sim F_{\lambda_1 + \lambda_2}$.



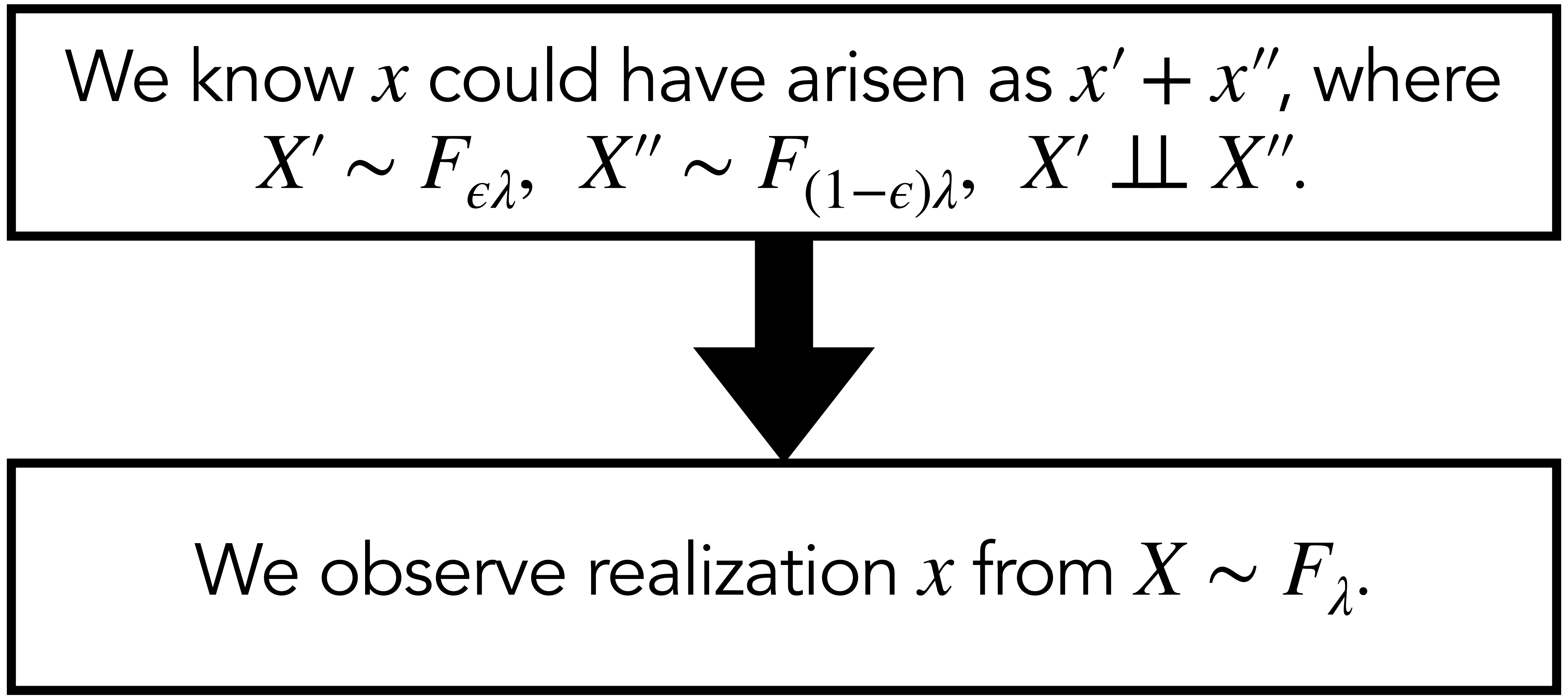
Distribution	Convolution-closed in:
$X \sim \text{Poisson}(\lambda)$	λ
$X \sim \text{N}(\mu, \sigma^2)$	(μ, σ^2)
$X \sim \text{NegativeBinomial}(\mu, b)$	(μ, b)
$X \sim \text{Gamma}(\alpha, \beta)$	α, if β is fixed
$X \sim \text{Binomial}(r, p)$	r, if p is fixed
$X \sim \text{N}_k\,(\mu, \Sigma)$.	(μ, Σ).
$X \sim \text{Multinomial}_k\,(r, p)$	r, if p is fixed
$X \sim \text{Wishart}_p\,(n, \Sigma)$	n, if p and Σ are fixed.

Data thinning for convolution-closed distributions

Data thinning for convolution-closed distributions

We observe realization x from $X \sim F_\lambda$.

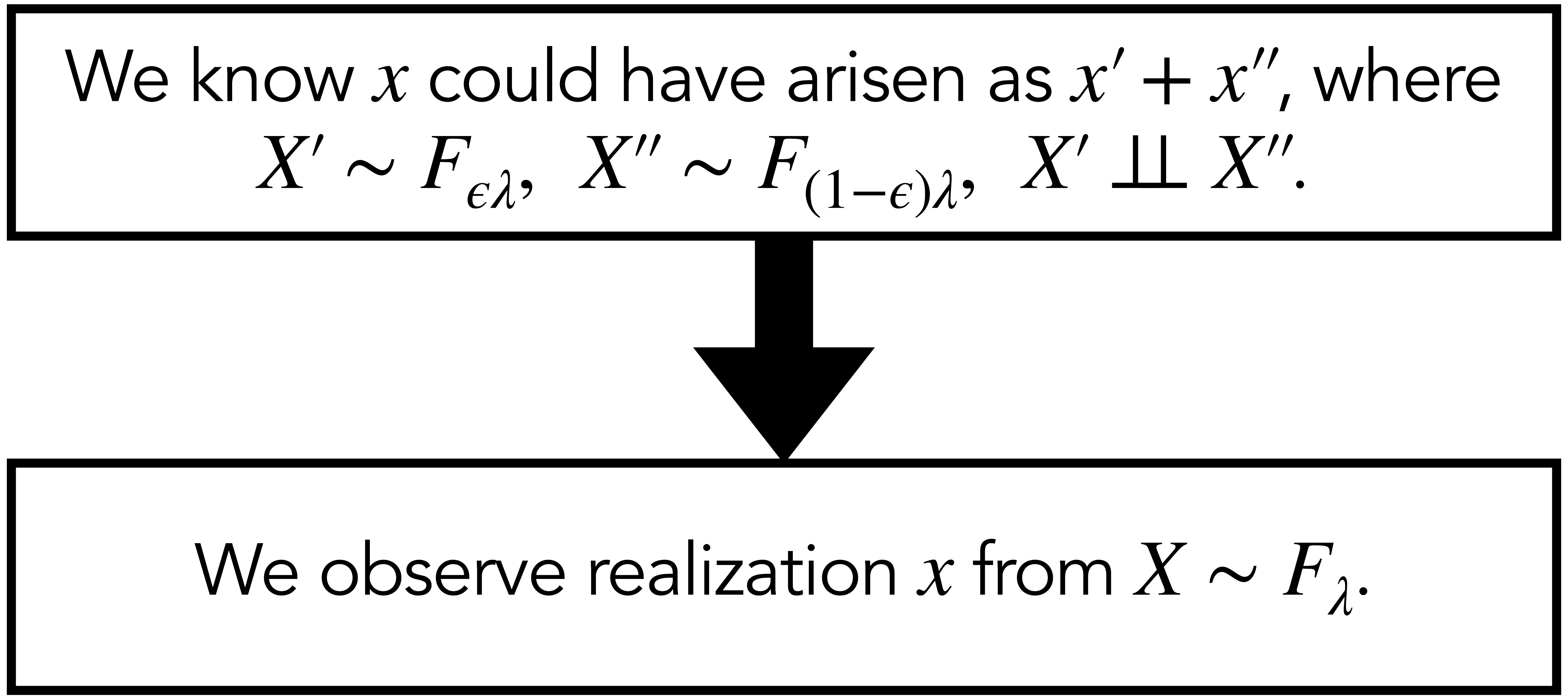
Data thinning for convolution-closed distributions



We know x could have arisen as $x' + x''$, where $X' \sim F_{\epsilon\lambda},\ \ X'' \sim F_{(1-\epsilon)\lambda},\ \ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim F_{\lambda}$.

Data thinning for convolution-closed distributions

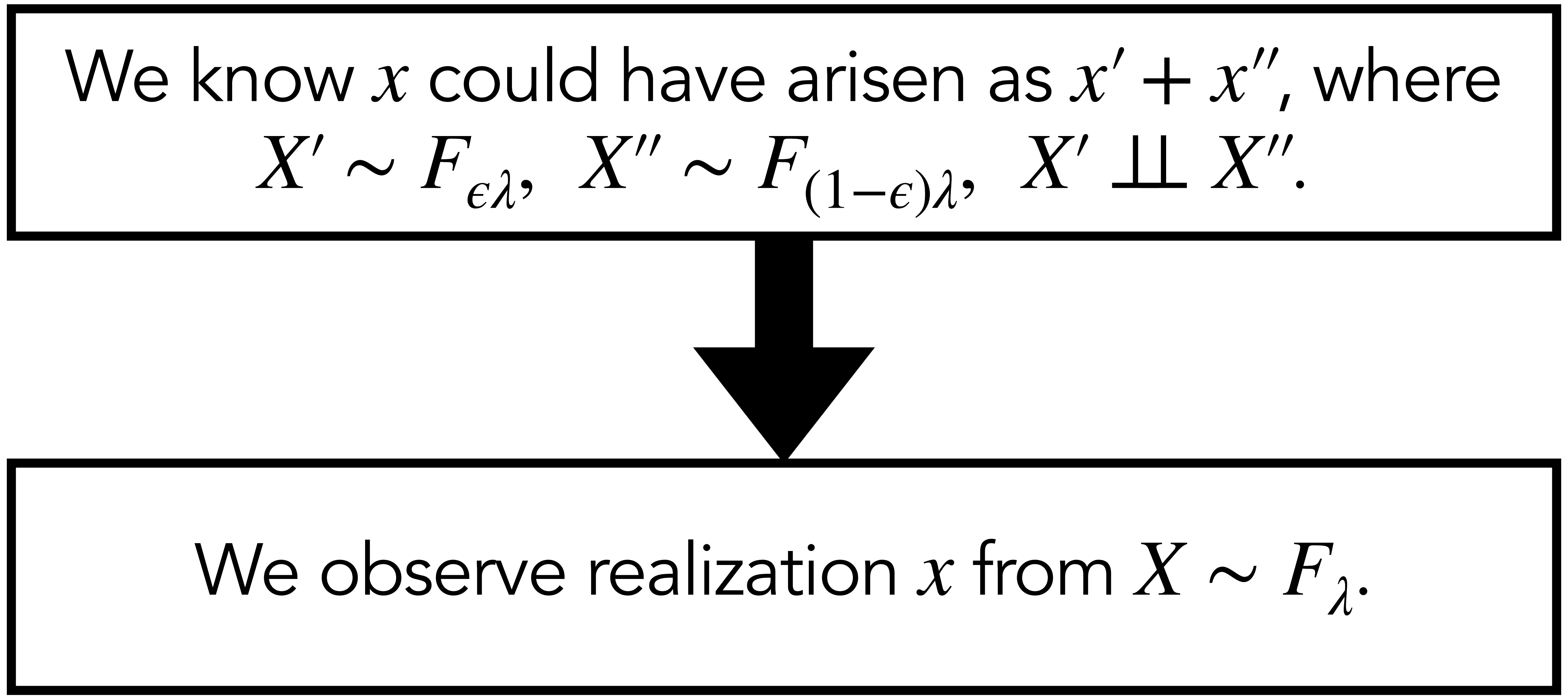


We know x could have arisen as $x' + x''$, where $X' \sim F_{\epsilon\lambda}, \; X'' \sim F_{(1-\epsilon)\lambda}, \; X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim F_{\lambda}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Data thinning for convolution-closed distributions



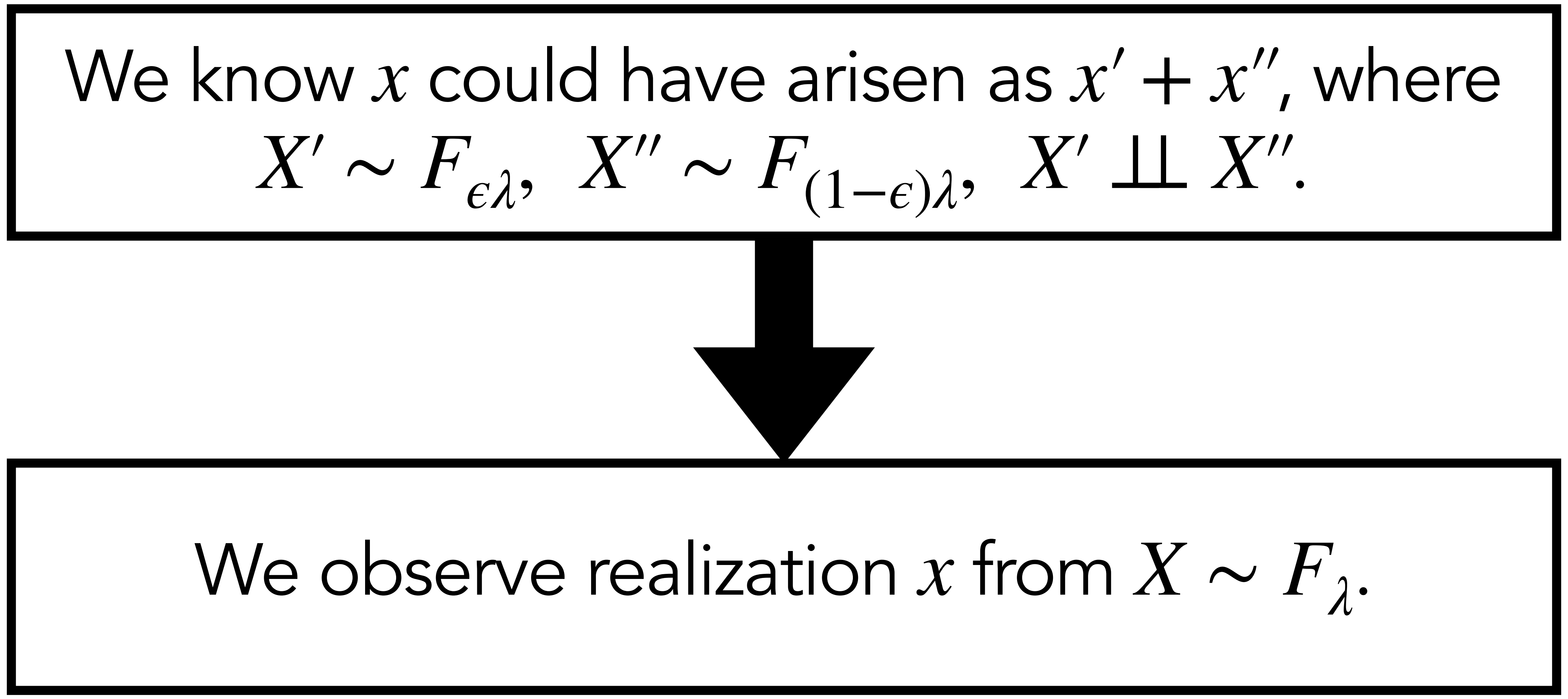
We know x could have arisen as $x' + x''$, where $X' \sim F_{\epsilon\lambda},\ X'' \sim F_{(1-\epsilon)\lambda},\ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim F_{\lambda}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

Data thinning for convolution-closed distributions



We know x could have arisen as $x' + x''$, where $X' \sim F_{\epsilon\lambda},\ \ X'' \sim F_{(1-\epsilon)\lambda},\ \ X' \perp\!\!\!\perp X''$.

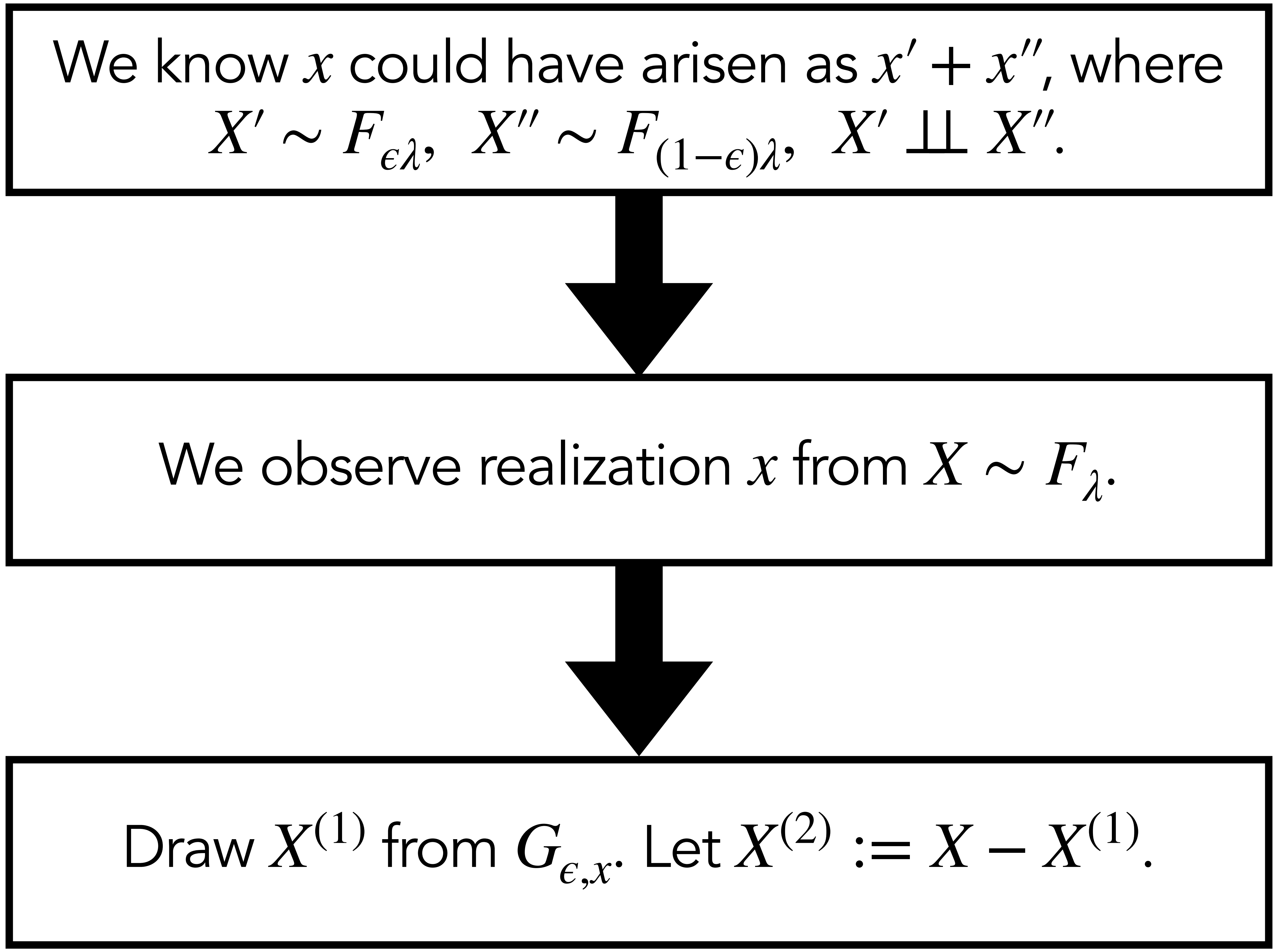
We observe realization x from $X \sim F_{\lambda}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

Let $G_{\epsilon,x}$ be the conditional distribution of $X' \mid X = x$.

Data thinning for convolution-closed distributions



We know x could have arisen as $x' + x''$, where $X' \sim F_{\epsilon\lambda}, \; X'' \sim F_{(1-\epsilon)\lambda}, \; X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim F_\lambda$.

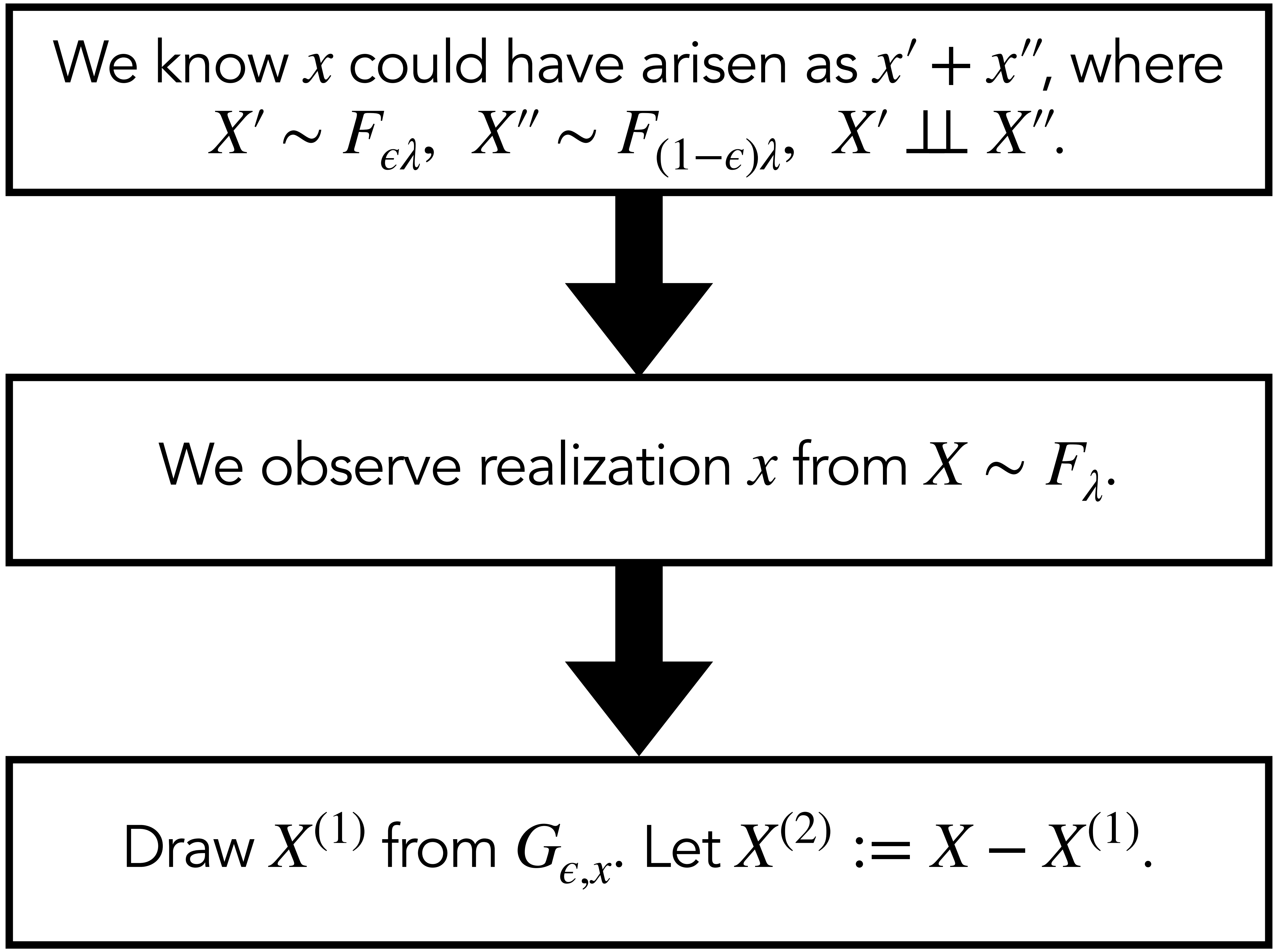
Draw $X^{(1)}$ from $G_{\epsilon,x}$. Let $X^{(2)} := X - X^{(1)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

Let $G_{\epsilon,x}$ be the conditional distribution of $X' \mid X = x$.

Data thinning for convolution-closed distributions



We know x could have arisen as $x' + x''$, where $X' \sim F_{\epsilon\lambda},\ \ X'' \sim F_{(1-\epsilon)\lambda},\ \ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim F_\lambda$.

Draw $X^{(1)}$ from $G_{\epsilon,x}$. Let $X^{(2)} := X - X^{(1)}$.

Theorem:

$X^{(1)} \sim F_{\epsilon\lambda},\ X^{(2)} \sim F_{(1-\epsilon)\lambda},\ \ X^{(1)} \perp\!\!\!\perp X^{(2)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

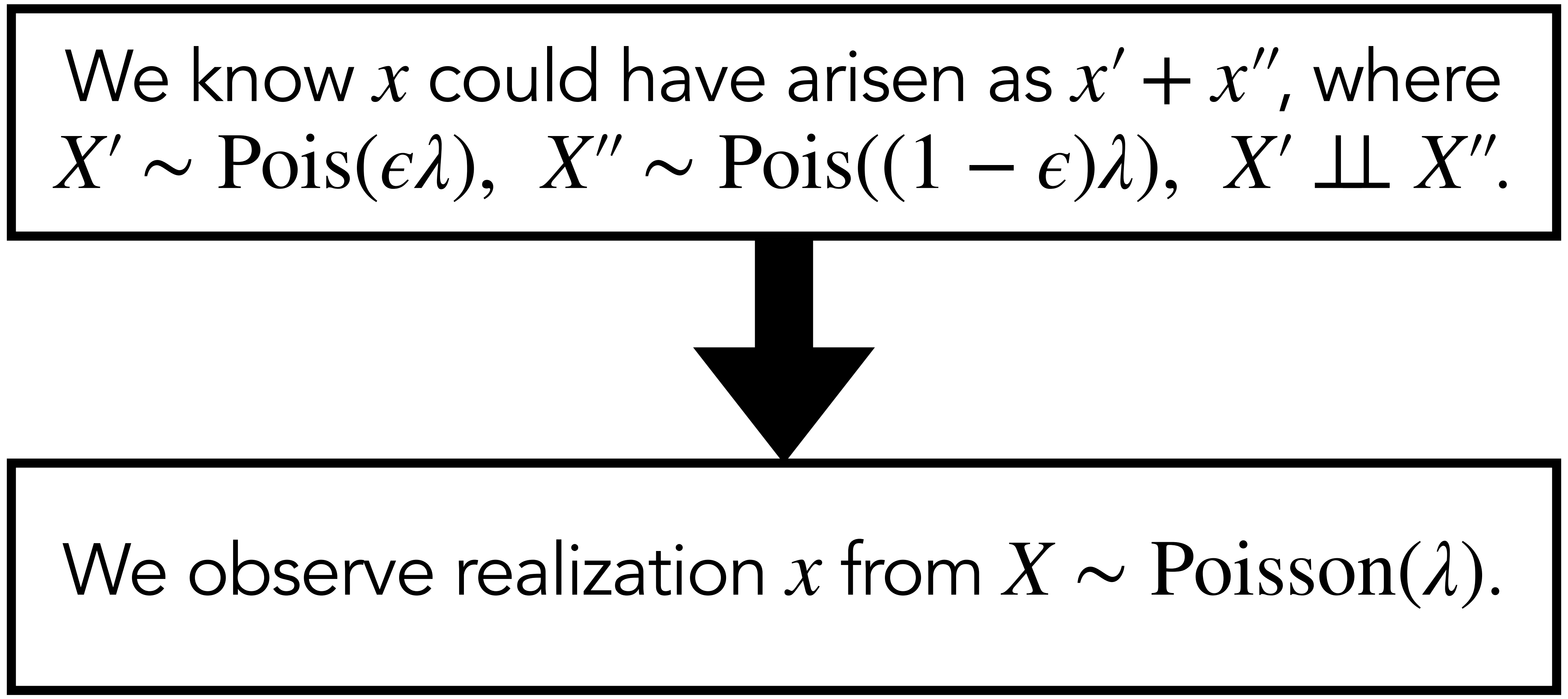
Let $G_{\epsilon,x}$ be the conditional distribution of $X' \mid X = x$.

Data thinning for the Poisson distribution

Data thinning for the Poisson distribution

We observe realization x from $X \sim \text{Poisson}(\lambda)$.

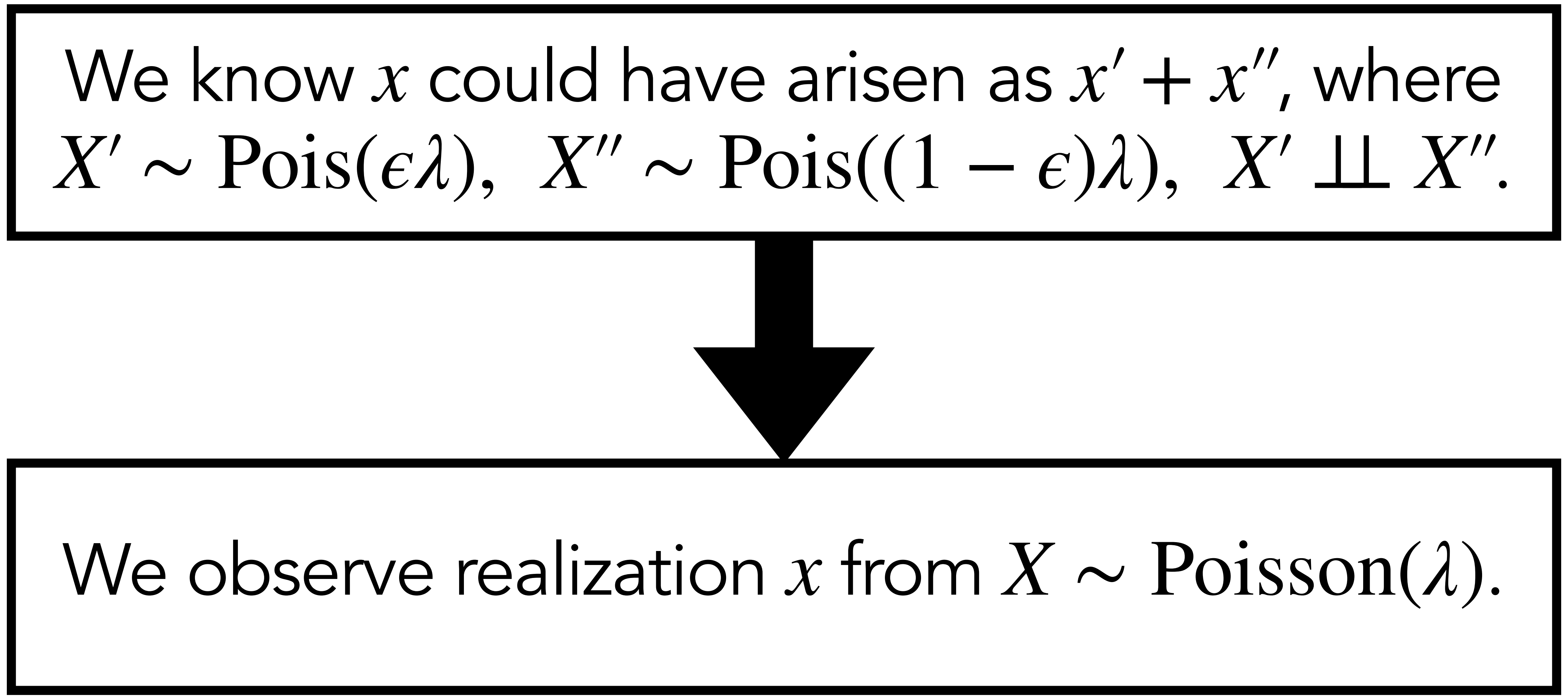
Data thinning for the Poisson distribution



We know x could have arisen as $x' + x''$, where $X' \sim \text{Pois}(\epsilon\lambda),\ \ X'' \sim \text{Pois}((1-\epsilon)\lambda),\ \ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \text{Poisson}(\lambda)$.

Data thinning for the Poisson distribution

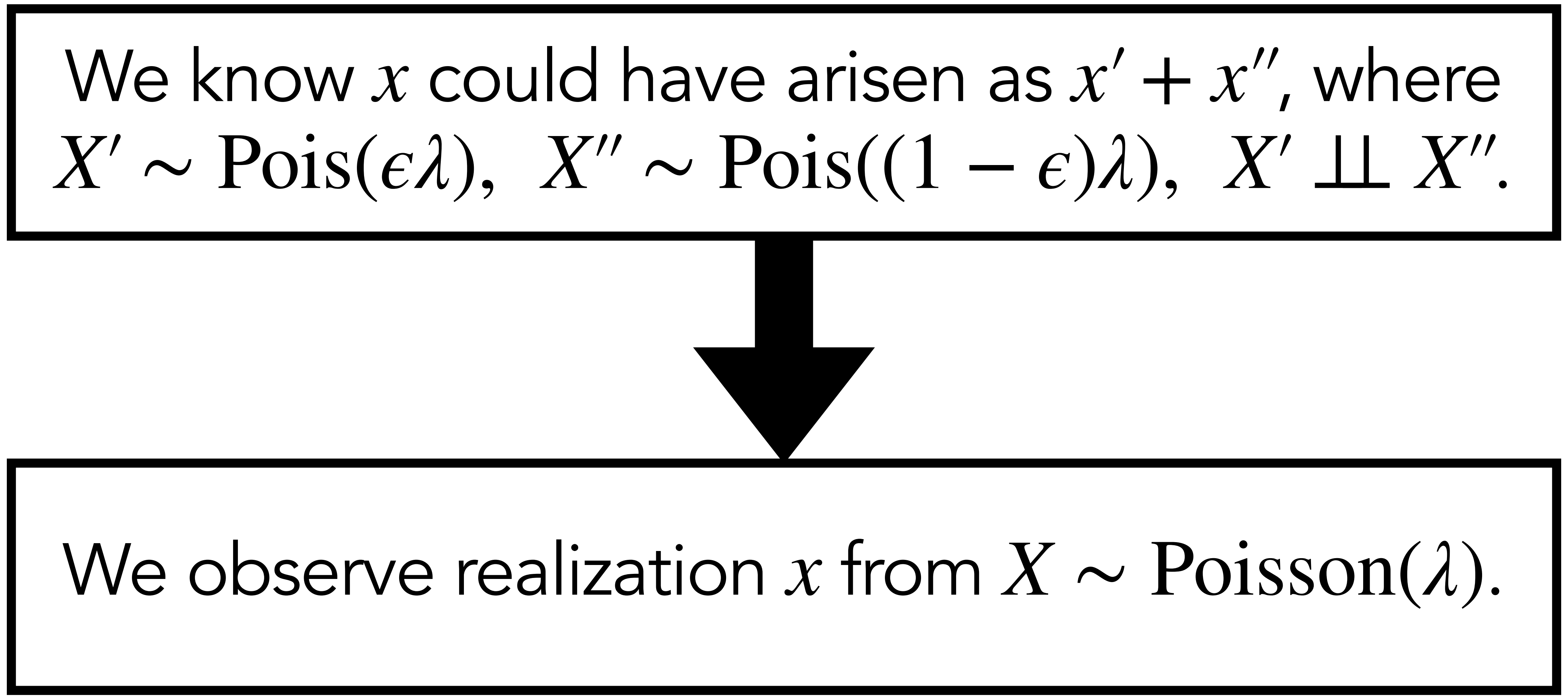


We know x could have arisen as $x' + x''$, where $X' \sim \text{Pois}(\epsilon\lambda),\ \ X'' \sim \text{Pois}((1-\epsilon)\lambda),\ \ X' \perp\!\!\!\perp X''.$

We observe realization x from $X \sim \text{Poisson}(\lambda).$

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Data thinning for the Poisson distribution



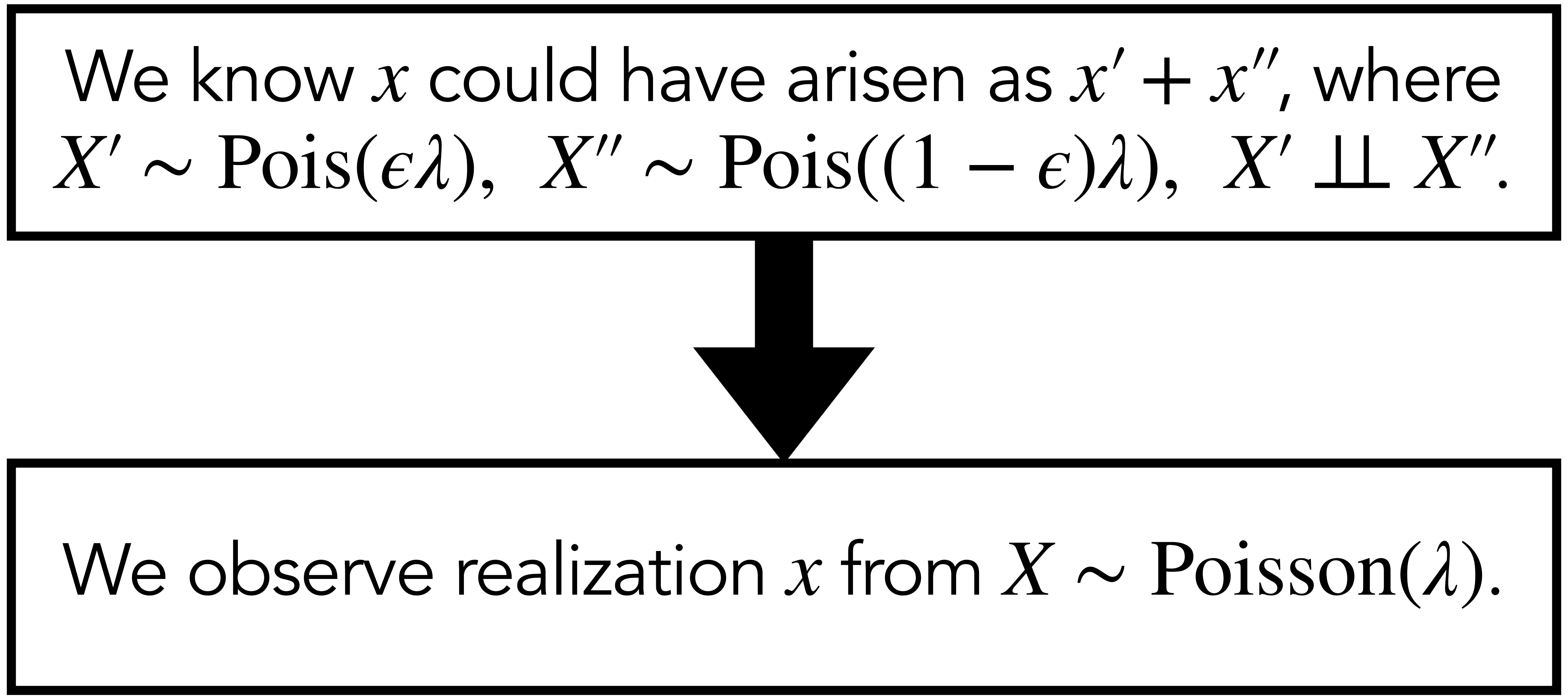
We know x could have arisen as $x' + x''$, where $X' \sim \text{Pois}(\epsilon\lambda),\ \ X'' \sim \text{Pois}((1-\epsilon)\lambda),\ \ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \text{Poisson}(\lambda)$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

Data thinning for the Poisson distribution



We know x could have arisen as $x' + x''$, where $X' \sim \text{Pois}(\epsilon\lambda),\ \ X'' \sim \text{Pois}((1-\epsilon)\lambda),\ \ X' \perp\!\!\!\perp X''$.

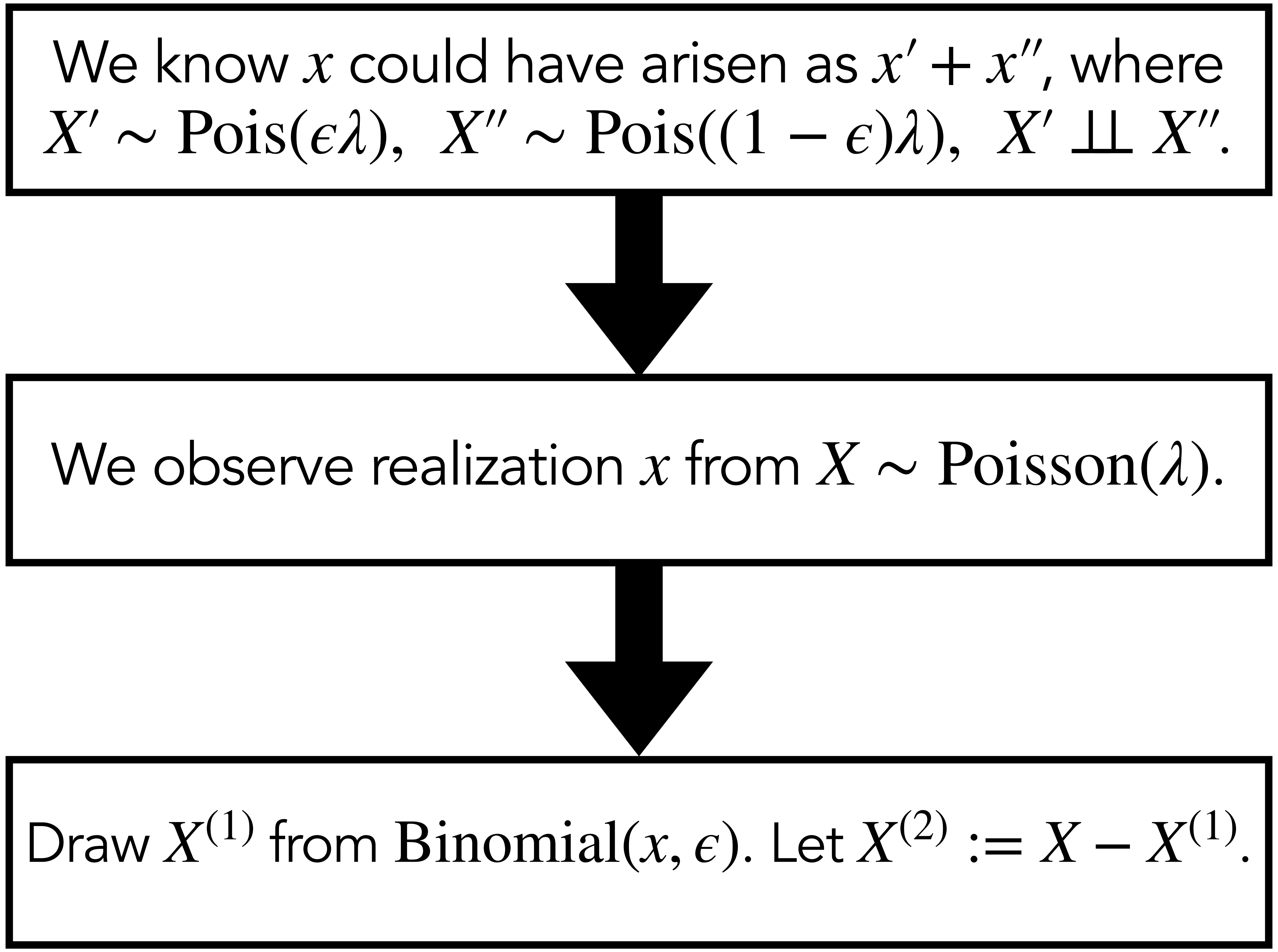
We observe realization x from $X \sim \text{Poisson}(\lambda)$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\text{Binomial}(x, \epsilon)$.

Data thinning for the Poisson distribution



We know x could have arisen as $x' + x''$, where $X' \sim \text{Pois}(\epsilon\lambda),\;\; X'' \sim \text{Pois}((1-\epsilon)\lambda),\;\; X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \text{Poisson}(\lambda)$.

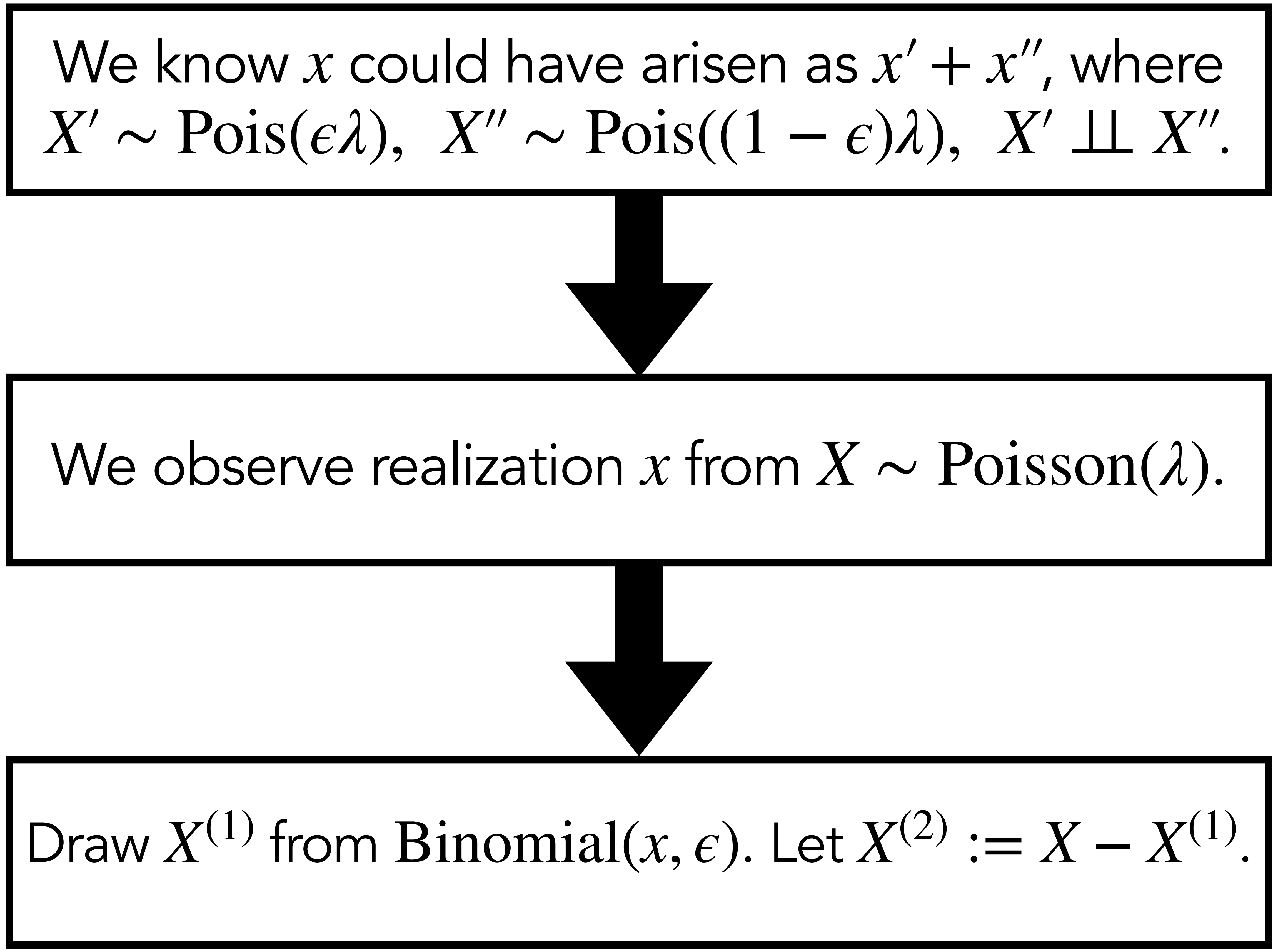
Draw $X^{(1)}$ from $\text{Binomial}(x, \epsilon)$. Let $X^{(2)} := X - X^{(1)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\text{Binomial}(x, \epsilon)$.

Data thinning for the Poisson distribution



We know x could have arisen as $x' + x''$, where $X' \sim \text{Pois}(\epsilon\lambda),\ \ X'' \sim \text{Pois}((1-\epsilon)\lambda),\ \ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \text{Poisson}(\lambda)$.

Draw $X^{(1)}$ from $\text{Binomial}(x, \epsilon)$. Let $X^{(2)} := X - X^{(1)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

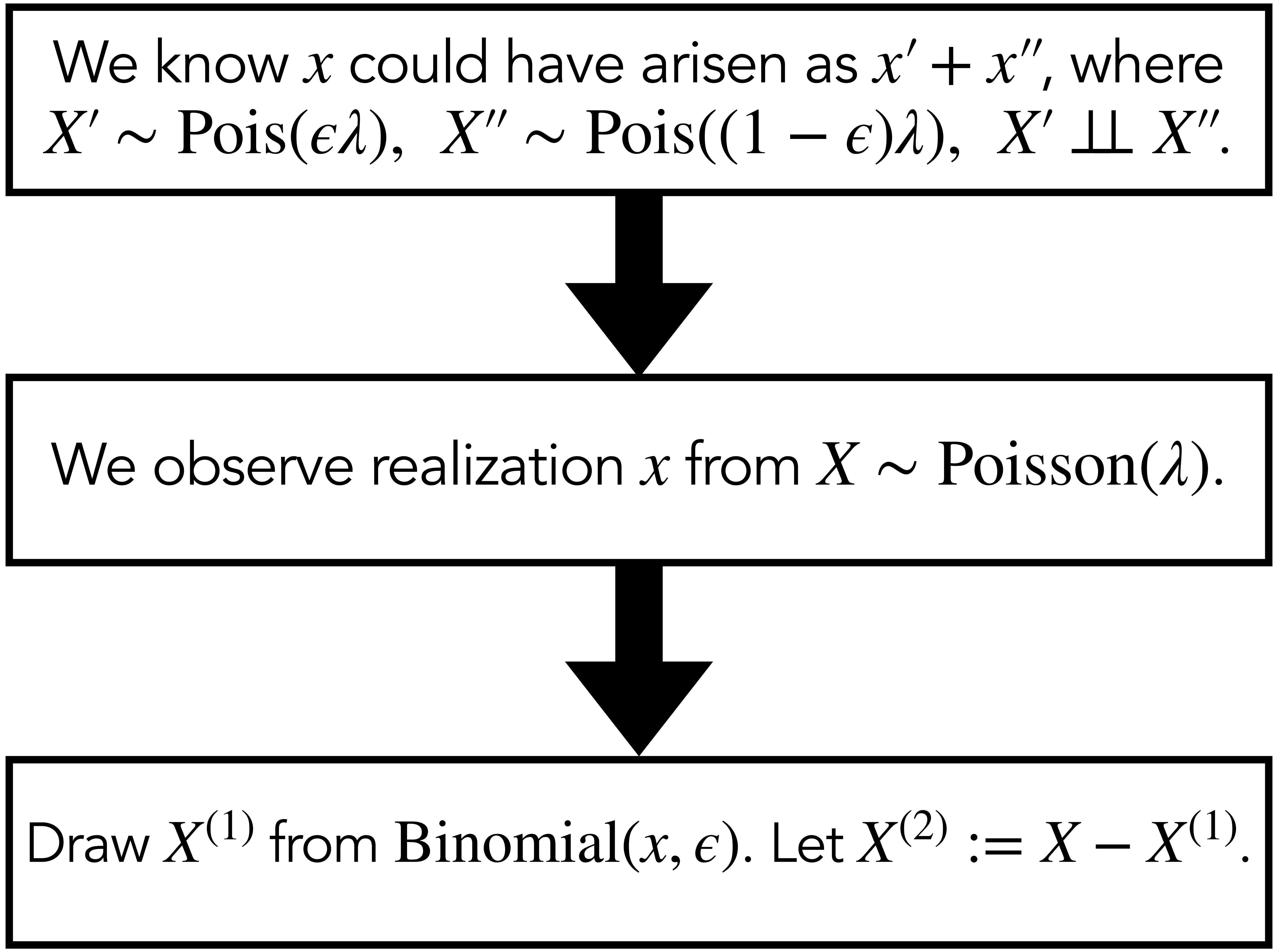
Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\text{Binomial}(x, \epsilon)$.

Theorem:

$X^{(1)} \sim \text{Pois}(\epsilon\lambda),\ X^{(2)} \sim \text{Pois}((1-\epsilon)\lambda),\ \ X^{(1)} \perp\!\!\!\perp X^{(2)}$.

Data thinning for the Poisson distribution



We know x could have arisen as $x' + x''$, where $X' \sim \text{Pois}(\epsilon\lambda),\ \ X'' \sim \text{Pois}((1-\epsilon)\lambda),\ \ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \text{Poisson}(\lambda)$.

Draw $X^{(1)}$ from $\text{Binomial}(x, \epsilon)$. Let $X^{(2)} := X - X^{(1)}$.

Theorem:

$X^{(1)} \sim \text{Pois}(\epsilon\lambda),\ X^{(2)} \sim \text{Pois}((1-\epsilon)\lambda),\ \ X^{(1)} \perp\!\!\!\perp X^{(2)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\text{Binomial}(x, \epsilon)$.

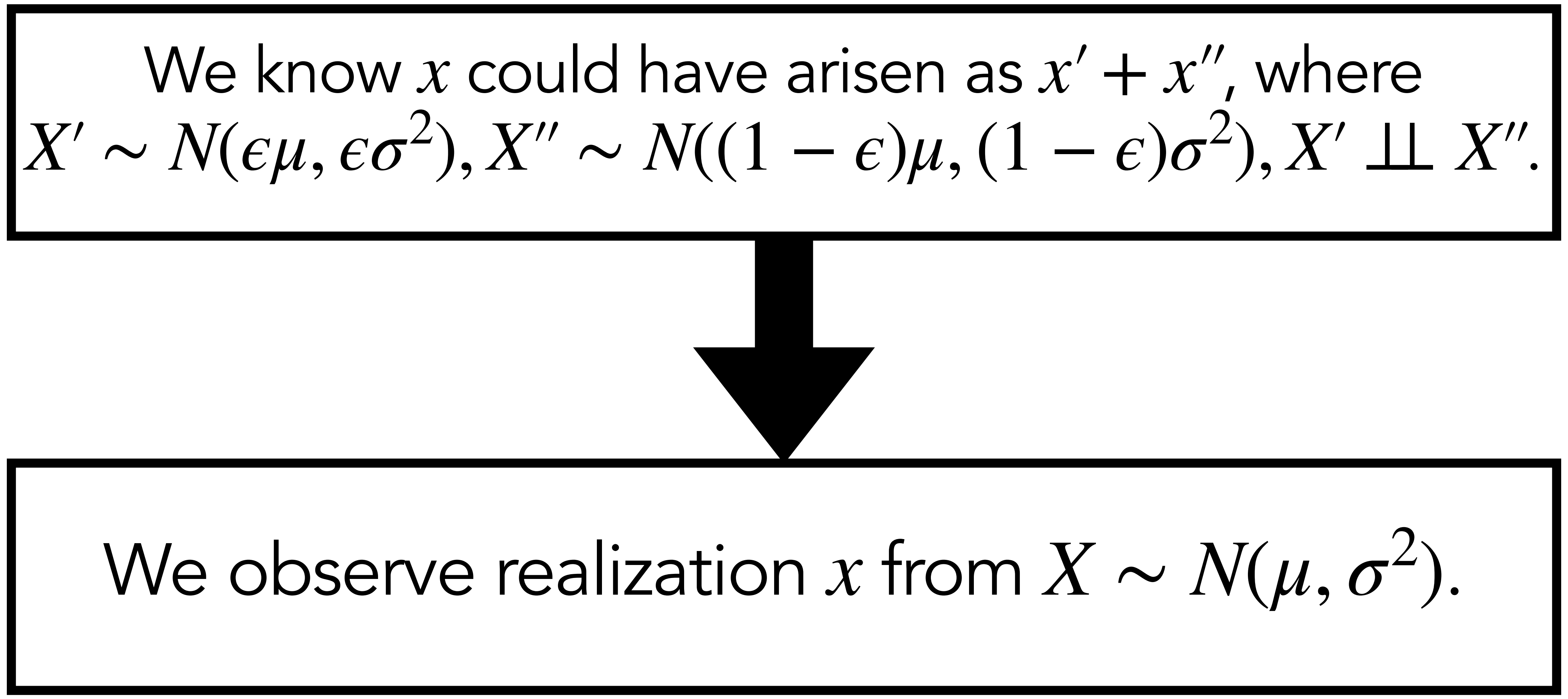
We have recovered Poisson thinning!

Data thinning for the Gaussian distribution

Data thinning for the Gaussian distribution

We observe realization x from $X \sim N(\mu, \sigma^2)$.

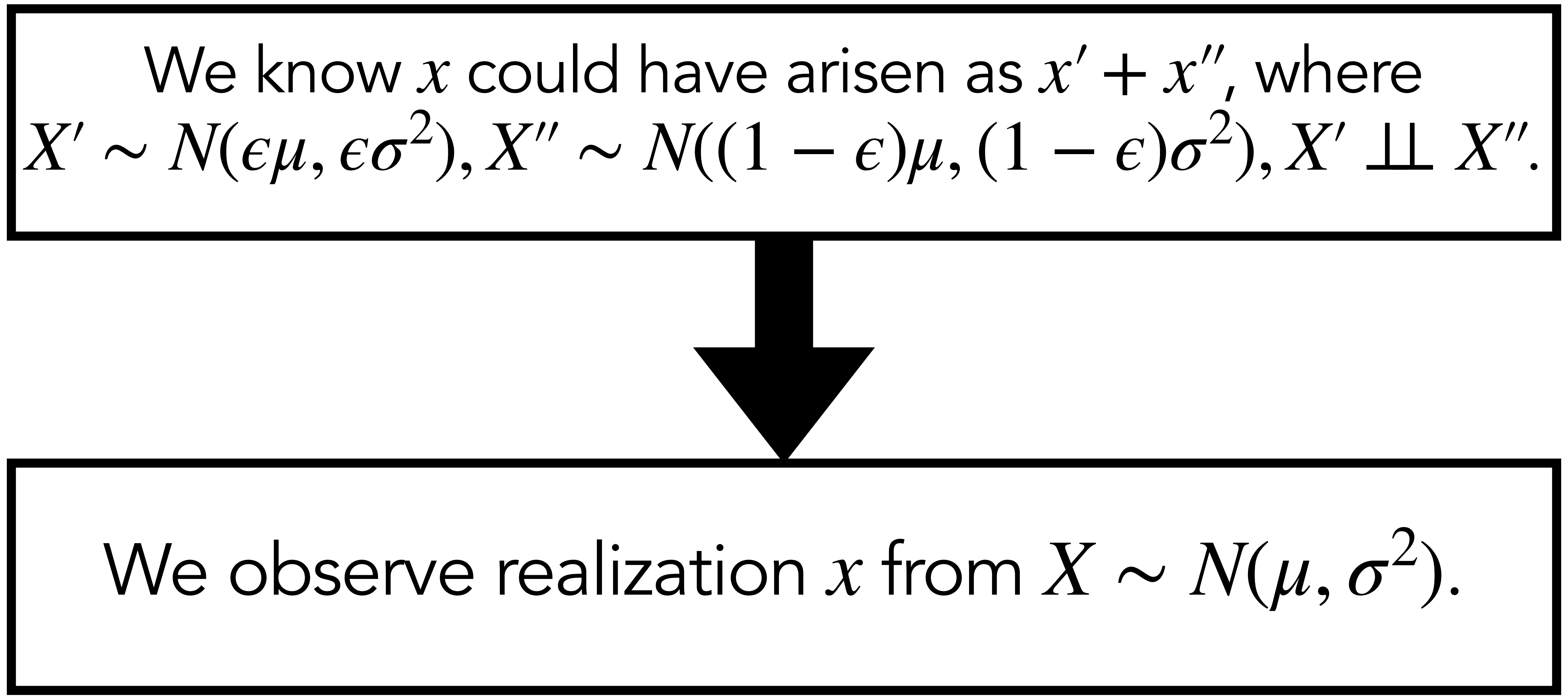
Data thinning for the Gaussian distribution



We know x could have arisen as $x' + x''$, where $X' \sim N(\epsilon\mu, \epsilon\sigma^2), X'' \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim N(\mu, \sigma^2)$.

Data thinning for the Gaussian distribution

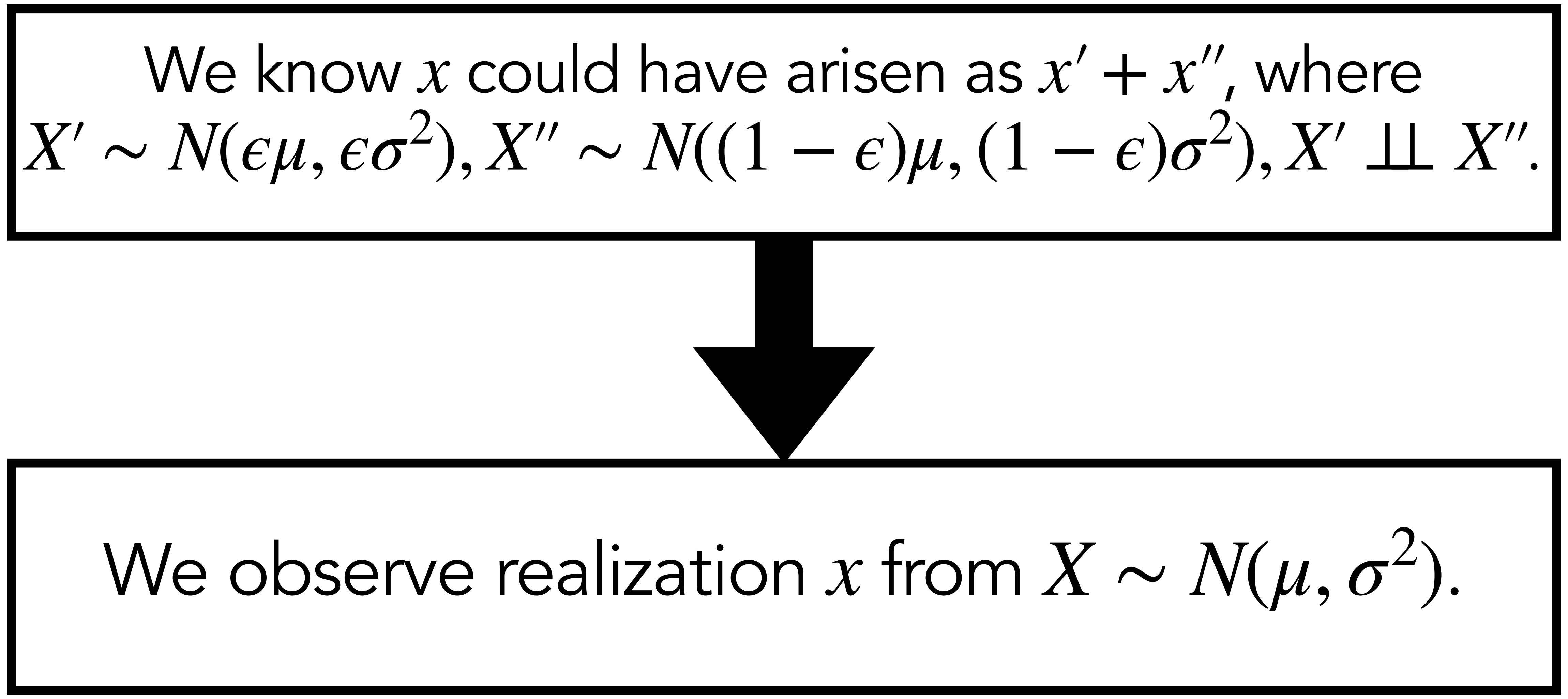


We know x could have arisen as $x' + x''$, where $X' \sim N(\epsilon\mu, \epsilon\sigma^2), X'' \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim N(\mu, \sigma^2)$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Data thinning for the Gaussian distribution



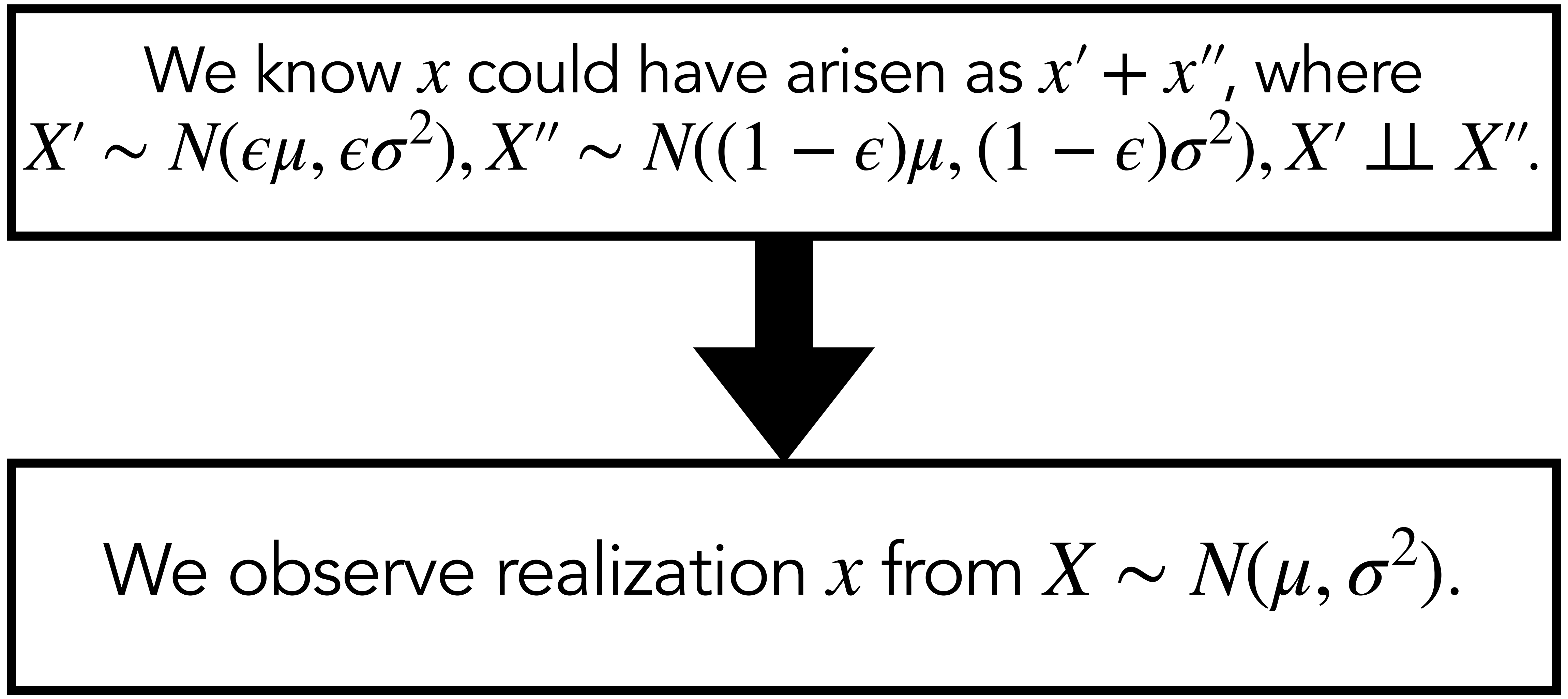
We know x could have arisen as $x' + x''$, where $X' \sim N(\epsilon\mu, \epsilon\sigma^2), X'' \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim N(\mu, \sigma^2)$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

Data thinning for the Gaussian distribution



We know x could have arisen as $x' + x''$, where $X' \sim N(\epsilon\mu, \epsilon\sigma^2), X'' \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X' \perp\!\!\!\perp X''$.

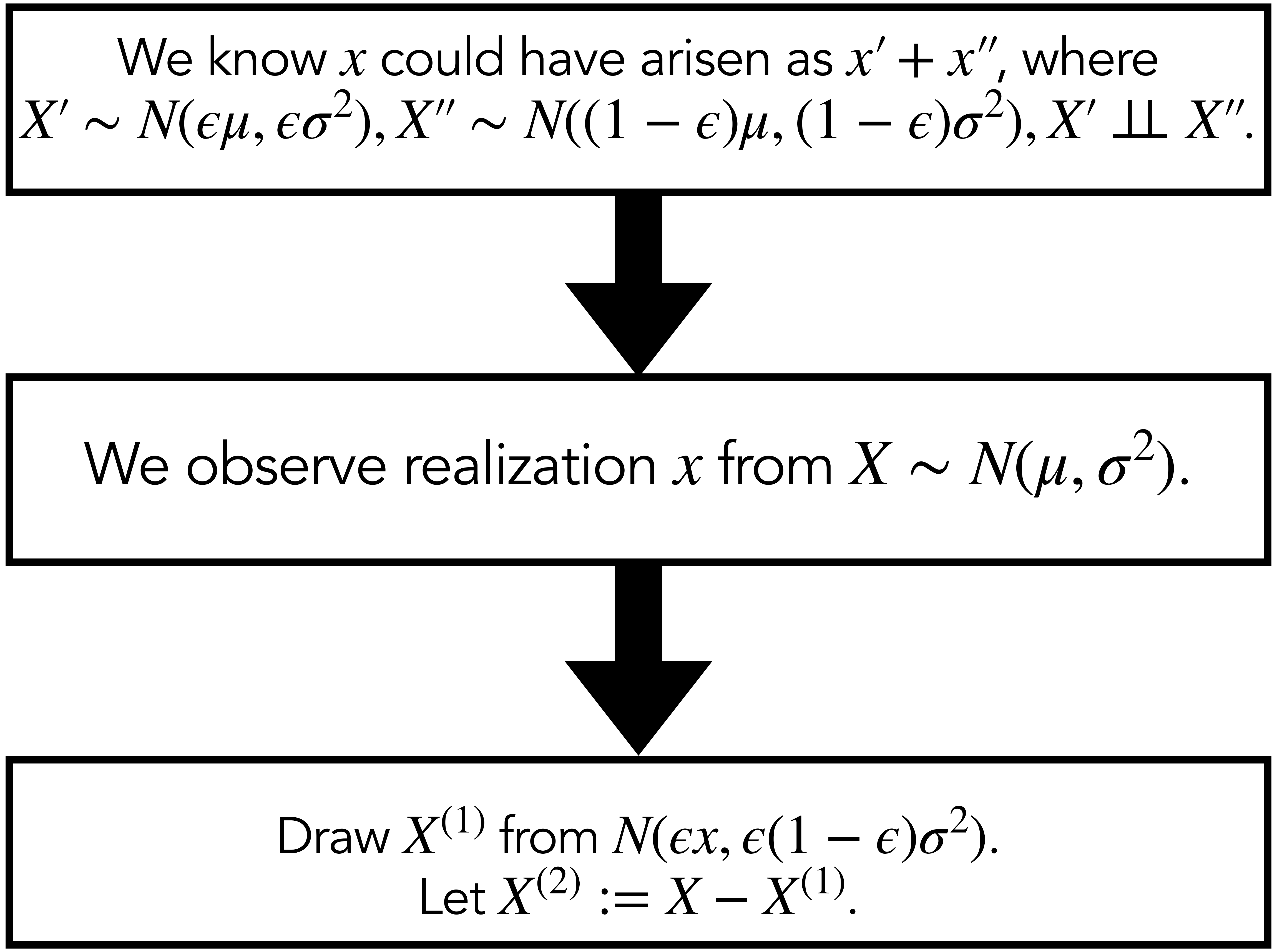
We observe realization x from $X \sim N(\mu, \sigma^2)$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\mathrm{N}(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$.

Data thinning for the Gaussian distribution



We know x could have arisen as $x' + x''$, where $X' \sim N(\epsilon\mu, \epsilon\sigma^2), X'' \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim N(\mu, \sigma^2)$.

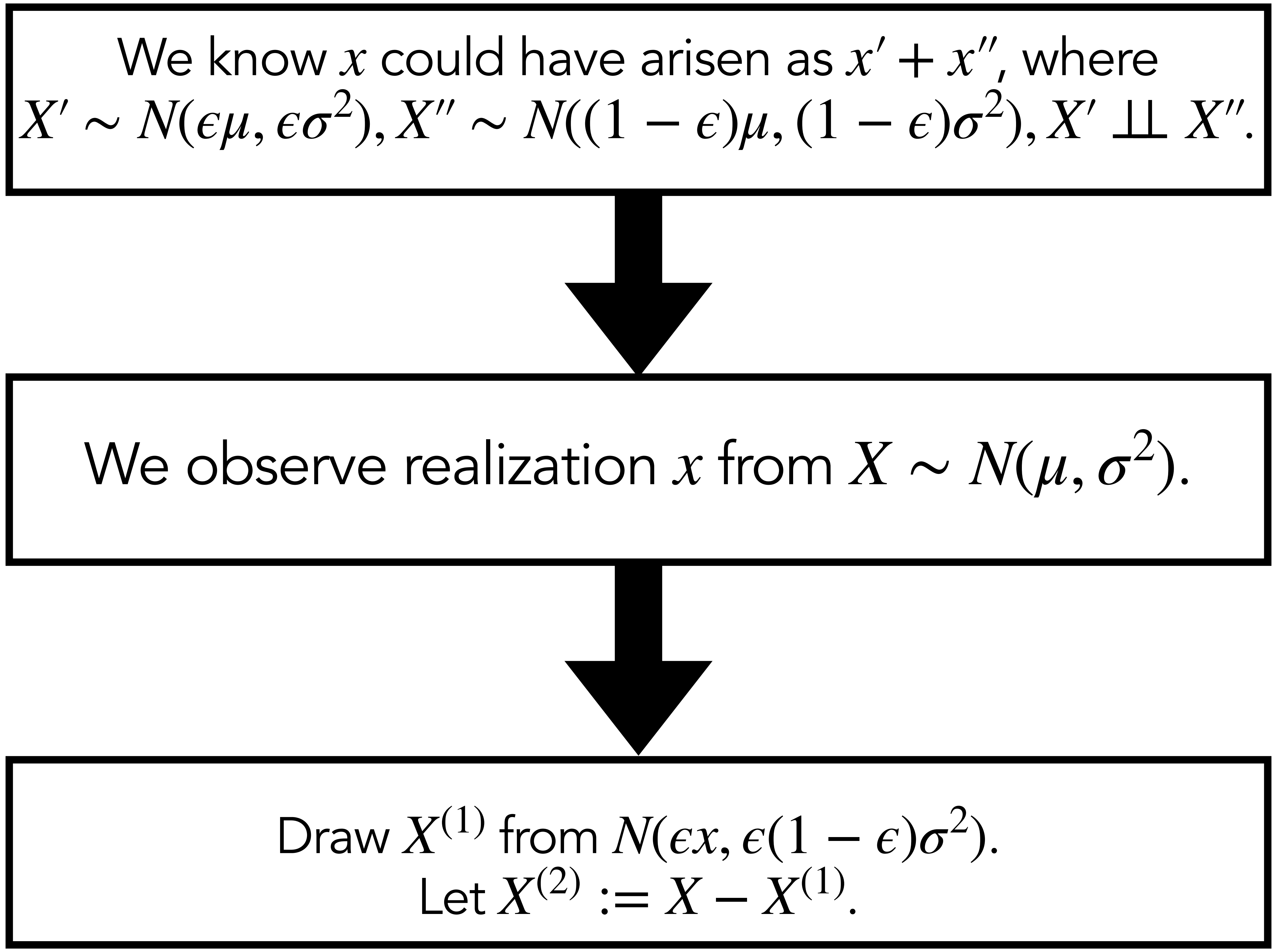
Draw $X^{(1)}$ from $N(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$.
Let $X^{(2)} := X - X^{(1)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\mathrm{N}(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$.

Data thinning for the Gaussian distribution



We know x could have arisen as $x' + x''$, where $X' \sim N(\epsilon\mu, \epsilon\sigma^2), X'' \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X' \perp\!\!\!\perp X''$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

We observe realization x from $X \sim N(\mu, \sigma^2)$.

Can we work backwards to recover x' and x''?

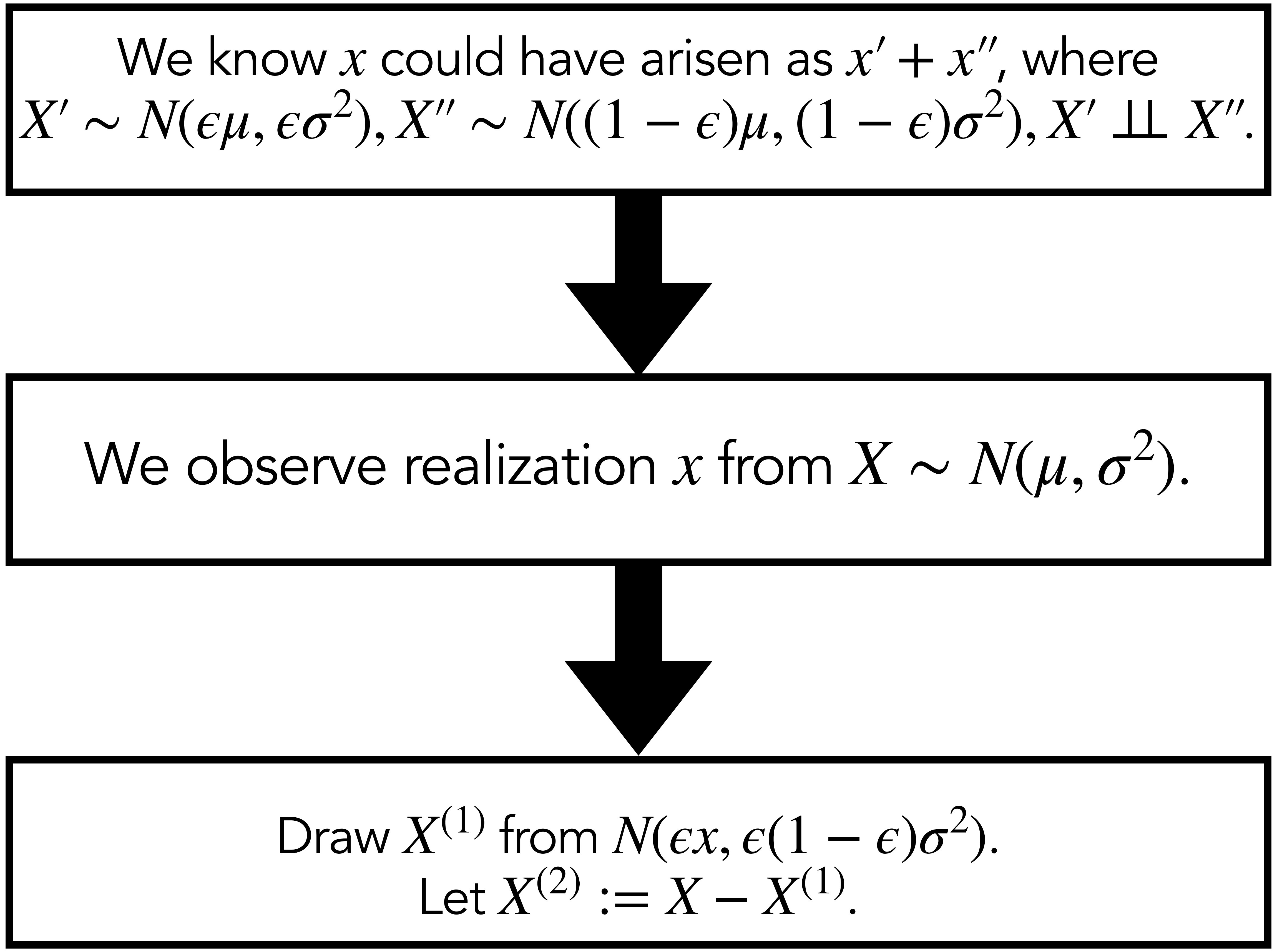
Draw $X^{(1)}$ from $N(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$.
Let $X^{(2)} := X - X^{(1)}$.

The conditional distribution of $X' \mid X = x$ is $\mathrm{N}(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$.

Theorem:

$X^{(1)} \sim N(\epsilon\mu, \epsilon\sigma^2), X^{(2)} \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X^{(1)} \perp\!\!\!\perp X^{(2)}$.

Data thinning for the Gaussian distribution



We know x could have arisen as $x' + x''$, where $X' \sim N(\epsilon\mu, \epsilon\sigma^2), X'' \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim N(\mu, \sigma^2)$.

Draw $X^{(1)}$ from $N(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$.
Let $X^{(2)} := X - X^{(1)}$.

Theorem:

$X^{(1)} \sim N(\epsilon\mu, \epsilon\sigma^2), X^{(2)} \sim N((1-\epsilon)\mu, (1-\epsilon)\sigma^2), X^{(1)} \perp\!\!\!\perp X^{(2)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\mathrm{N}(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$.

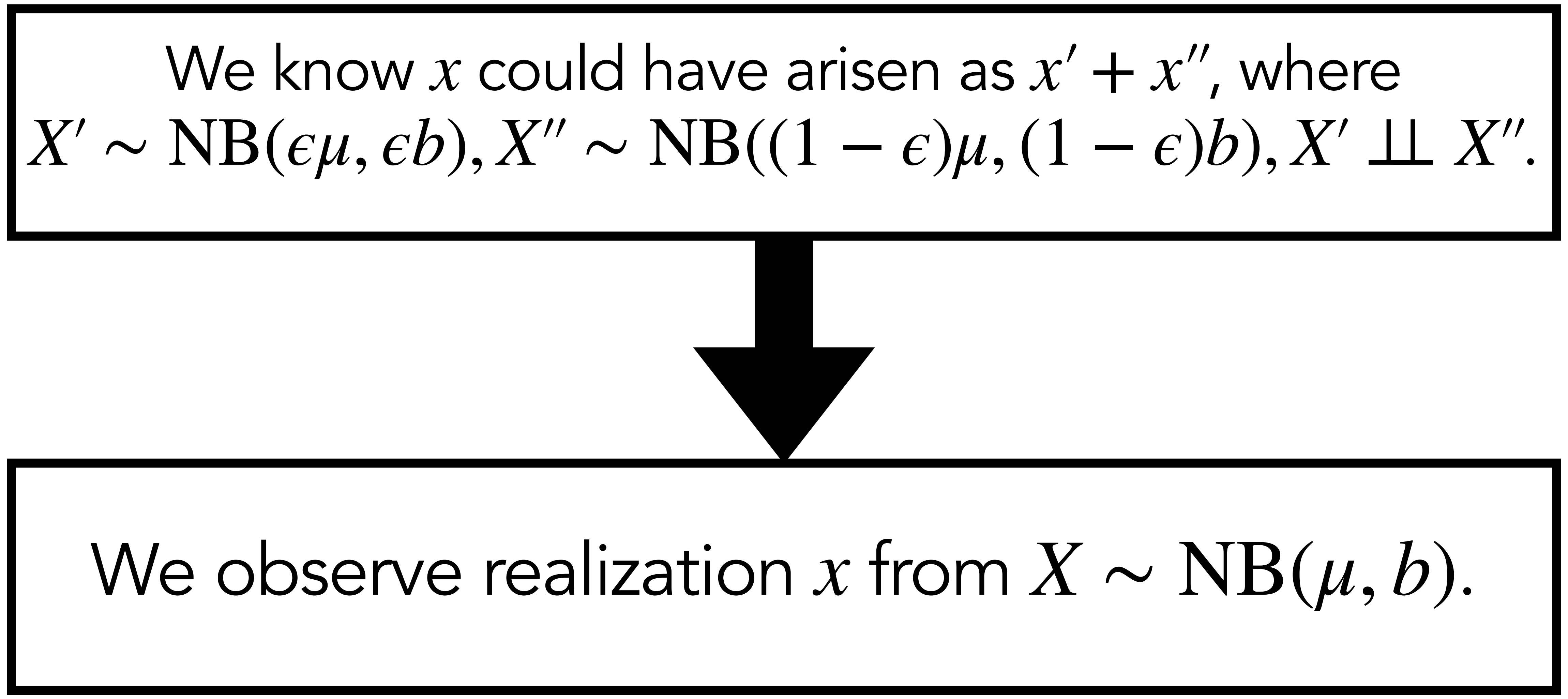
This is (similar to) a well-known result!

Data thinning recipe for the negative binomial distribution

Data thinning recipe for the negative binomial distribution

We observe realization x from $X \sim \text{NB}(\mu, b)$.

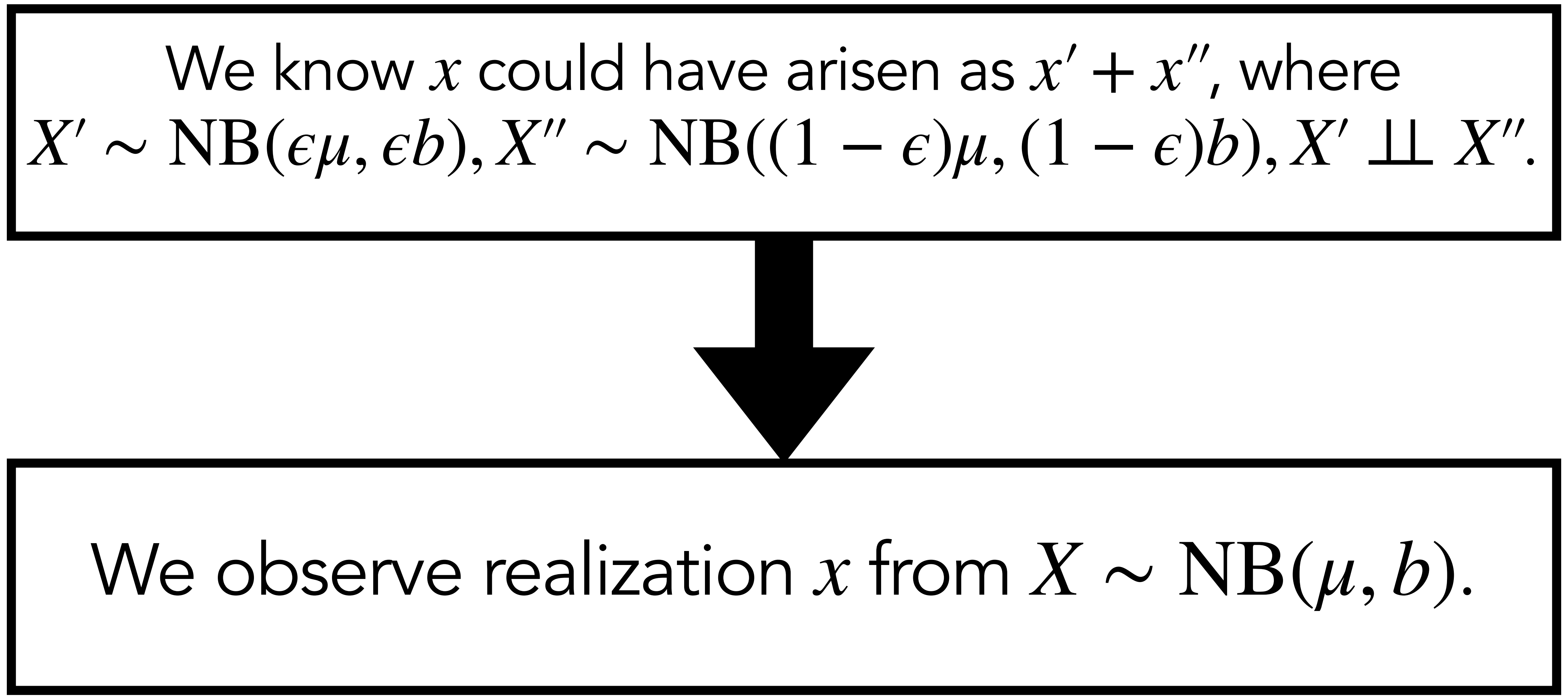
Data thinning recipe for the negative binomial distribution



We know x could have arisen as $x' + x''$, where $X' \sim \mathrm{NB}(\epsilon\mu, \epsilon b), X'' \sim \mathrm{NB}((1-\epsilon)\mu, (1-\epsilon)b), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \mathrm{NB}(\mu, b)$.

Data thinning recipe for the negative binomial distribution

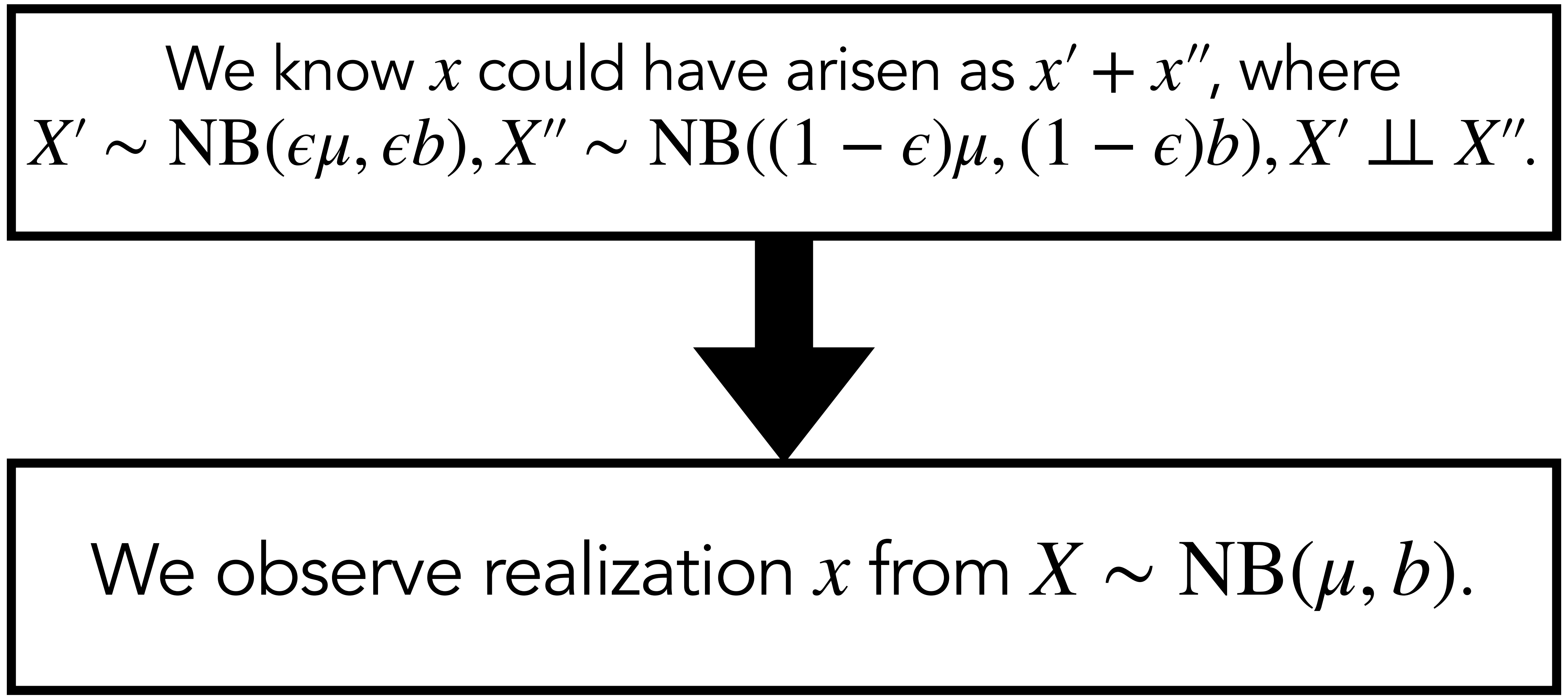


We know x could have arisen as $x' + x''$, where $X' \sim \mathrm{NB}(\epsilon\mu, \epsilon b), X'' \sim \mathrm{NB}((1-\epsilon)\mu, (1-\epsilon)b), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \mathrm{NB}(\mu, b)$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Data thinning recipe for the negative binomial distribution



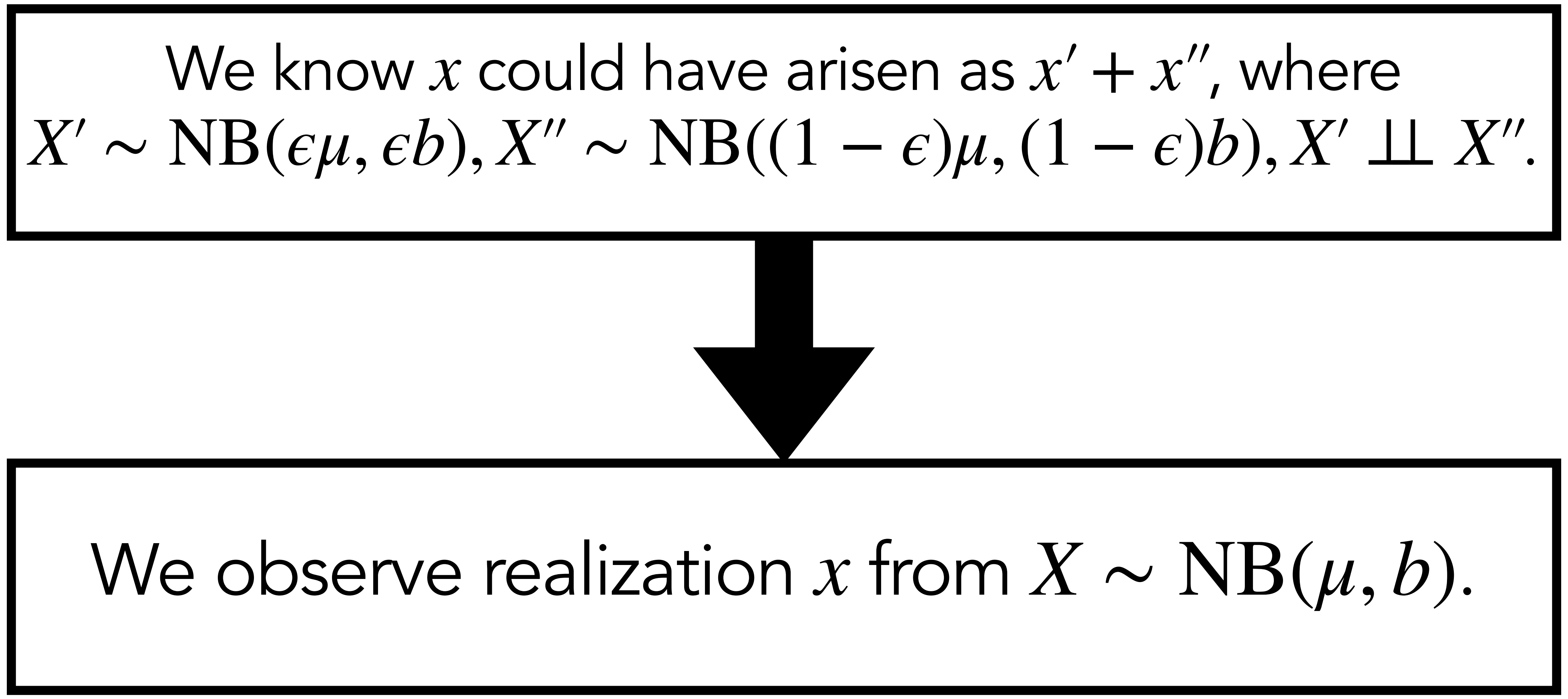
We know x could have arisen as $x' + x''$, where $X' \sim \text{NB}(\epsilon\mu, \epsilon b), X'' \sim \text{NB}((1-\epsilon)\mu, (1-\epsilon)b), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \text{NB}(\mu, b)$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

Data thinning recipe for the negative binomial distribution



We know x could have arisen as $x' + x''$, where $X' \sim \text{NB}(\epsilon\mu, \epsilon b), X'' \sim \text{NB}((1-\epsilon)\mu, (1-\epsilon)b), X' \perp\!\!\!\perp X''$.

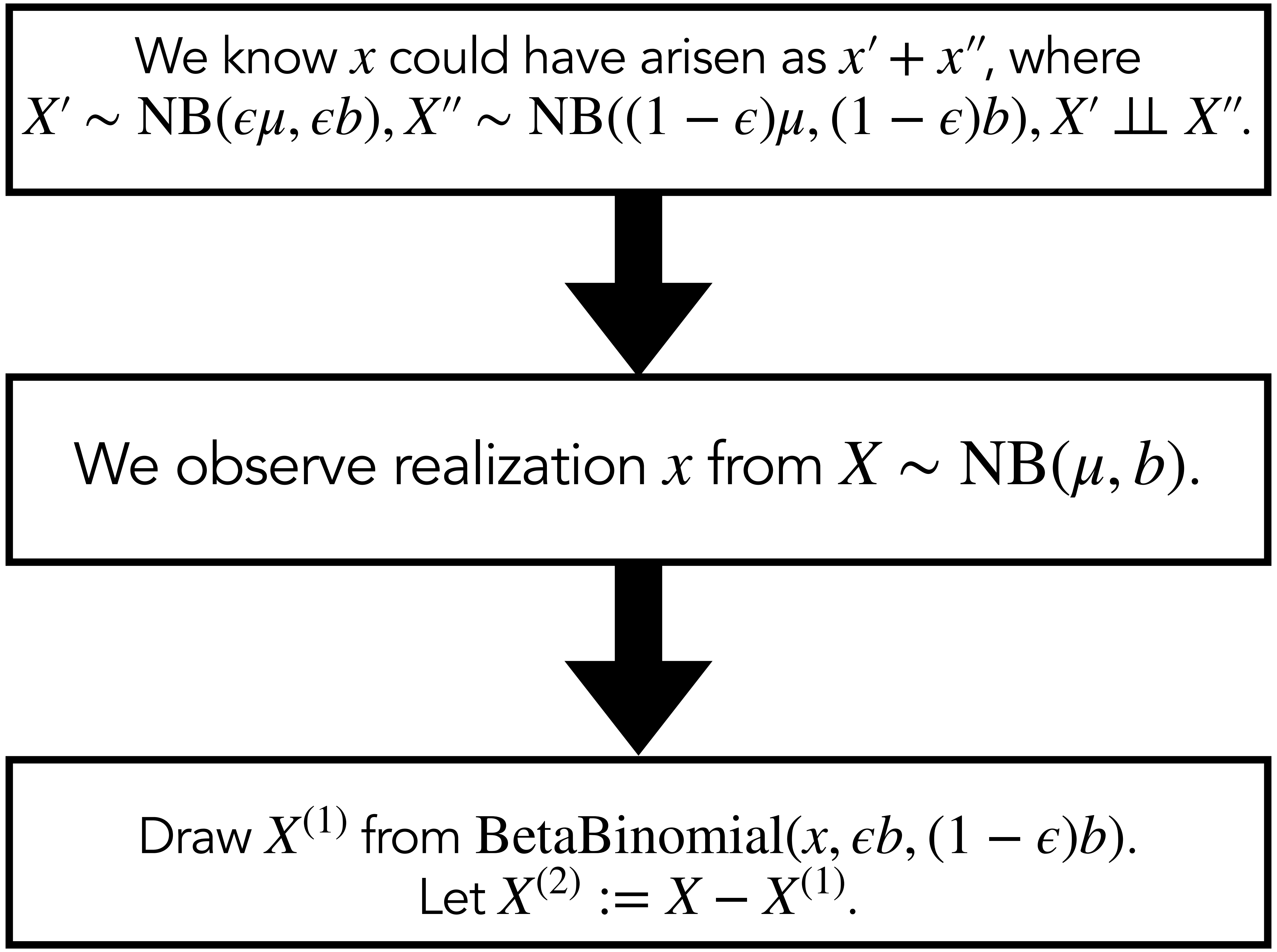
We observe realization x from $X \sim \text{NB}(\mu, b)$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\text{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$.

Data thinning recipe for the negative binomial distribution



We know x could have arisen as $x' + x''$, where $X' \sim \mathrm{NB}(\epsilon\mu, \epsilon b), X'' \sim \mathrm{NB}((1-\epsilon)\mu, (1-\epsilon)b), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \mathrm{NB}(\mu, b)$.

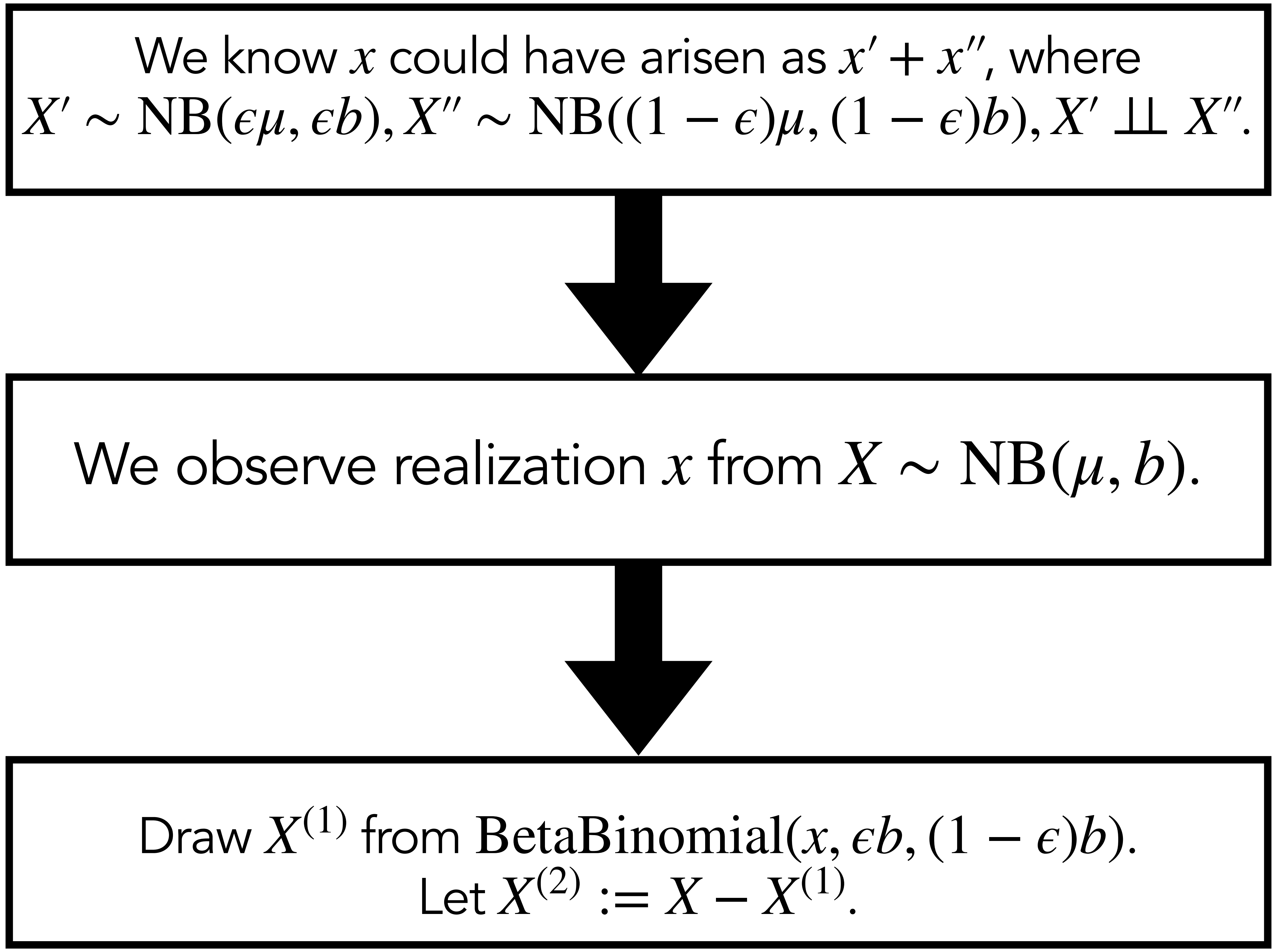
Draw $X^{(1)}$ from $\mathrm{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$.
Let $X^{(2)} := X - X^{(1)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\mathrm{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$.

Data thinning recipe for the negative binomial distribution



We know x could have arisen as $x' + x''$, where $X' \sim \mathrm{NB}(\epsilon\mu, \epsilon b), X'' \sim \mathrm{NB}((1-\epsilon)\mu, (1-\epsilon)b), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \mathrm{NB}(\mu, b)$.

Draw $X^{(1)}$ from $\mathrm{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$.
Let $X^{(2)} := X - X^{(1)}$.

Theorem:

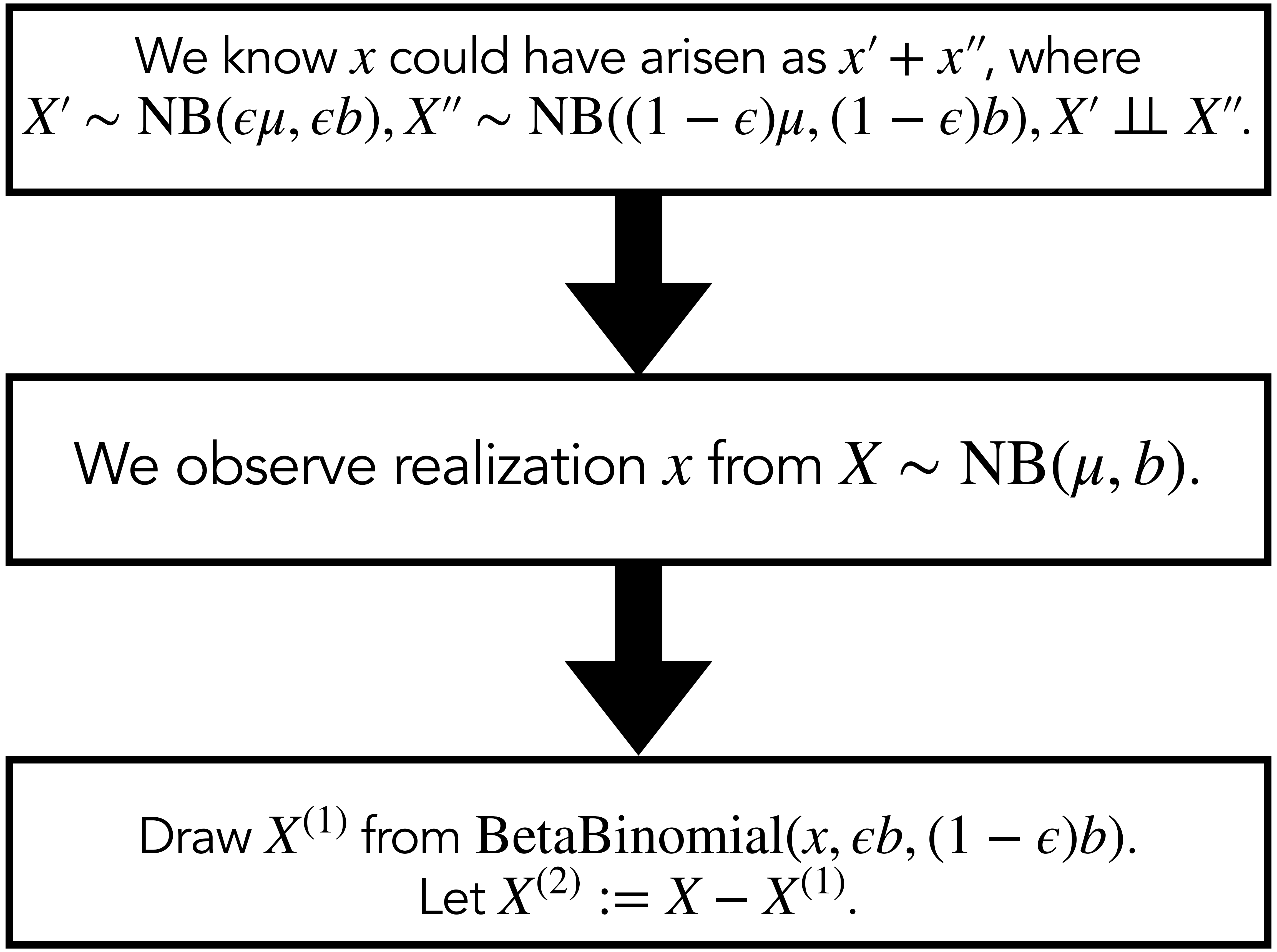
$X^{(1)} \sim \mathrm{NB}(\epsilon\mu, \epsilon b),\ X^{(2)} \sim \mathrm{NB}((1-\epsilon)\mu, (1-\epsilon)b),\ X^{(1)} \perp\!\!\!\perp X^{(2)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\mathrm{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$.

Data thinning recipe for the negative binomial distribution



We know x could have arisen as $x' + x''$, where $X' \sim \mathrm{NB}(\epsilon\mu, \epsilon b), X'' \sim \mathrm{NB}((1-\epsilon)\mu, (1-\epsilon)b), X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim \mathrm{NB}(\mu, b)$.

Draw $X^{(1)}$ from $\mathrm{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$. Let $X^{(2)} := X - X^{(1)}$.

Theorem:

$X^{(1)} \sim \mathrm{NB}(\epsilon\mu, \epsilon b),\ X^{(2)} \sim \mathrm{NB}((1-\epsilon)\mu, (1-\epsilon)b),\ X^{(1)} \perp\!\!\!\perp X^{(2)}$.

If we had observed x' and x'', we would have satisfied our goal of data thinning!

Can we work backwards to recover x' and x''?

The conditional distribution of $X' \mid X = x$ is $\mathrm{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$.

This is a new result!

For many common distributions, the distribution $G_{\epsilon,x}$ has a simple form

Distribution of X:	Draw $X^{(1)} \mid X = x$ from $G_{\epsilon,x}$, where $G_{\epsilon,x}$ is:	Distribution of $X^{(1)}$:	Distribution of $X^{(2)}$, where $X^{(2)} = X - X^{(1)}$:
Poisson(λ)	Binomial(x, ϵ)	Poisson$(\epsilon\lambda)$	Poisson$((1-\epsilon)\lambda)$

For many common distributions, the distribution $G_{\epsilon,x}$ has a simple form

Distribution of X:	Draw $X^{(1)} \mid X = x$ from $G_{\epsilon,x}$, where $G_{\epsilon,x}$ is:	Distribution of $X^{(1)}$:	Distribution of $X^{(2)}$, where $X^{(2)} = X - X^{(1)}$:
Poisson(λ)	Binomial(x, ϵ)	Poisson($\epsilon\lambda$)	Poisson($(1-\epsilon)\lambda$)

Related work on Poisson thinning:

- Sarkar and Stephens, 2021, Nature Genetics.
- Chen et al., 2021, arXiv:2108.03336
- Leiner et al., 2021, arXiv:2112.11079.
- Neufeld et al., 2022, Biostatistics.
- Oliveira, Lei, and Tibshirani, 2022, arXiv:2212.01943.

For many common distributions, the distribution $G_{\epsilon,x}$ has a simple form

Distribution of X:	Draw $X^{(1)} \mid X = x$ from $G_{\epsilon,x}$, where $G_{\epsilon,x}$ is:	Distribution of $X^{(1)}$:	Distribution of $X^{(2)}$, where $X^{(2)} = X - X^{(1)}$:
$\text{Poisson}(\lambda)$	$\text{Binomial}(x, \epsilon)$	$\text{Poisson}(\epsilon\lambda)$	$\text{Poisson}((1-\epsilon)\lambda)$
$\text{N}(\mu, \sigma^2)$	$\text{N}(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$	$\text{N}(\epsilon\mu, \epsilon\sigma^2)$	$\text{N}((1-\epsilon)\mu, (1-\epsilon)\sigma^2)$

For many common distributions, the distribution $G_{\epsilon,x}$ has a simple form

Distribution of X:	Draw $X^{(1)} \mid X = x$ from $G_{\epsilon,x}$, where $G_{\epsilon,x}$ is:	Distribution of $X^{(1)}$:	Distribution of $X^{(2)}$, where $X^{(2)} = X - X^{(1)}$:
$\text{Poisson}(\lambda)$	$\text{Binomial}(x, \epsilon)$	$\text{Poisson}(\epsilon\lambda)$	$\text{Poisson}((1-\epsilon)\lambda)$
$\text{N}(\mu, \sigma^2)$	$\text{N}(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$	$\text{N}(\epsilon\mu, \epsilon\sigma^2)$	$\text{N}((1-\epsilon)\mu, (1-\epsilon)\sigma^2)$

Related work on Gaussian thinning:

- Tian and Taylor, 2018, Annals of Statistics.
- Tian, 2020, Annals of Statistics.
- Rasines and Young, 2022, Biometrika.
- Leiner et al., 2022, arXiv:2112.11079.
- Oliveira, Lei, and Tibshirani, 2022, arXiv:2111.09447.

For many common distributions, the distribution $G_{\epsilon,x}$ has a simple form

Distribution of X:	Draw $X^{(1)} \mid X = x$ from $G_{\epsilon,x}$, where $G_{\epsilon,x}$ is:	Distribution of $X^{(1)}$:	Distribution of $X^{(2)}$, where $X^{(2)} = X - X^{(1)}$:
$\text{Poisson}(\lambda)$	$\text{Binomial}(x, \epsilon)$	$\text{Poisson}(\epsilon\lambda)$	$\text{Poisson}((1-\epsilon)\lambda)$
$\text{N}(\mu, \sigma^2)$	$\text{N}(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$	$\text{N}(\epsilon\mu, \epsilon\sigma^2)$	$\text{N}((1-\epsilon)\mu, (1-\epsilon)\sigma^2)$
$\text{NegativeBinomial}(\mu, b)$	$\text{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$.	$\text{NegativeBinomial}(\epsilon\mu, \epsilon b)$	$\text{NegativeBinomial}((1-\epsilon)\mu, (1-\epsilon)b)$

For many common distributions, the distribution $G_{\epsilon,x}$ has a simple form

Distribution of X:	Draw $X^{(1)} \mid X = x$ from $G_{\epsilon,x}$, where $G_{\epsilon,x}$ is:	Distribution of $X^{(1)}$:	Distribution of $X^{(2)}$, where $X^{(2)} = X - X^{(1)}$:
$\text{Poisson}(\lambda)$	$\text{Binomial}(x, \epsilon)$	$\text{Poisson}(\epsilon\lambda)$	$\text{Poisson}((1-\epsilon)\lambda)$
$\text{N}(\mu, \sigma^2)$	$\text{N}(\epsilon x, \epsilon(1-\epsilon)\sigma^2)$	$\text{N}(\epsilon\mu, \epsilon\sigma^2)$	$\text{N}((1-\epsilon)\mu, (1-\epsilon)\sigma^2)$
$\text{NegativeBinomial}(\mu, b)$	$\text{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$.	$\text{NegativeBinomial}(\epsilon\mu, \epsilon b)$	$\text{NegativeBinomial}((1-\epsilon)\mu, (1-\epsilon)b)$
$\text{Binomial}(r, p)$	$\text{Hypergeometric}(\epsilon r, (1-\epsilon)r, x)$.	$\text{Binomial}(\epsilon r, p)$	$\text{Binomial}((1-\epsilon)r, p)$
$\text{Gamma}(\alpha, \beta)$	$x \cdot \text{Beta}\,(\epsilon\alpha, (1-\epsilon)\alpha)$.	$\text{Gamma}(\epsilon\alpha, \beta)$	$\text{Gamma}((1-\epsilon)\alpha, \beta)$
$\text{Exponential}(\lambda)$	$x \cdot \text{Beta}(\epsilon, (1-\epsilon))$.	$\text{Gamma}(\epsilon, \lambda)$	$\text{Gamma}(1-\epsilon, \lambda)$
$\text{N}_k(\mu, \Sigma)$	$\text{N}(\epsilon x, \epsilon(1-\epsilon)\Sigma)$.	$\text{N}_k(\epsilon\mu, \epsilon\Sigma)$	$\text{N}_k((1-\epsilon)\mu, (1-\epsilon)\Sigma)$
$\text{Multinomial}_k(r, p)$	$\text{MultivarHypergeom}(x_1, \ldots, x_K, \epsilon r)$	$\text{Multinom}_k(\epsilon r, p)$	$\text{Multinomial}_k((1-\epsilon)r, p)$
$\text{Wishart}_p(n, \Sigma)$.	$x^{1/2} Z x^{1/2}$, where . $Z \sim \text{MatrixBeta}_p(\epsilon n/2, (1-\epsilon)n/2)$	$\text{Wishart}_p(\epsilon n, \Sigma)$	$\text{Wishart}_p((1-\epsilon)n, \Sigma)$

What if we get a nuisance parameter wrong?

Negative binomial thinning algorithm

Suppose $X \sim \text{NegBin}\left(\mu, b\right)$.

Draw

$X^{(1)} \sim \text{BetaBinomial}(x, \epsilon b, (1-\epsilon)b)$,

$X^{(2)} = X - X^{(1)}$, then:

1) $X^{(1)} \sim \text{NegBin}\left(\epsilon\mu, \epsilon b\right)$.
2) $X^{(2)} \sim \text{NegBin}\left((1-\epsilon)\mu, (1-\epsilon)b\right)$
3) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

What if we get a nuisance parameter wrong?

Negative binomial thinning algorithm

Suppose $X \sim \text{NegBin}\left(\mu, b\right)$.

Draw

$X^{(1)} \sim \text{BetaBinomial}(x, \epsilon\tilde{b}, (1-\epsilon)\tilde{b})$,

$X^{(2)} = X - X^{(1)}$, then:

1) $X^{(1)} \sim \text{NegBin}\left(\epsilon\mu, \epsilon b\right)$.
2) $X^{(2)} \sim \text{NegBin}\left((1-\epsilon)\mu, (1-\epsilon)b\right)$
3) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

What if we get a nuisance parameter wrong?

Negative binomial thinning algorithm

Suppose $X \sim \text{NegBin}\left(\mu, b\right)$.

Draw

$X^{(1)} \sim \text{BetaBinomial}(x, \epsilon\tilde{b}, (1-\epsilon)\tilde{b}),$

$X^{(2)} = X - X^{(1)}$, then:

~~1) $X^{(1)} \sim \text{NegBin}\left(\epsilon\mu, \epsilon b\right)$.~~

~~2) $X^{(2)} \sim \text{NegBin}\left((1-\epsilon)\mu, (1-\epsilon)b\right)$~~

~~3) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.~~

What if we get a nuisance parameter wrong?

Negative binomial thinning algorithm

Suppose $X \sim \text{NegBin}\left(\mu, b\right)$.

Draw

$X^{(1)} \sim \text{BetaBinomial}(x, \epsilon\tilde{b}, (1-\epsilon)\tilde{b}\ ),$

$X^{(2)} = X - X^{(1)}$, then:

1) $\text{E}[X^{(1)}] = \epsilon\mu.$
2) $\text{E}[X^{(2)}] = (1-\epsilon)\mu$
3) $\text{Cov}\left(X^{(1)}, X^{(2)}\right) = \epsilon(1-\epsilon)\frac{\mu^2}{b}\left(1 - \frac{b+1}{\tilde{b}+1}\right).$

What if we get a nuisance parameter wrong?

Negative binomial thinning algorithm

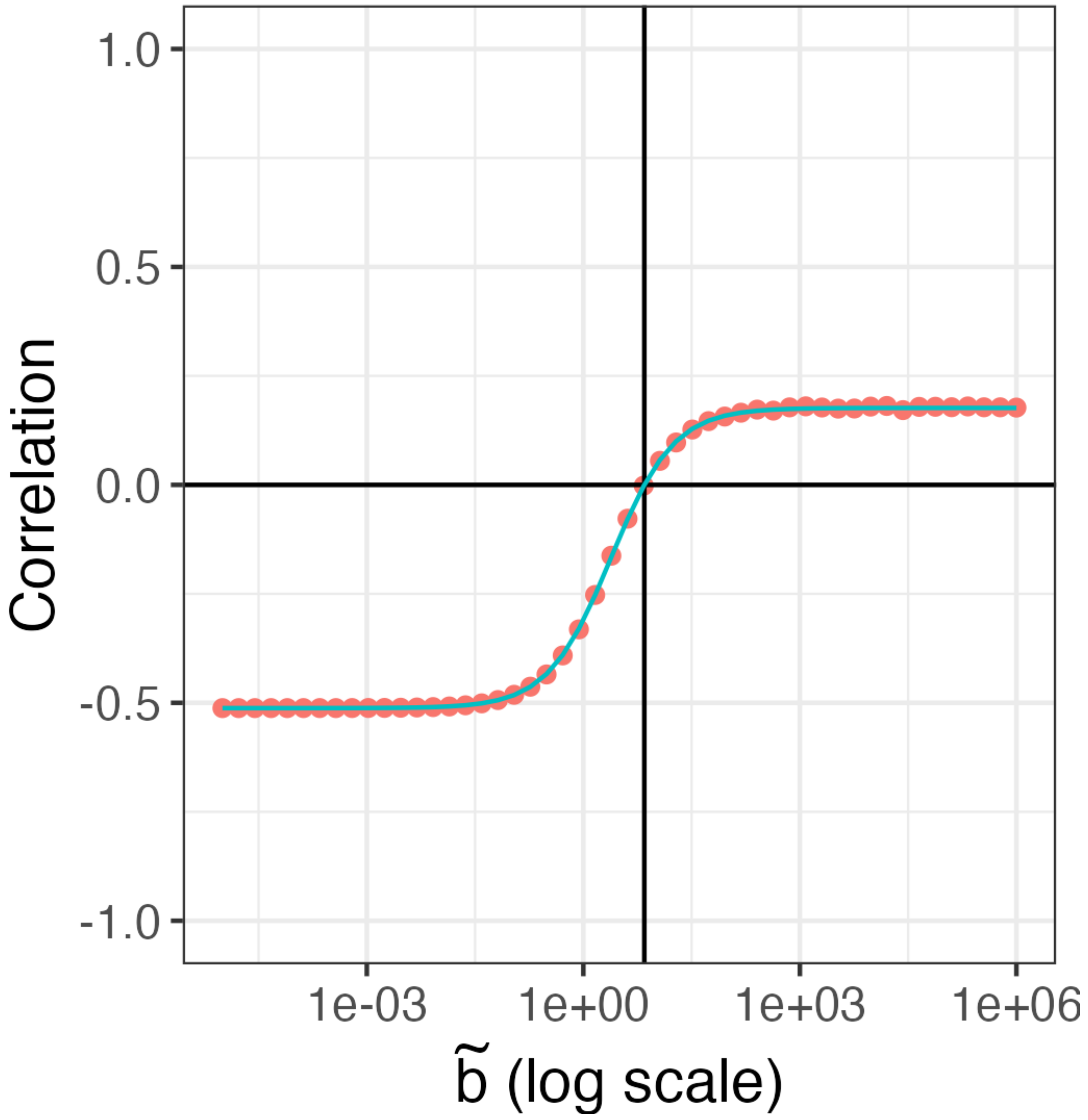
Suppose $X \sim \mathrm{NegBin}\left(\mu, b\right)$.

Draw

$X^{(1)} \sim \mathrm{BetaBinomial}(x, \epsilon\tilde{b}, (1-\epsilon)\tilde{b}\,)$,

$X^{(2)} = X - X^{(1)}$, then:

1) $\mathrm{E}[X^{(1)}] = \epsilon\mu.$
2) $\mathrm{E}[X^{(2)}] = (1-\epsilon)\mu$
3) $\mathrm{Cov}\left(X^{(1)}, X^{(2)}\right) = \epsilon(1-\epsilon)\frac{\mu^2}{b}\left(1 - \frac{b+1}{\tilde{b}+1}\right).$



What if we get a nuisance parameter wrong?

Negative binomial thinning algorithm

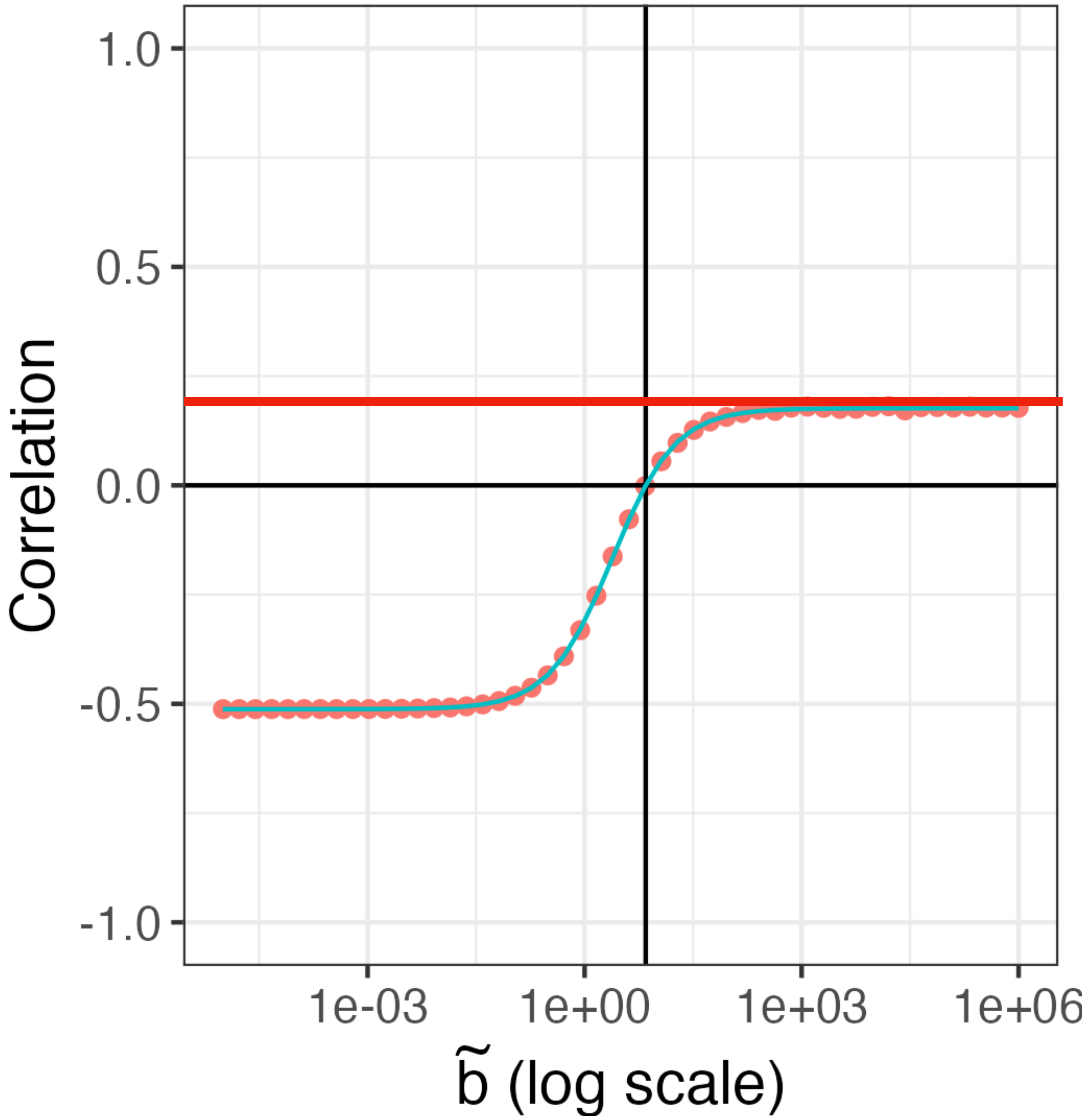
Suppose $X \sim \text{NegBin}\left(\mu, b\right)$.

Draw

$X^{(1)} \sim \text{BetaBinomial}(x, \epsilon\tilde{b}, (1-\epsilon)\tilde{b}\ )$,

$X^{(2)} = X - X^{(1)}$, then:

1) $\text{E}[X^{(1)}] = \epsilon\mu.$
2) $\text{E}[X^{(2)}] = (1-\epsilon)\mu$
3) $\text{Cov}\left(X^{(1)}, X^{(2)}\right) = \epsilon(1-\epsilon)\frac{\mu^2}{b}\left(1 - \frac{b+1}{\tilde{b}+1}\right).$



Corresponds to assuming that the data are Poisson.

What if we get a nuisance parameter wrong?

Negative binomial thinning algorithm

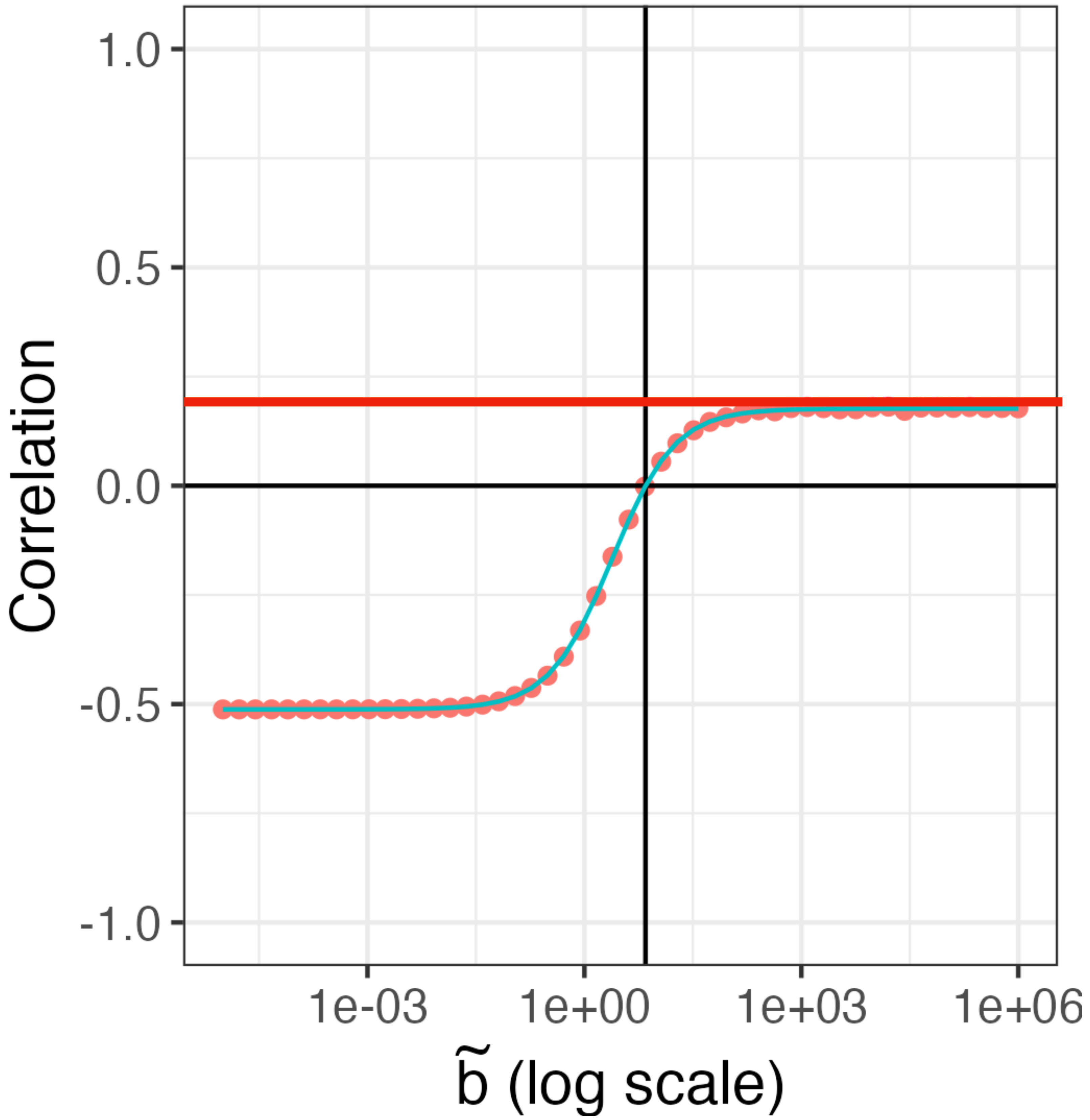
Suppose $X \sim \text{NegBin}\left(\mu, b\right)$.

Draw

$X^{(1)} \sim \text{BetaBinomial}(x, \epsilon\tilde{b}, (1-\epsilon)\tilde{b}\ )$,

$X^{(2)} = X - X^{(1)}$, then:

1) $\text{E}[X^{(1)}] = \epsilon\mu.$
2) $\text{E}[X^{(2)}] = (1-\epsilon)\mu$
3) $\text{Cov}\left(X^{(1)}, X^{(2)}\right) = \epsilon(1-\epsilon)\frac{\mu^2}{b}\left(1 - \frac{b+1}{\tilde{b}+1}\right).$



Corresponds to assuming that the data are Poisson.

Similar results can be derived for other decompositions.

The parameter ϵ governs an information tradeoff

Gaussian thinning algorithm

Suppose $X \sim \mathrm{N}\left(\mu, \sigma^2\right)$.

Draw

$X^{(1)} \sim \mathrm{N}(\epsilon x,\ \epsilon(1-\epsilon)\sigma^2)$ and

$X^{(2)} = X - X^{(1)}$.

Then:

1) $X^{(1)} \sim \mathrm{N}\left(\epsilon\mu, \epsilon\sigma^2\right)$
2) $X^{(2)} \sim \mathrm{N}\left((1-\epsilon)\mu, (1-\epsilon)\sigma^2\right)$
3) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

The parameter ϵ governs an information tradeoff

Gaussian thinning algorithm

Suppose $X \sim \mathrm{N}\left(\mu, \sigma^2\right)$.

Draw

$X^{(1)} \sim \mathrm{N}(\epsilon x,\ \epsilon(1-\epsilon)\sigma^2)$ and

$X^{(2)} = X - X^{(1)}$.

Then:

1) $X^{(1)} \sim \mathrm{N}\left(\epsilon\mu, \epsilon\sigma^2\right)$
2) $X^{(2)} \sim \mathrm{N}\left((1-\epsilon)\mu, (1-\epsilon)\sigma^2\right)$
3) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

Theorem: If we data thin with parameter ϵ, the Fisher information in X about μ is divided between $X^{(1)}$ and $X^{(2)}$ with proportions ϵ and $1-\epsilon$.

The parameter ϵ governs an information tradeoff

Gaussian thinning algorithm

Suppose $X \sim \mathrm{N}\left(\mu, \sigma^2\right)$.

Draw

$X^{(1)} \sim \mathrm{N}(\epsilon x,\ \epsilon(1-\epsilon)\sigma^2)$ and

$X^{(2)} = X - X^{(1)}$.

Then:

1) $X^{(1)} \sim \mathrm{N}\left(\epsilon\mu, \epsilon\sigma^2\right)$
2) $X^{(2)} \sim \mathrm{N}\left((1-\epsilon)\mu, (1-\epsilon)\sigma^2\right)$
3) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

Theorem: If we data thin with parameter ϵ, the Fisher information in X about μ is divided between $X^{(1)}$ and $X^{(2)}$ with proportions ϵ and $1-\epsilon$.

Similar results can be derived for other decompositions.

Our recipe extends naturally to splitting into M>2 folds

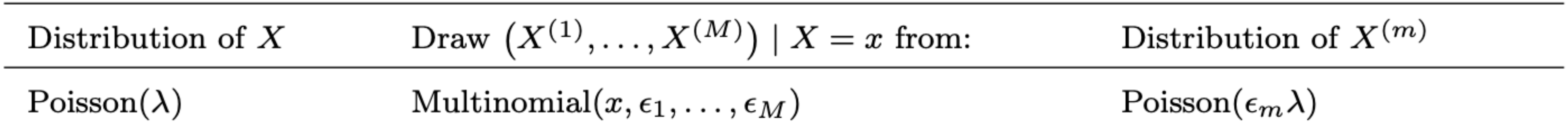
Our recipe extends naturally to splitting into M>2 folds

Goal: split a single observation X into $\left(X^{(1)}, \ldots, X^{(M)}\right)$ such that:

(1) Each $X^{(m)}$ has the same distribution as X, up to a parameter scaling.

(2) The $X^{(m)}$ are mutually independent.

Our recipe extends naturally to splitting into M>2 folds



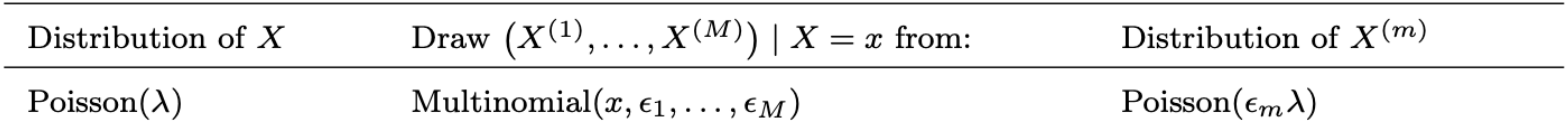
Distribution of X	Draw $(X^{(1)},\ldots,X^{(M)}) \mid X = x$ from:	Distribution of $X^{(m)}$
$\mathrm{Poisson}(\lambda)$	$\mathrm{Multinomial}(x,\epsilon_1,\ldots,\epsilon_M)$	$\mathrm{Poisson}(\epsilon_m\lambda)$

Goal: split a single observation X into $\left(X^{(1)},\ldots,X^{(M)}\right)$ such that:

(1) Each $X^{(m)}$ has the same distribution as X, up to a parameter scaling.

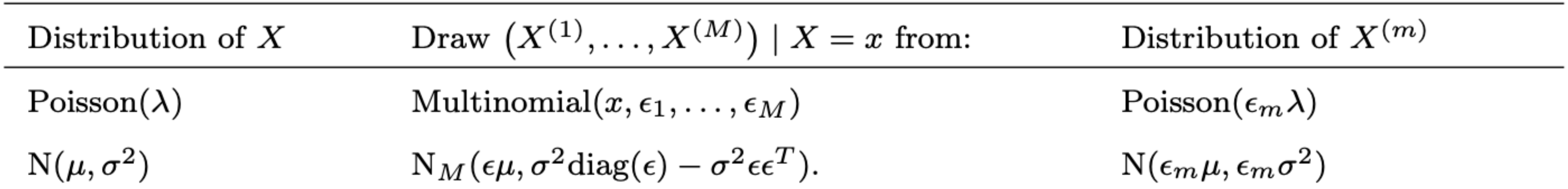
(2) The $X^{(m)}$ are mutually independent.

Our recipe extends naturally to splitting into M>2 folds



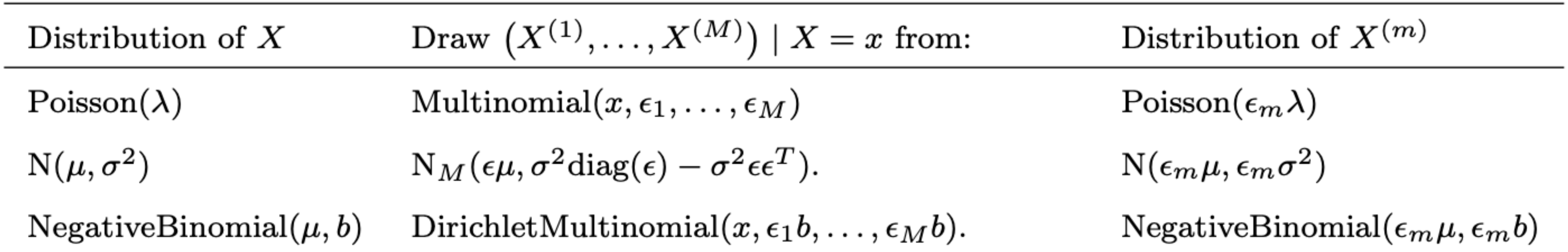
Distribution of X	Draw $\left(X^{(1)},\ldots,X^{(M)}\right) \mid X = x$ from:	Distribution of $X^{(m)}$
Poisson(λ)	Multinomial($x, \epsilon_1, \ldots, \epsilon_M$)	Poisson($\epsilon_m \lambda$)

Our recipe extends naturally to splitting into M>2 folds



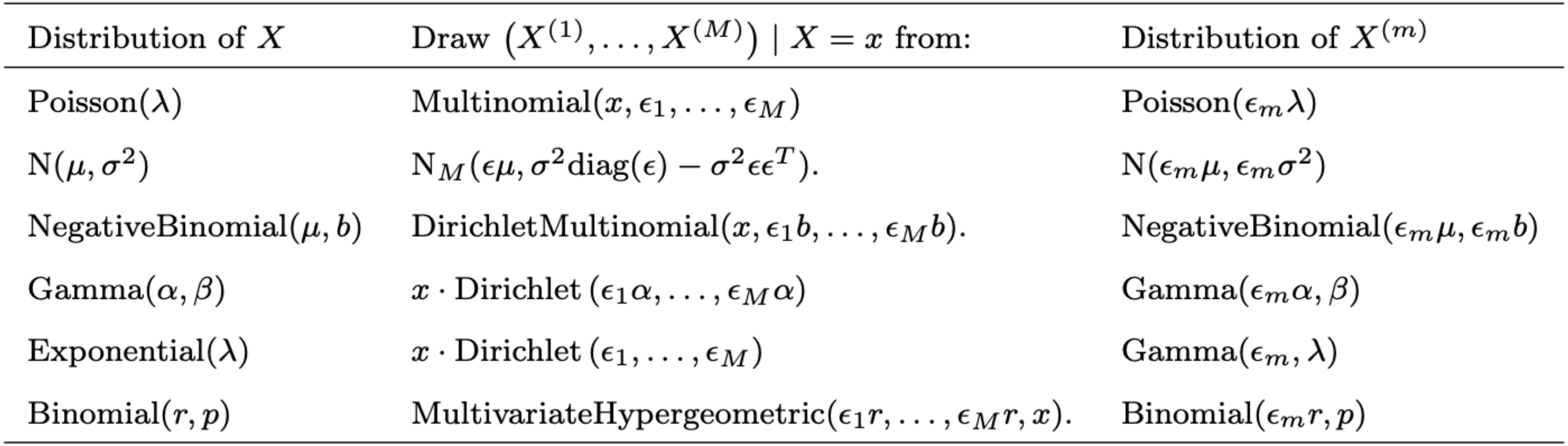
Distribution of X	Draw $\left(X^{(1)}, \ldots, X^{(M)}\right) \mid X = x$ from:	Distribution of $X^{(m)}$
Poisson(λ)	Multinomial$(x, \epsilon_1, \ldots, \epsilon_M)$	Poisson$(\epsilon_m \lambda)$
N(μ, σ^2)	N$_M(\epsilon\mu, \sigma^2 \text{diag}(\epsilon) - \sigma^2 \epsilon\epsilon^T)$.	N$(\epsilon_m \mu, \epsilon_m \sigma^2)$

Our recipe extends naturally to splitting into M>2 folds



Distribution of X	Draw $(X^{(1)}, \ldots, X^{(M)}) \mid X = x$ from:	Distribution of $X^{(m)}$
Poisson(λ)	Multinomial($x, \epsilon_1, \ldots, \epsilon_M$)	Poisson($\epsilon_m \lambda$)
N(μ, σ^2)	$\mathrm{N}_M(\epsilon\mu, \sigma^2 \mathrm{diag}(\epsilon) - \sigma^2 \epsilon\epsilon^T)$.	N($\epsilon_m \mu, \epsilon_m \sigma^2$)
NegativeBinomial(μ, b)	DirichletMultinomial($x, \epsilon_1 b, \ldots, \epsilon_M b$).	NegativeBinomial($\epsilon_m \mu, \epsilon_m b$)

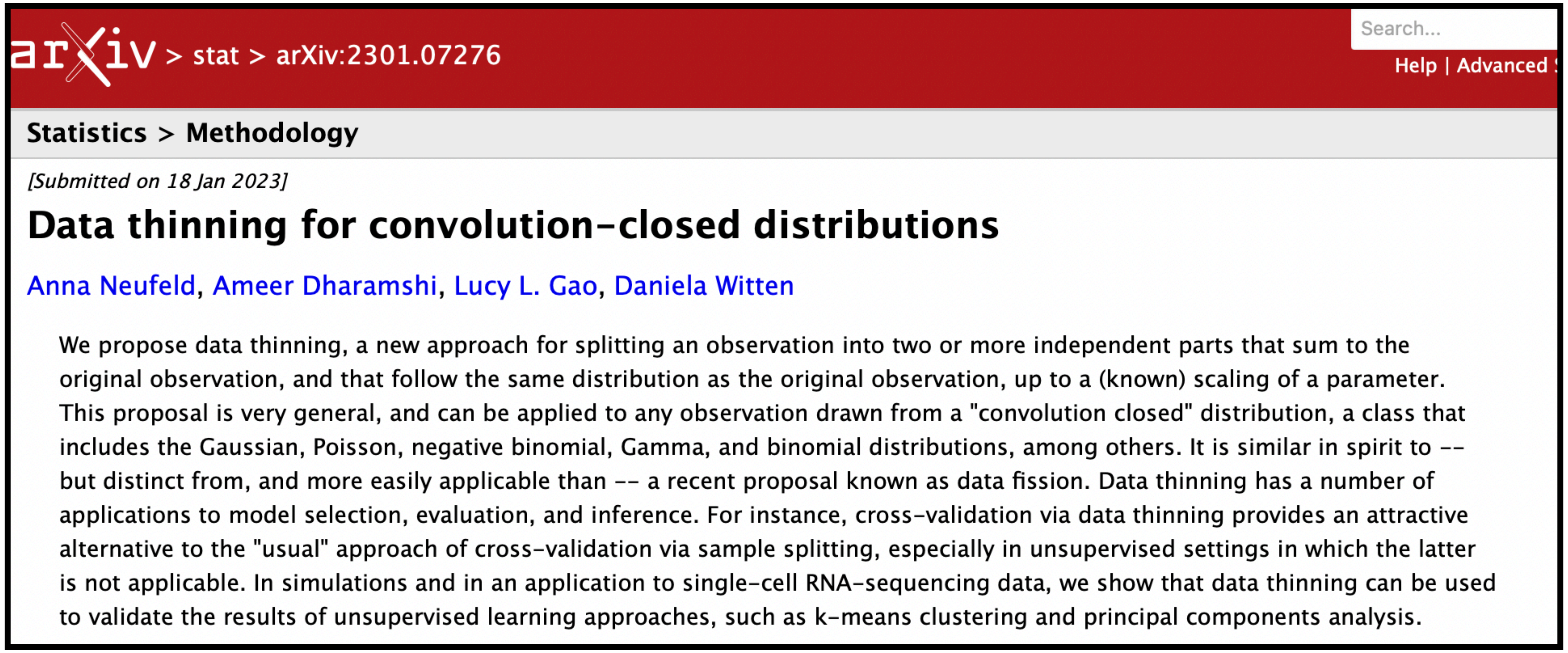
Our recipe extends naturally to splitting into M>2 folds



Distribution of X	Draw $\left(X^{(1)}, \ldots, X^{(M)}\right) \mid X = x$ from:	Distribution of $X^{(m)}$
$\text{Poisson}(\lambda)$	$\text{Multinomial}(x, \epsilon_1, \ldots, \epsilon_M)$	$\text{Poisson}(\epsilon_m \lambda)$
$\text{N}(\mu, \sigma^2)$	$\text{N}_M(\epsilon\mu, \sigma^2 \text{diag}(\epsilon) - \sigma^2 \epsilon\epsilon^T)$.	$\text{N}(\epsilon_m \mu, \epsilon_m \sigma^2)$
$\text{NegativeBinomial}(\mu, b)$	$\text{DirichletMultinomial}(x, \epsilon_1 b, \ldots, \epsilon_M b)$.	$\text{NegativeBinomial}(\epsilon_m \mu, \epsilon_m b)$
$\text{Gamma}(\alpha, \beta)$	$x \cdot \text{Dirichlet}\,(\epsilon_1 \alpha, \ldots, \epsilon_M \alpha)$	$\text{Gamma}(\epsilon_m \alpha, \beta)$
$\text{Exponential}(\lambda)$	$x \cdot \text{Dirichlet}\,(\epsilon_1, \ldots, \epsilon_M)$	$\text{Gamma}(\epsilon_m, \lambda)$
$\text{Binomial}(r, p)$	$\text{MultivariateHypergeometric}(\epsilon_1 r, \ldots, \epsilon_M r, x)$.	$\text{Binomial}(\epsilon_m r, p)$

Data thinning is a simple alternative to sample splitting that can be used in a variety of settings

Project 3



Statistics > Methodology

[Submitted on 18 Jan 2023]

Data thinning for convolution-closed distributions

Anna Neufeld, Ameer Dharamshi, Lucy L. Gao, Daniela Witten

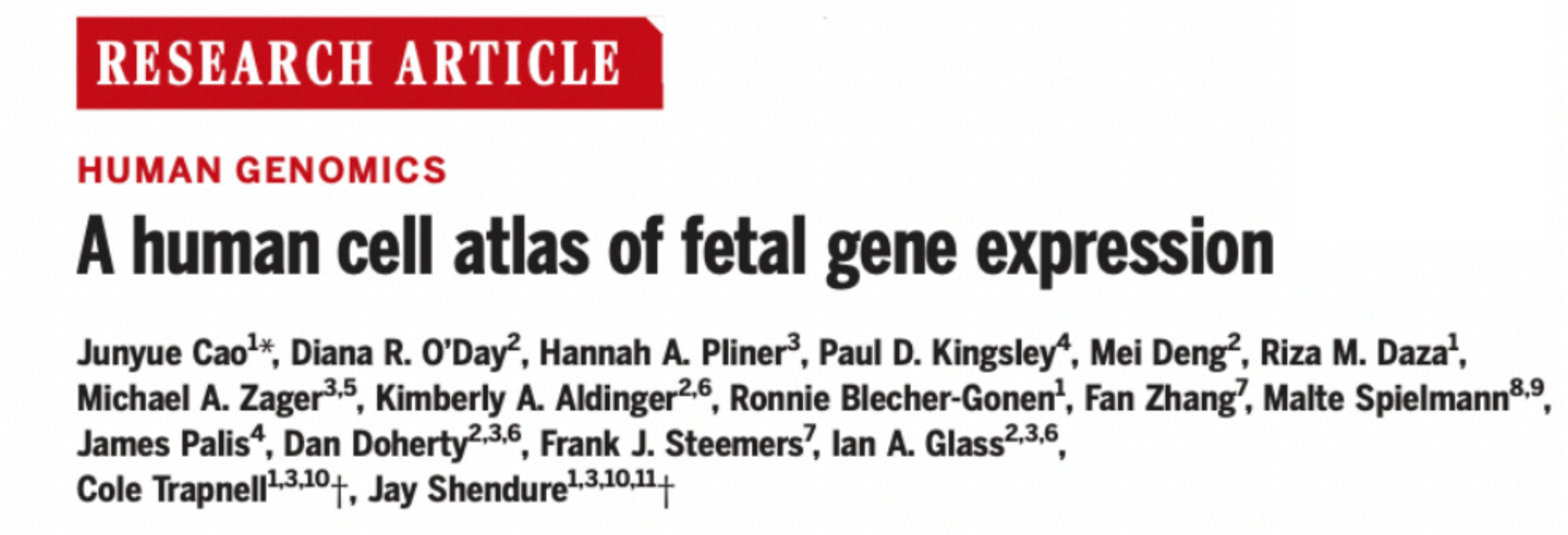
We propose data thinning, a new approach for splitting an observation into two or more independent parts that sum to the original observation, and that follow the same distribution as the original observation, up to a (known) scaling of a parameter. This proposal is very general, and can be applied to any observation drawn from a "convolution closed" distribution, a class that includes the Gaussian, Poisson, negative binomial, Gamma, and binomial distributions, among others. It is similar in spirit to -- but distinct from, and more easily applicable than -- a recent proposal known as data fission. Data thinning has a number of applications to model selection, evaluation, and inference. For instance, cross-validation via data thinning provides an attractive alternative to the "usual" approach of cross-validation via sample splitting, especially in unsupervised settings in which the latter is not applicable. In simulations and in an application to single-cell RNA-sequencing data, we show that data thinning can be used to validate the results of unsupervised learning approaches, such as k-means clustering and principal components analysis.

R package and tutorials: https://anna-neufeld.github.io/datathin/

Outline

1. Motivation: settings where sample splitting doesn’t work
2. Poisson thinning
3. Data thinning
4. **Application to single-cell RNA sequencing data**
5. Ongoing work

How can we validate the results of clustering?



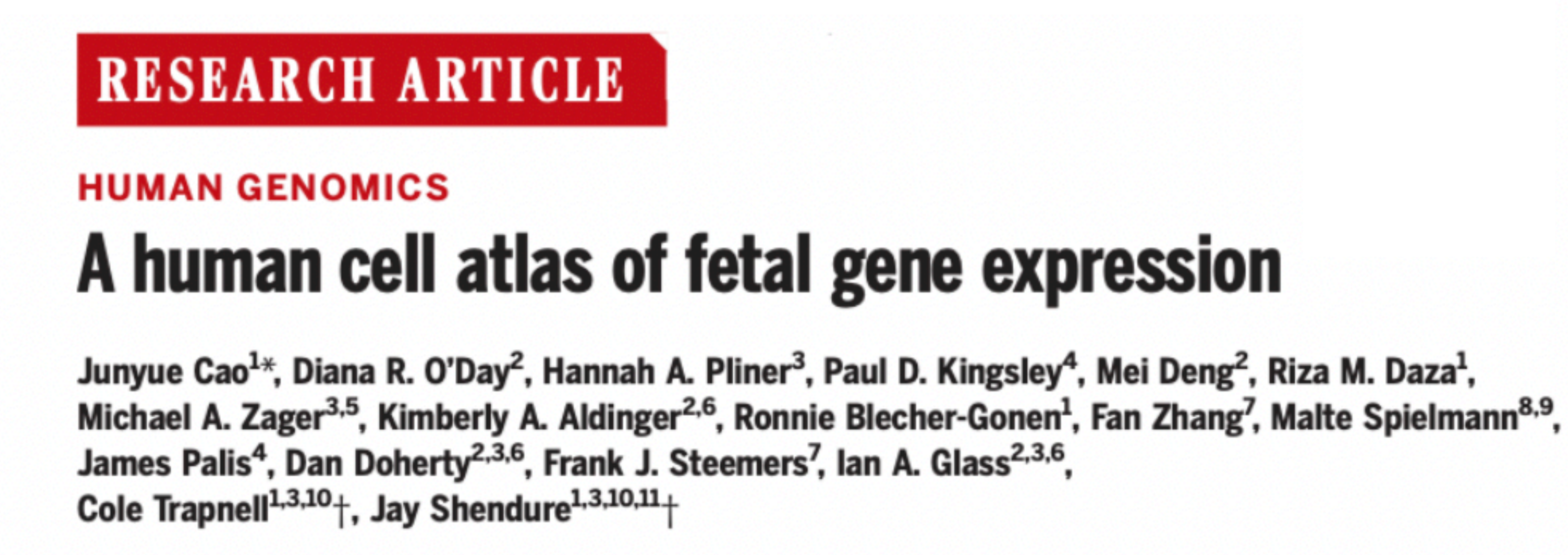
RESEARCH ARTICLE

HUMAN GENOMICS

A human cell atlas of fetal gene expression

Junyue Cao[1]*, Diana R. O'Day[2], Hannah A. Pliner[3], Paul D. Kingsley[4], Mei Deng[2], Riza M. Daza[1], Michael A. Zager[3,5], Kimberly A. Aldinger[2,6], Ronnie Blecher-Gonen[1], Fan Zhang[7], Malte Spielmann[8,9], James Palis[4], Dan Doherty[2,3,6], Frank J. Steemers[7], Ian A. Glass[2,3,6], Cole Trapnell[1,3,10]†, Jay Shendure[1,3,10,11]†

How can we validate the results of clustering?

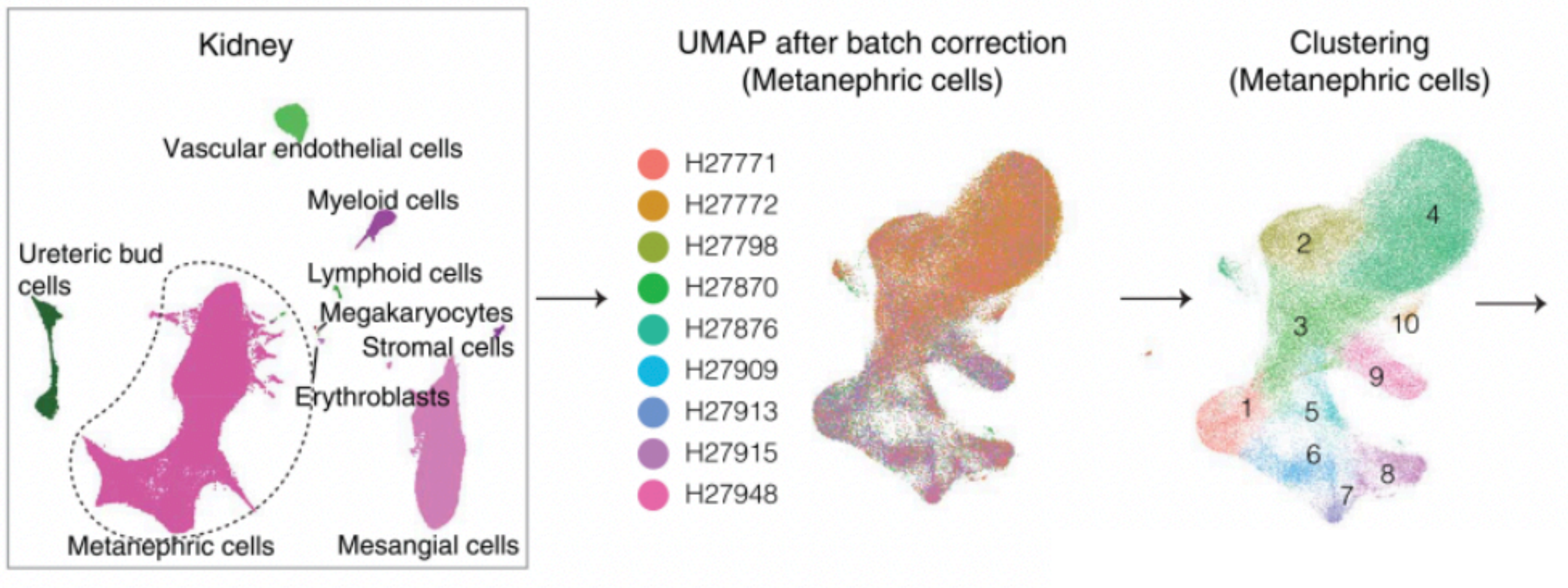


RESEARCH ARTICLE

HUMAN GENOMICS

A human cell atlas of fetal gene expression

Junyue Cao[1]*, Diana R. O'Day[2], Hannah A. Pliner[3], Paul D. Kingsley[4], Mei Deng[2], Riza M. Daza[1], Michael A. Zager[3,5], Kimberly A. Aldinger[2,6], Ronnie Blecher-Gonen[1], Fan Zhang[7], Malte Spielmann[8,9], James Palis[4], Dan Doherty[2,3,6], Frank J. Steemers[7], Ian A. Glass[2,3,6], Cole Trapnell[1,3,10]†, Jay Shendure[1,3,10,11]†



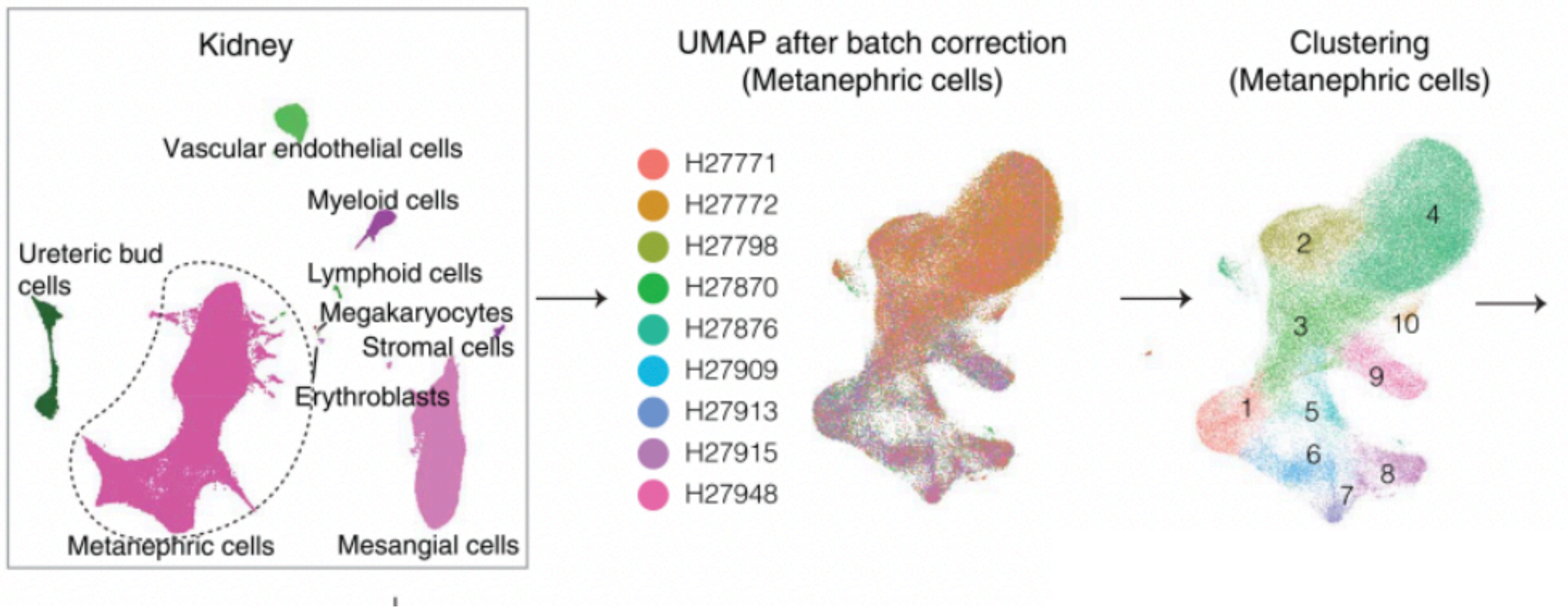
How can we validate the results of clustering?

RESEARCH ARTICLE

HUMAN GENOMICS

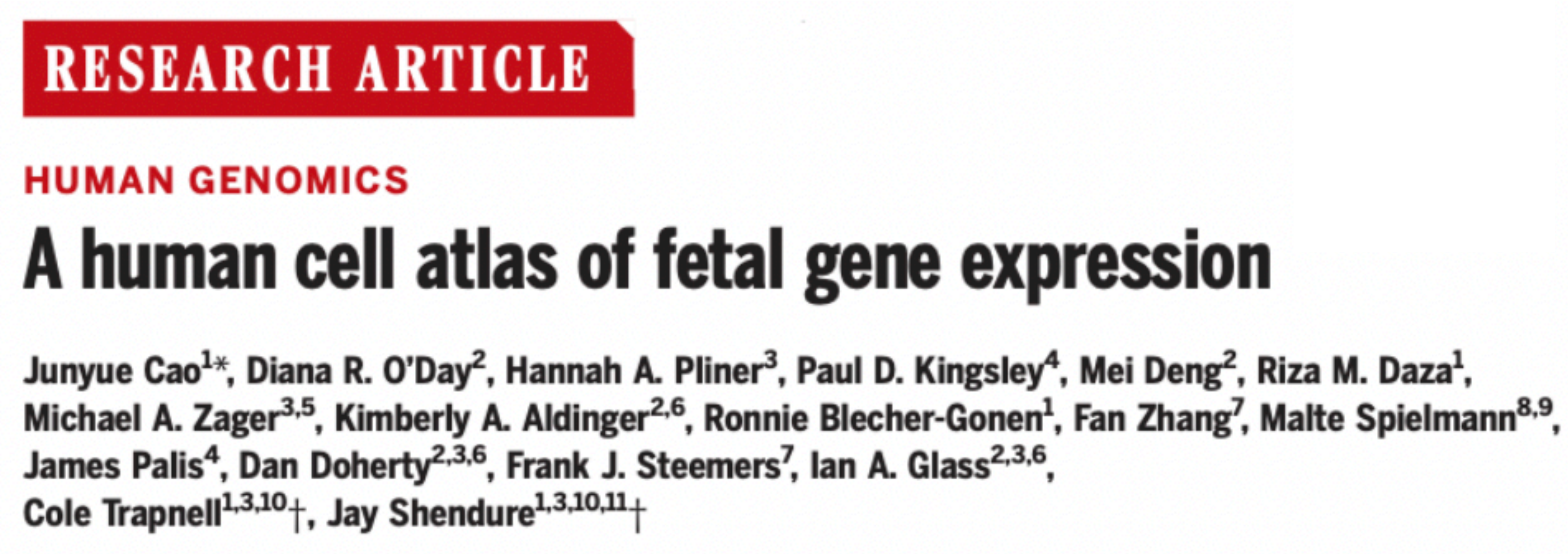
A human cell atlas of fetal gene expression

Junyue Cao[1]*, Diana R. O'Day[2], Hannah A. Pliner[3], Paul D. Kingsley[4], Mei Deng[2], Riza M. Daza[1], Michael A. Zager[3,5], Kimberly A. Aldinger[2,6], Ronnie Blecher-Gonen[1], Fan Zhang[7], Malte Spielmann[8,9], James Palis[4], Dan Doherty[2,3,6], Frank J. Steemers[7], Ian A. Glass[2,3,6], Cole Trapnell[1,3,10]†, Jay Shendure[1,3,10,11]†



- **Step 1:** Cluster cells.

How can we validate the results of clustering?

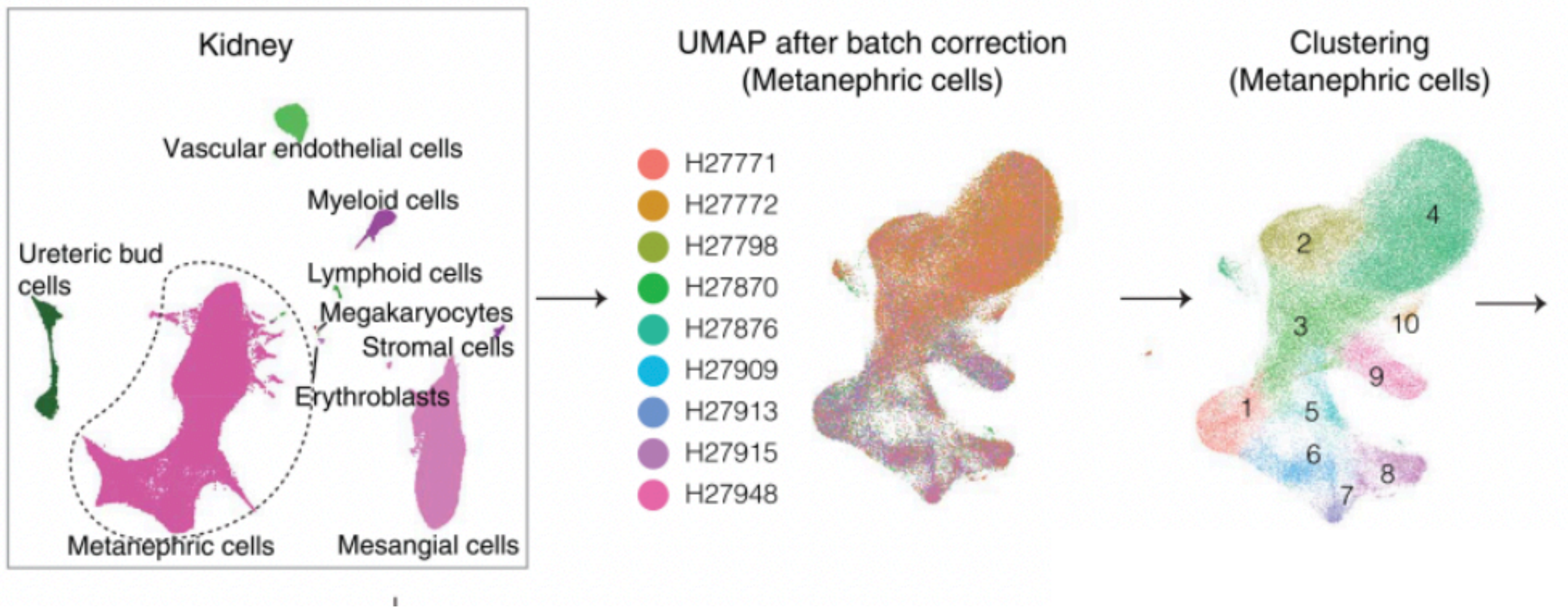


RESEARCH ARTICLE

HUMAN GENOMICS

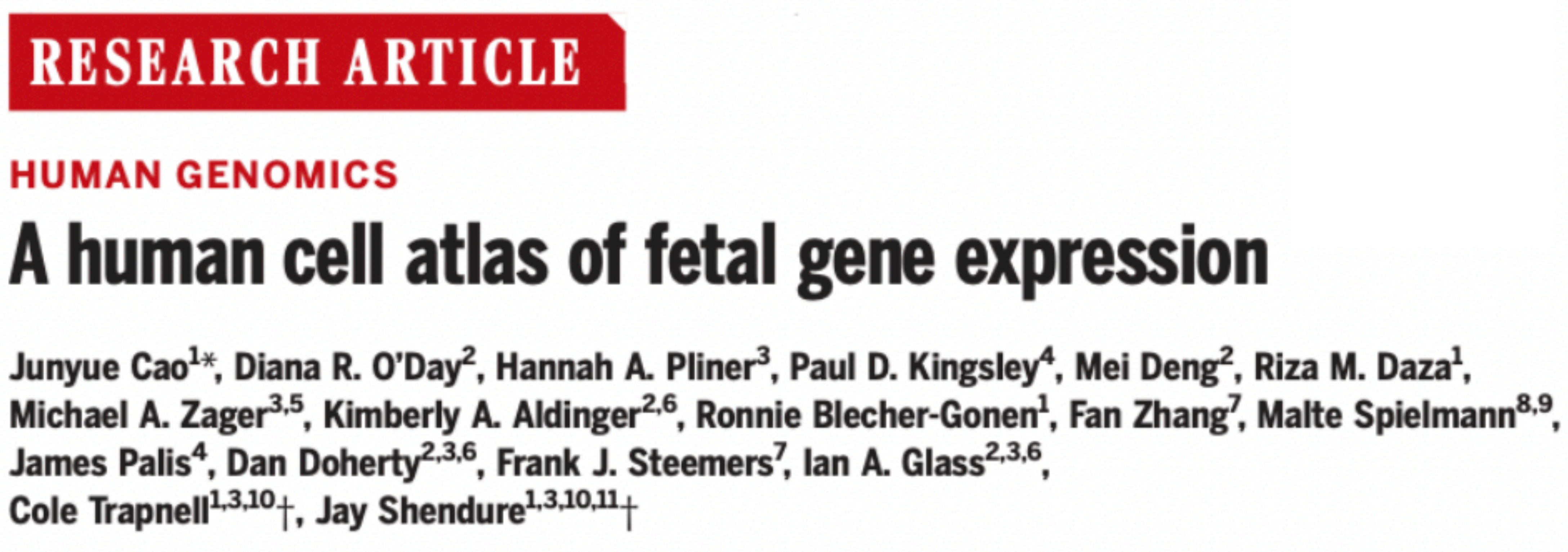
A human cell atlas of fetal gene expression

Junyue Cao[1]*, Diana R. O'Day[2], Hannah A. Pliner[3], Paul D. Kingsley[4], Mei Deng[2], Riza M. Daza[1], Michael A. Zager[3,5], Kimberly A. Aldinger[2,6], Ronnie Blecher-Gonen[1], Fan Zhang[7], Malte Spielmann[8,9], James Palis[4], Dan Doherty[2,3,6], Frank J. Steemers[7], Ian A. Glass[2,3,6], Cole Trapnell[1,3,10]†, Jay Shendure[1,3,10,11]†



- **Step 1:** Cluster cells.

How can we validate the results of clustering?

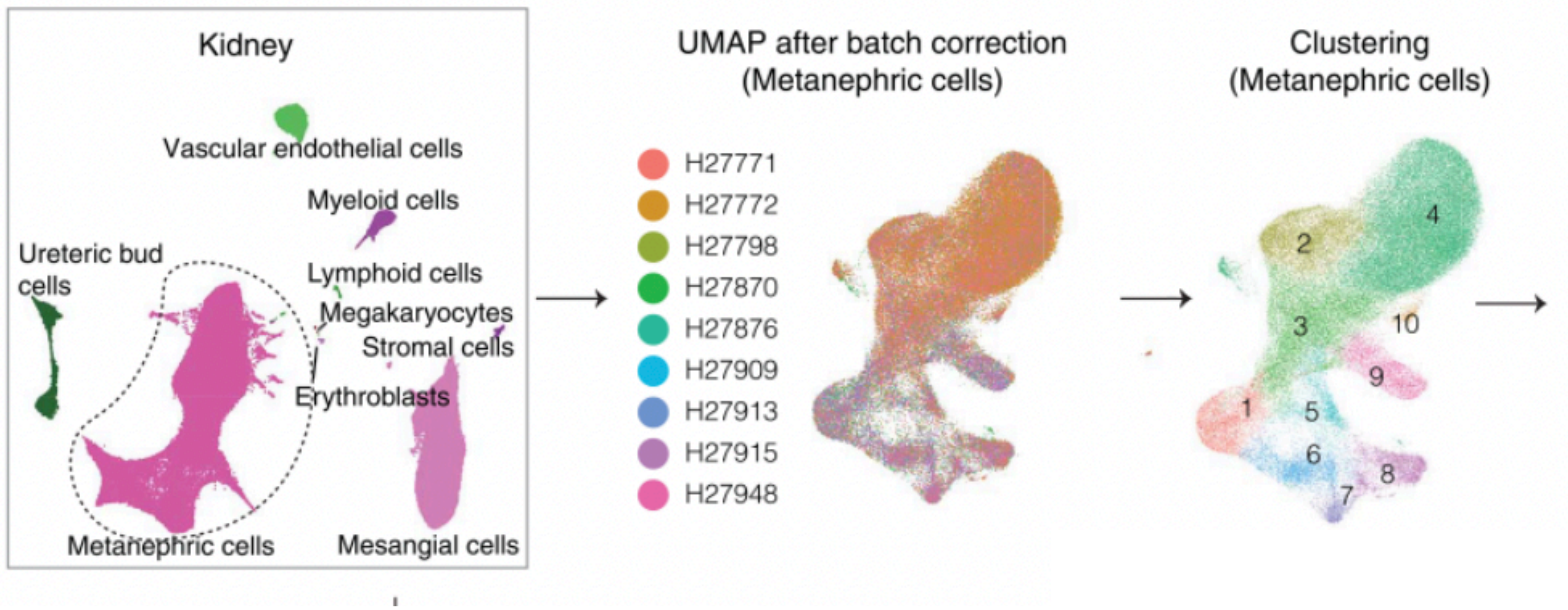


RESEARCH ARTICLE

HUMAN GENOMICS

A human cell atlas of fetal gene expression

Junyue Cao[1]*, Diana R. O'Day[2], Hannah A. Pliner[3], Paul D. Kingsley[4], Mei Deng[2], Riza M. Daza[1], Michael A. Zager[3,5], Kimberly A. Aldinger[2,6], Ronnie Blecher-Gonen[1], Fan Zhang[7], Malte Spielmann[8,9], James Palis[4], Dan Doherty[2,3,6], Frank J. Steemers[7], Ian A. Glass[2,3,6], Cole Trapnell[1,3,10]†, Jay Shendure[1,3,10,11]†



- **Step 1:** Cluster cells.
- **Step 2:** Treat clusters as truth. Do 5-fold cross validation with SVM.

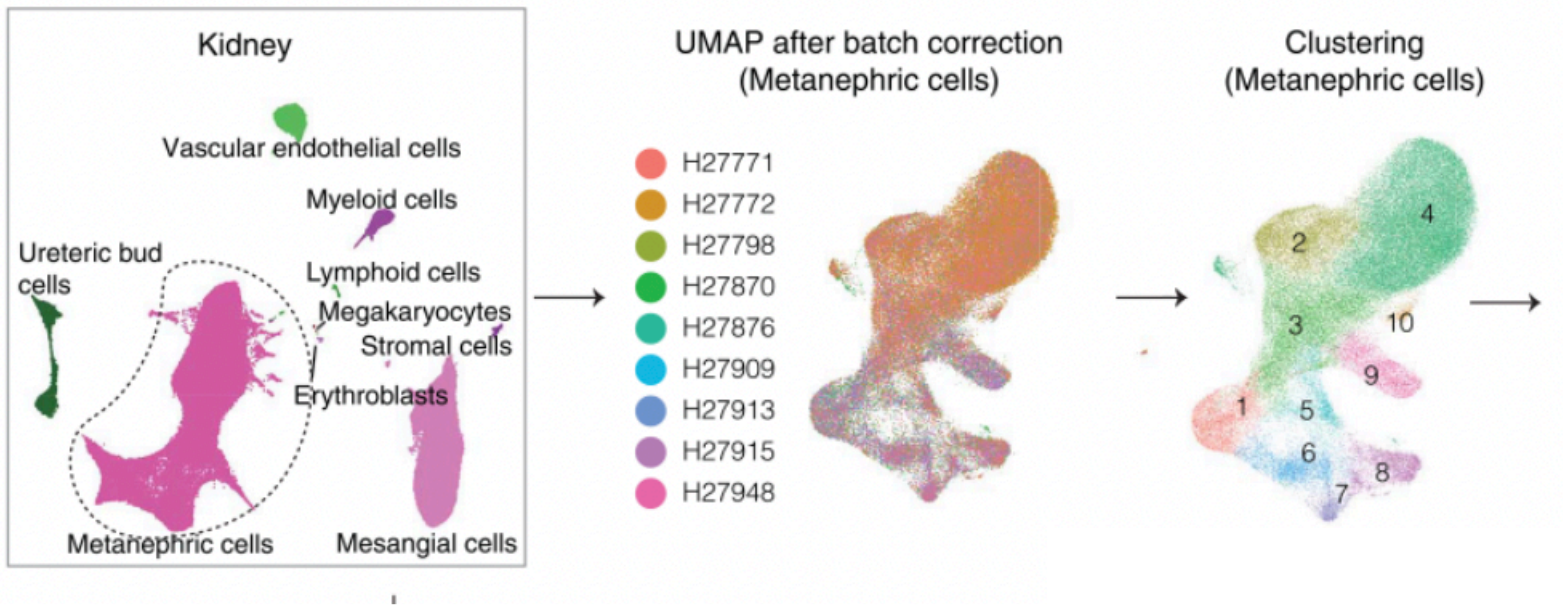
How can we validate the results of clustering?

RESEARCH ARTICLE

HUMAN GENOMICS

A human cell atlas of fetal gene expression

Junyue Cao[1]*, Diana R. O'Day[2], Hannah A. Pliner[3], Paul D. Kingsley[4], Mei Deng[2], Riza M. Daza[1], Michael A. Zager[3,5], Kimberly A. Aldinger[2,6], Ronnie Blecher-Gonen[1], Fan Zhang[7], Malte Spielmann[8,9], James Palis[4], Dan Doherty[2,3,6], Frank J. Steemers[7], Ian A. Glass[2,3,6], Cole Trapnell[1,3,10]†, Jay Shendure[1,3,10,11]†



- **Step 1:** Cluster cells.

- **Step 2:** Treat clusters as truth. Do 5-fold cross validation with SVM.

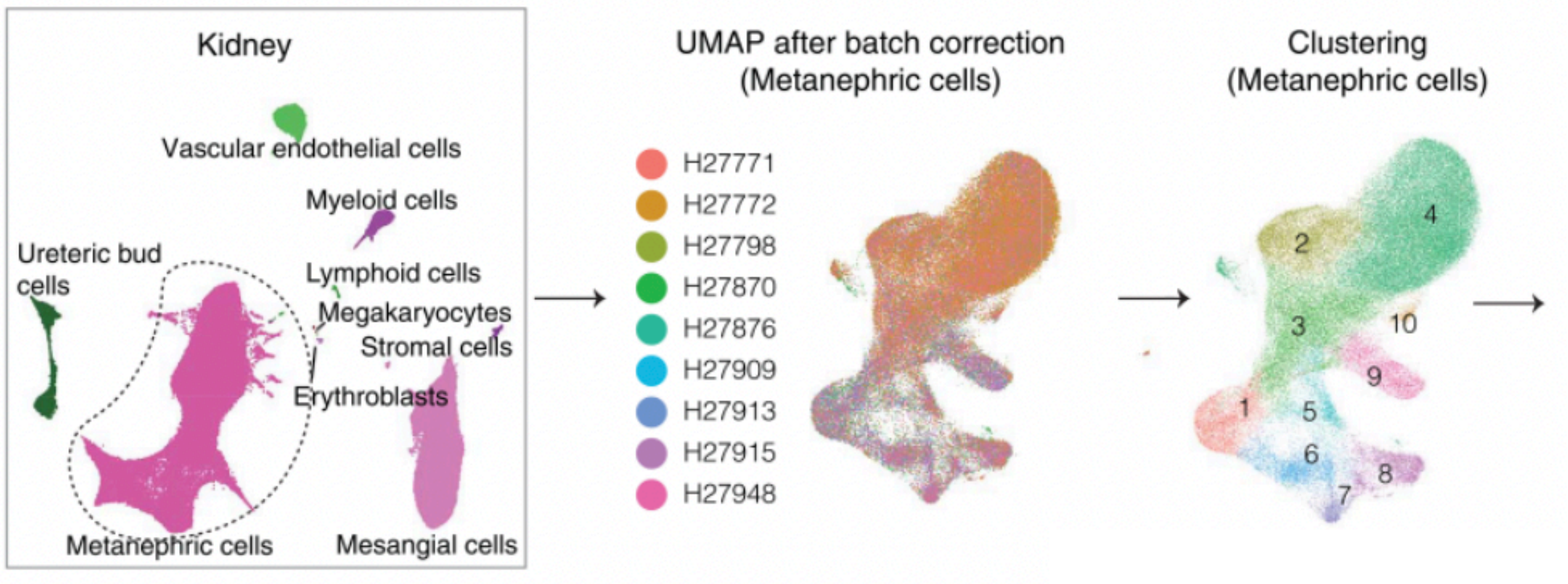
How can we validate the results of clustering?

RESEARCH ARTICLE

HUMAN GENOMICS

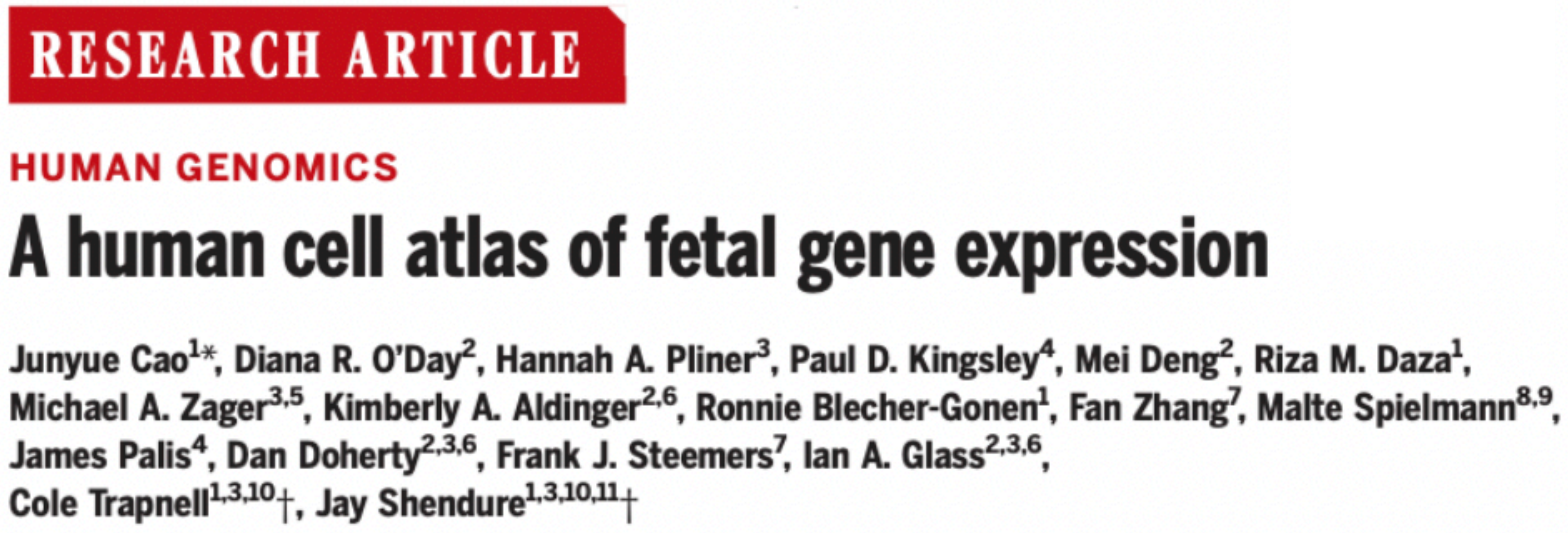
A human cell atlas of fetal gene expression

Junyue Cao[1*], Diana R. O'Day[2], Hannah A. Pliner[3], Paul D. Kingsley[4], Mei Deng[2], Riza M. Daza[1], Michael A. Zager[3,5], Kimberly A. Aldinger[2,6], Ronnie Blecher-Gonen[1], Fan Zhang[7], Malte Spielmann[8,9], James Palis[4], Dan Doherty[2,3,6], Frank J. Steemers[7], Ian A. Glass[2,3,6], Cole Trapnell[1,3,10†], Jay Shendure[1,3,10,11†]



- **Step 1:** Cluster cells.
- **Step 2:** Treat clusters as truth. Do 5-fold cross validation with SVM.
- **Step 3:** Compare clusters to SVM predictions.

How can we validate the results of clustering?



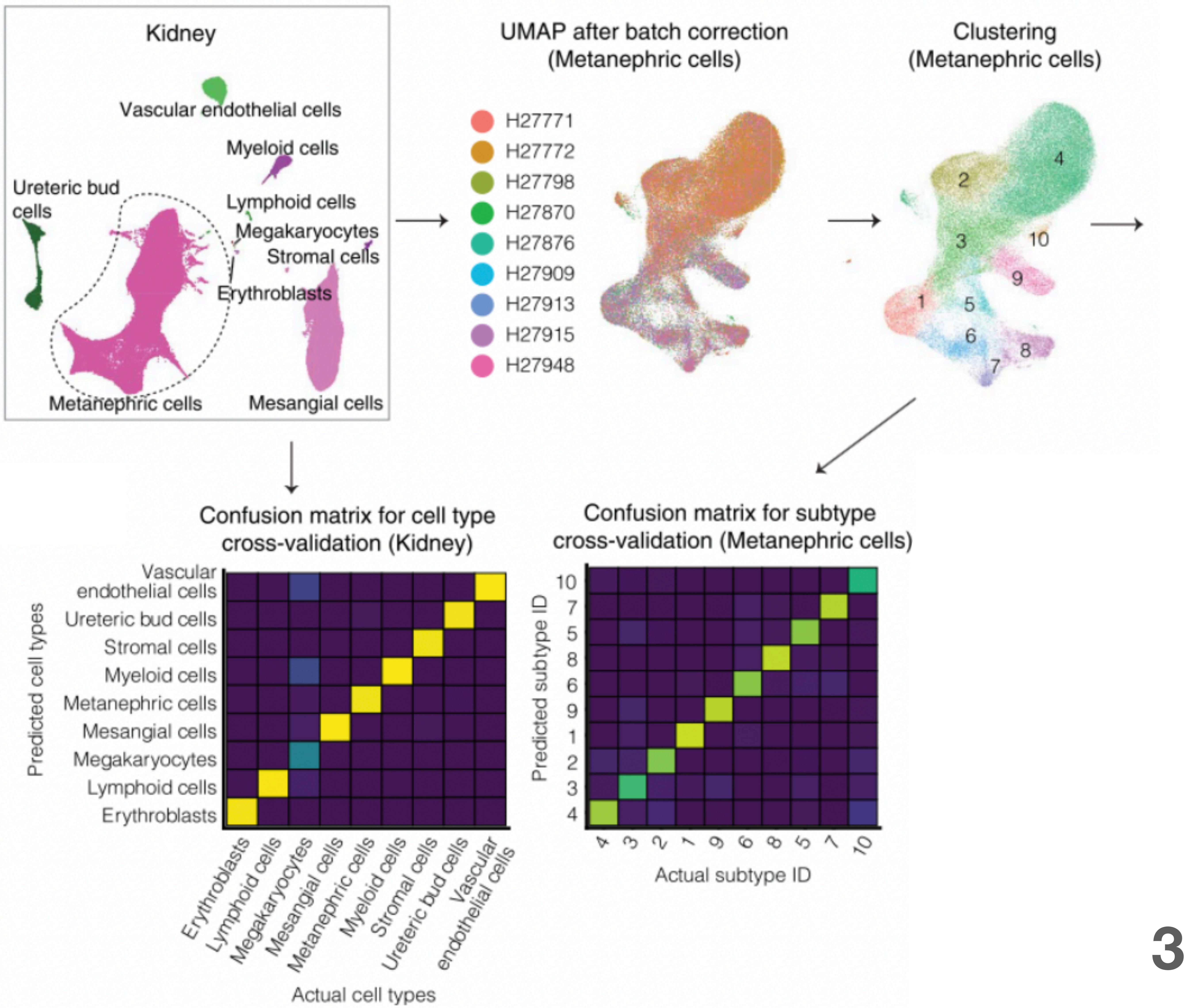
RESEARCH ARTICLE

HUMAN GENOMICS

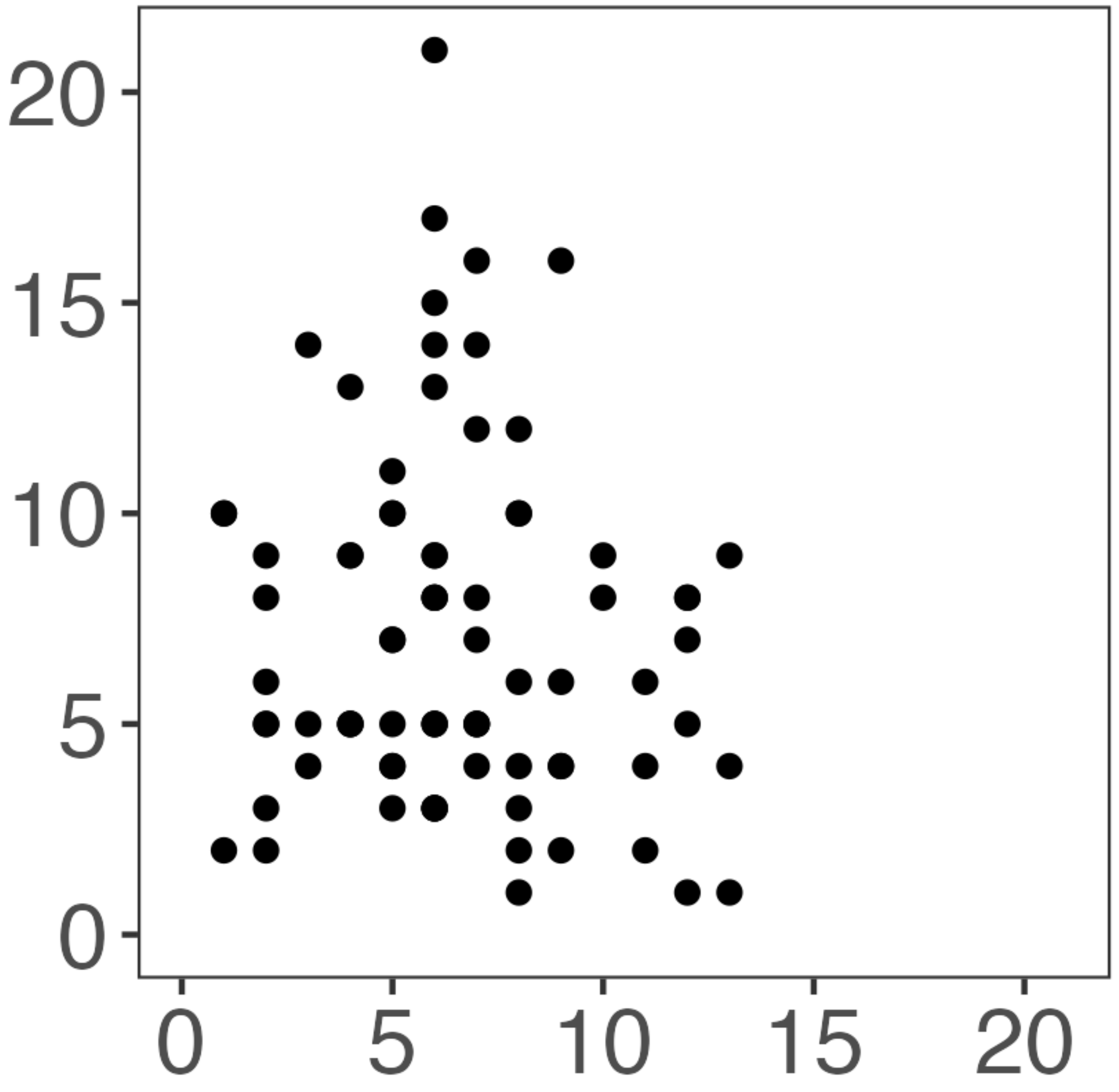
A human cell atlas of fetal gene expression

Junyue Cao[1]*, Diana R. O'Day[2], Hannah A. Pliner[3], Paul D. Kingsley[4], Mei Deng[2], Riza M. Daza[1], Michael A. Zager[3,5], Kimberly A. Aldinger[2,6], Ronnie Blecher-Gonen[1], Fan Zhang[7], Malte Spielmann[8,9], James Palis[4], Dan Doherty[2,3,6], Frank J. Steemers[7], Ian A. Glass[2,3,6], Cole Trapnell[1,3,10]†, Jay Shendure[1,3,10,11]†

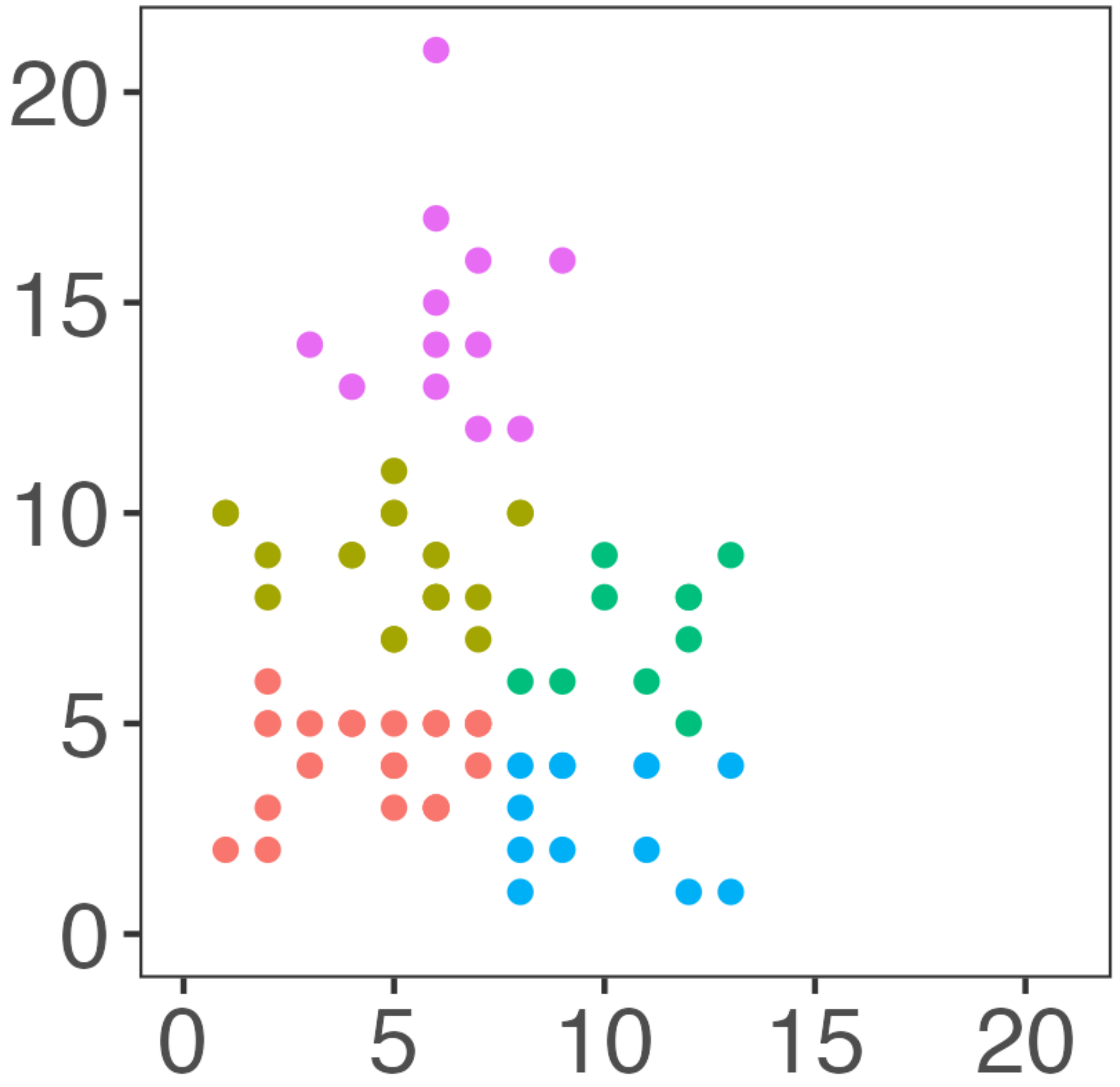
- **Step 1:** Cluster cells.
- **Step 2:** Treat clusters as truth. Do 5-fold cross validation with SVM.
- **Step 3:** Compare clusters to SVM predictions.



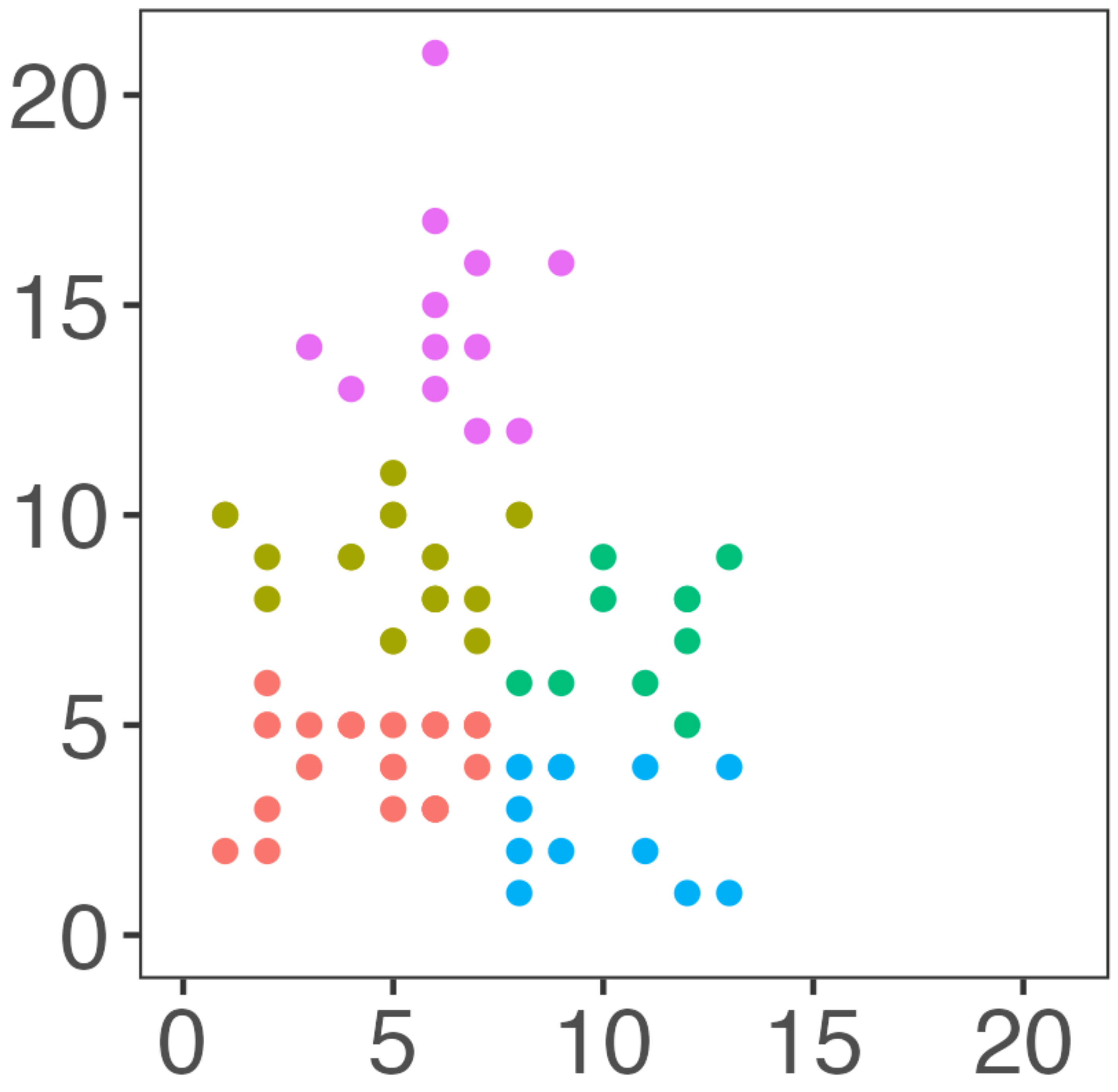
This cross validation procedure double dips

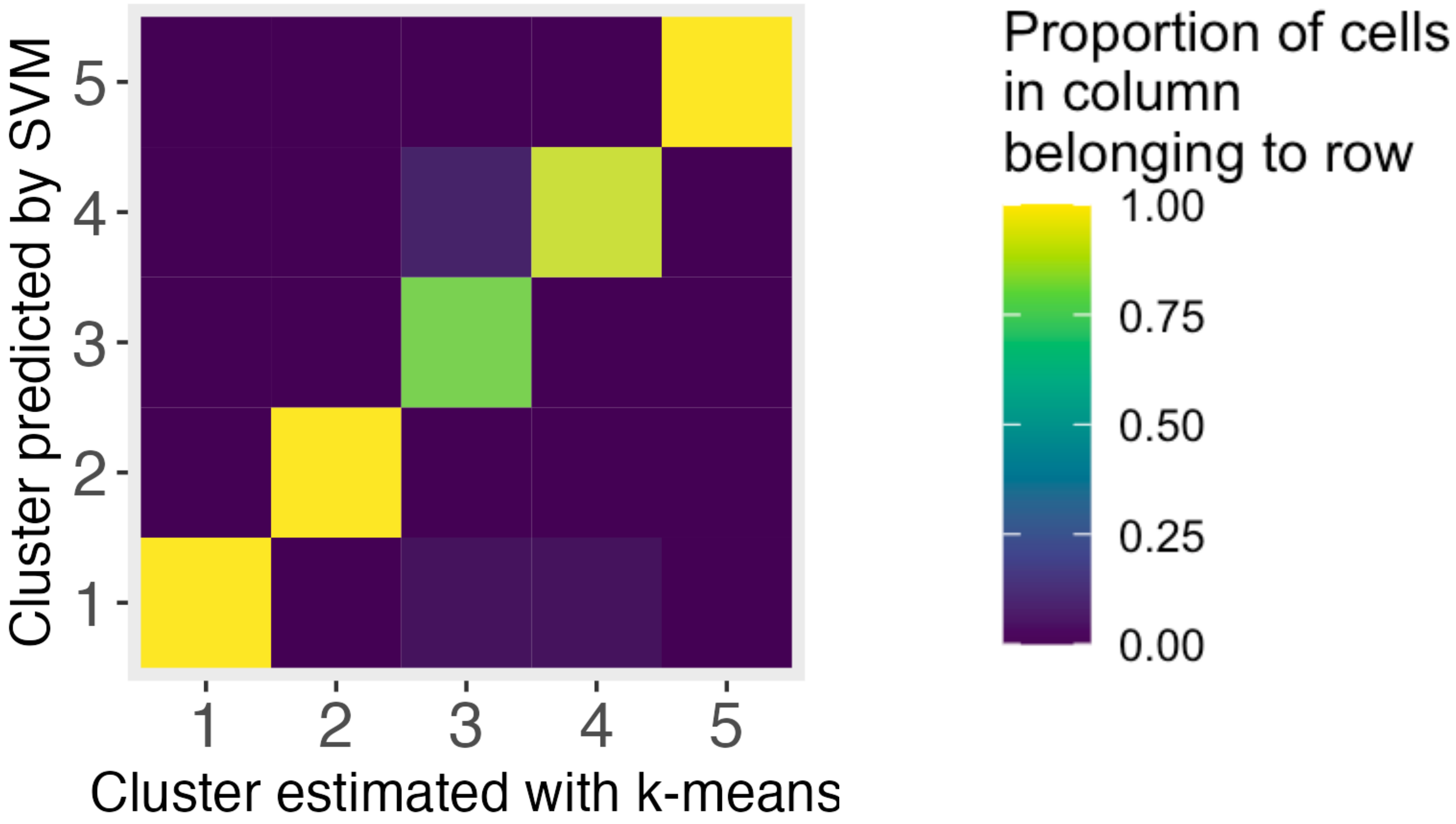


This cross validation procedure double dips



This cross validation procedure double dips

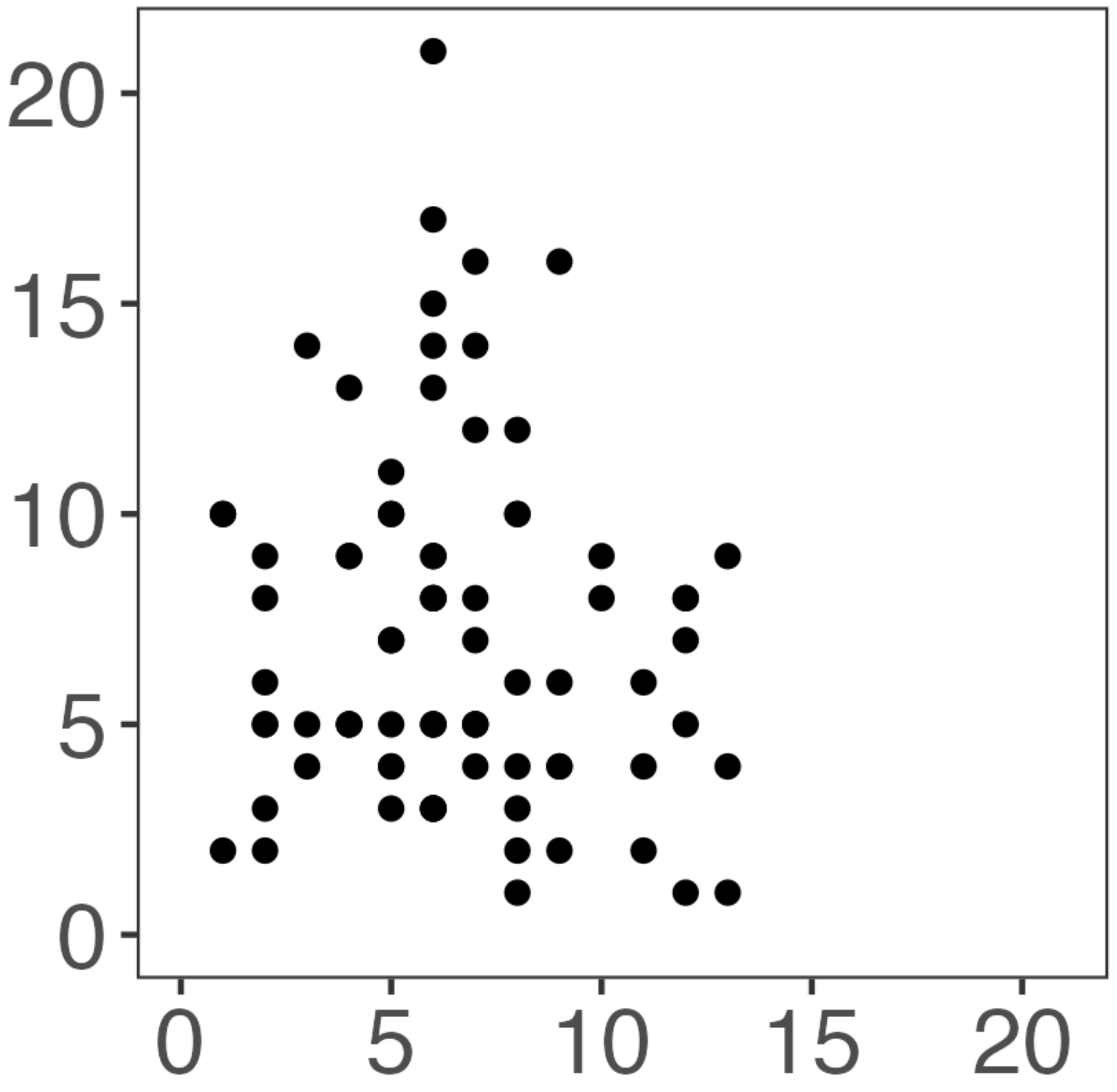




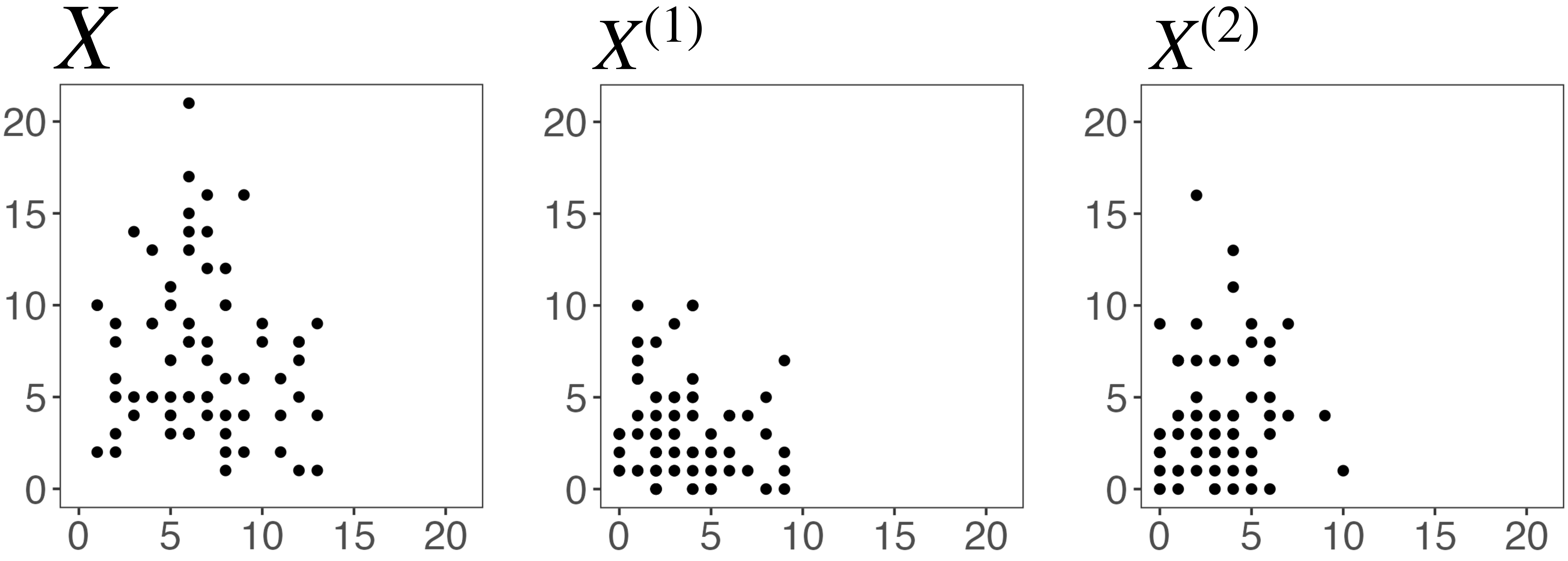
SVM gets 96% accuracy on test set, despite the fact that clusters are not “real”.

Data thinning provides a simple alternative

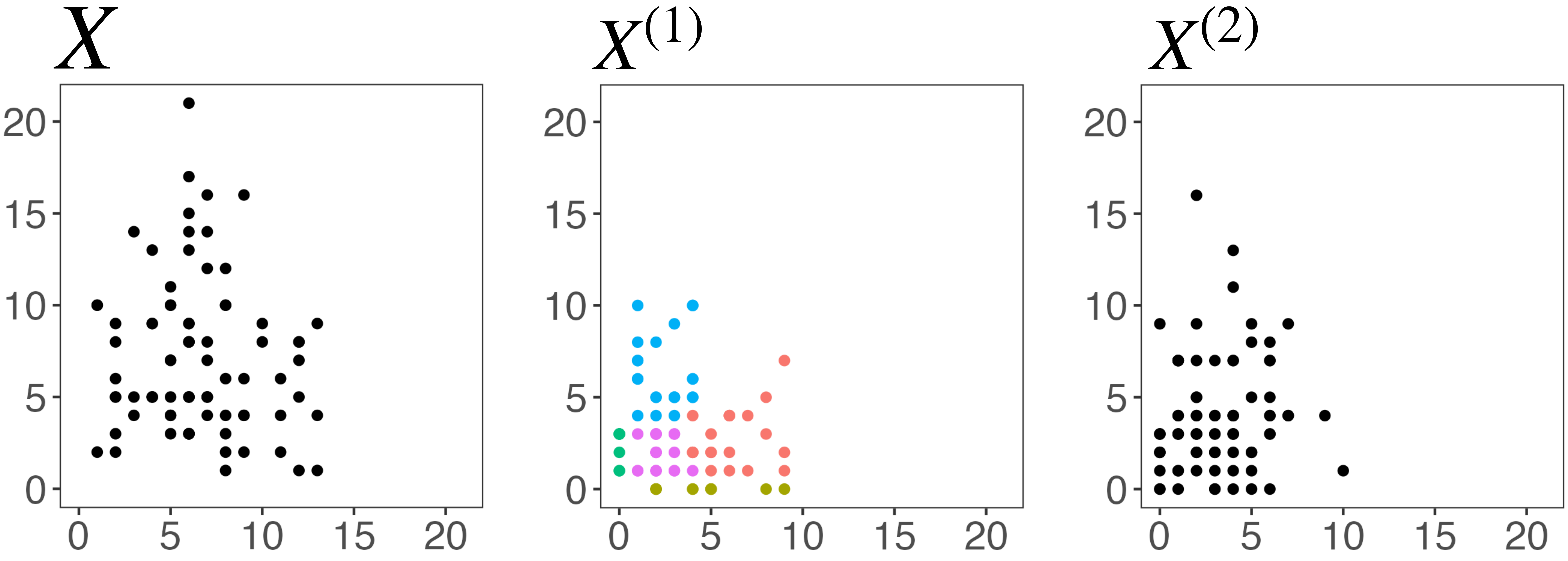
X



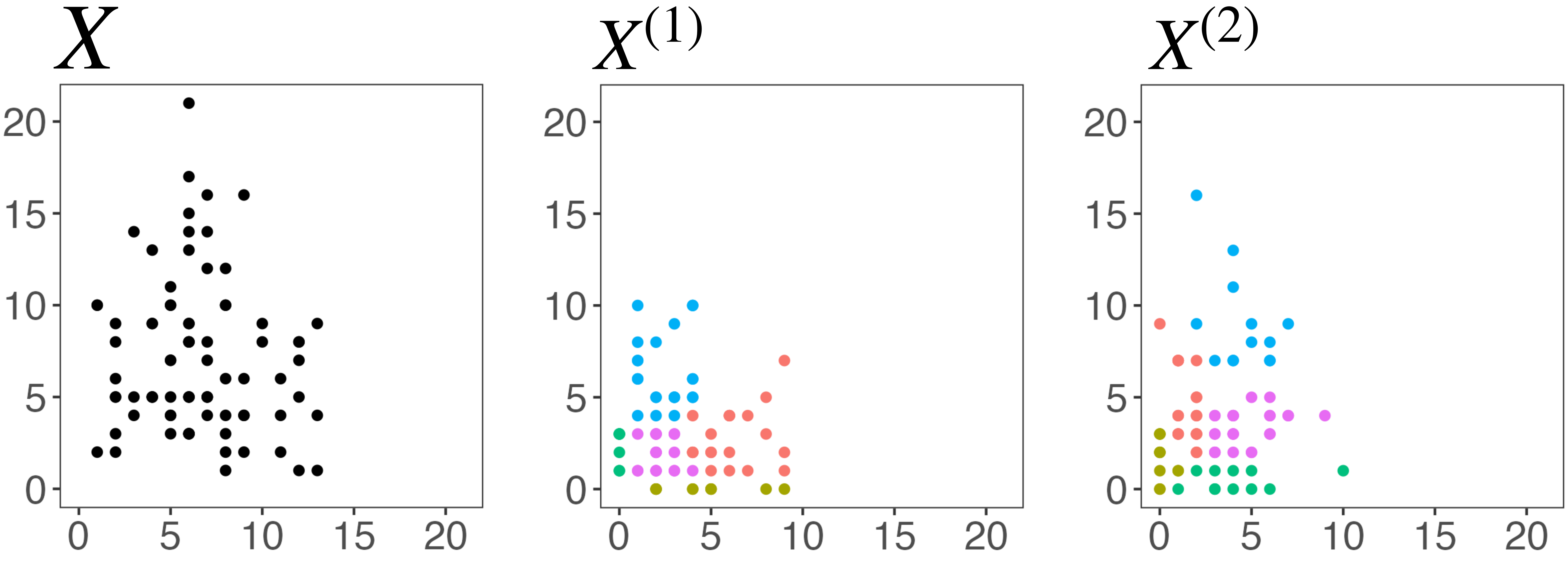
Data thinning provides a simple alternative


X $X^{(1)}$ $X^{(2)}$

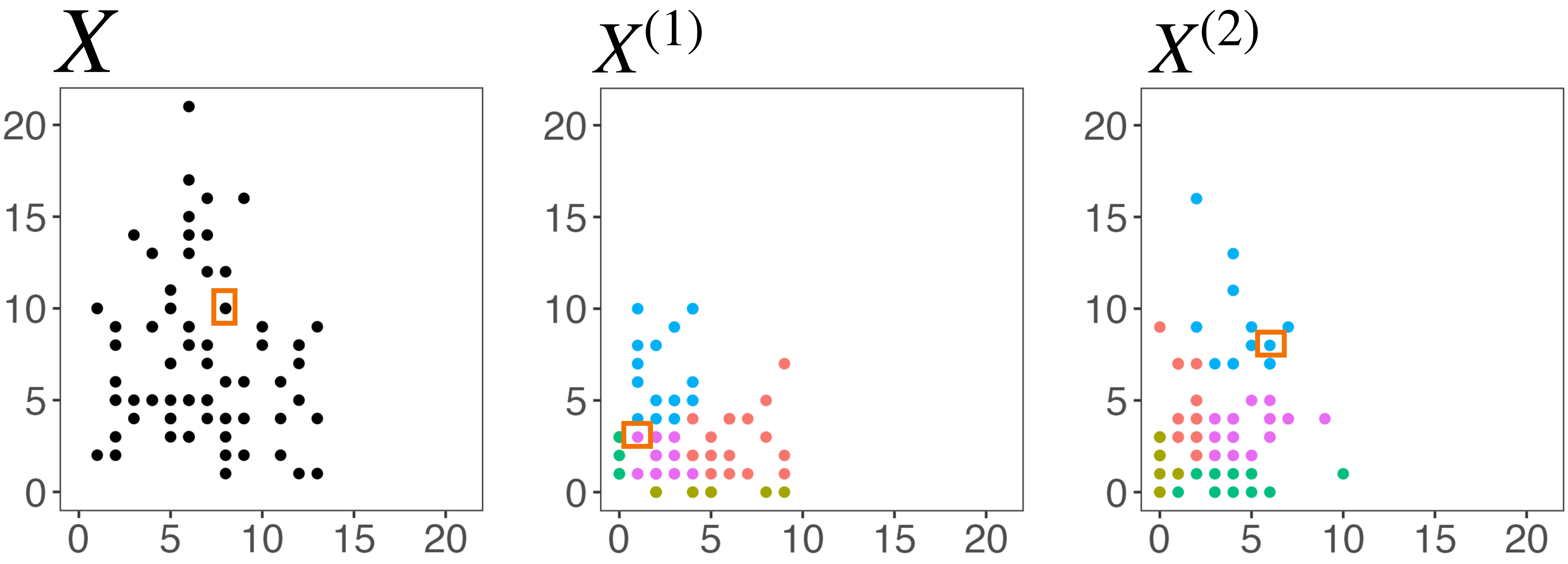
Data thinning provides a simple alternative



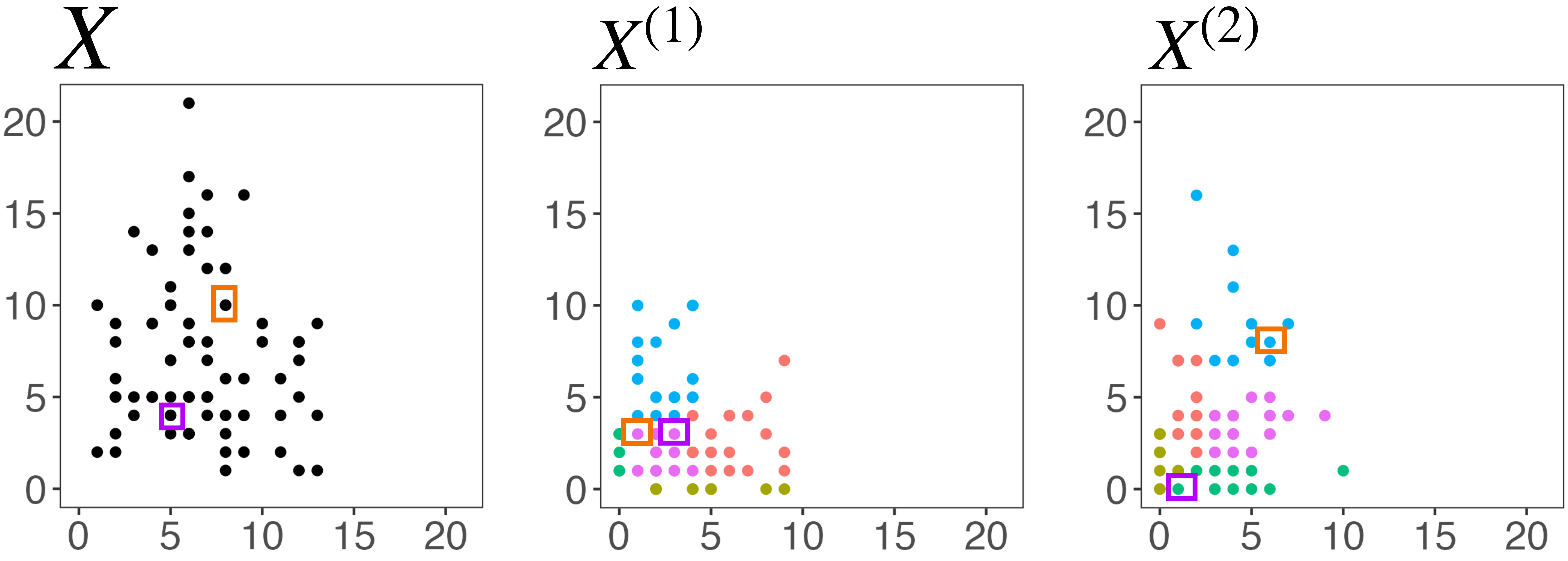
Data thinning provides a simple alternative



Data thinning provides a simple alternative



Data thinning provides a simple alternative



Data thinning provides a simple alternative

Cluster estimated on X$^{(2)}$
5
4
3
2
1
1
2
3
4
5
Cluster estimated on X$^{(1)}$
$X^{(1)}$
20
15
10
5
0
0
5
10
15
20
$X^{(2)}$
20
15
10
5
0
0
5
10
15
20
Proportion of cells in column belonging to row
1.00
0.75
0.50
0.25
0.00

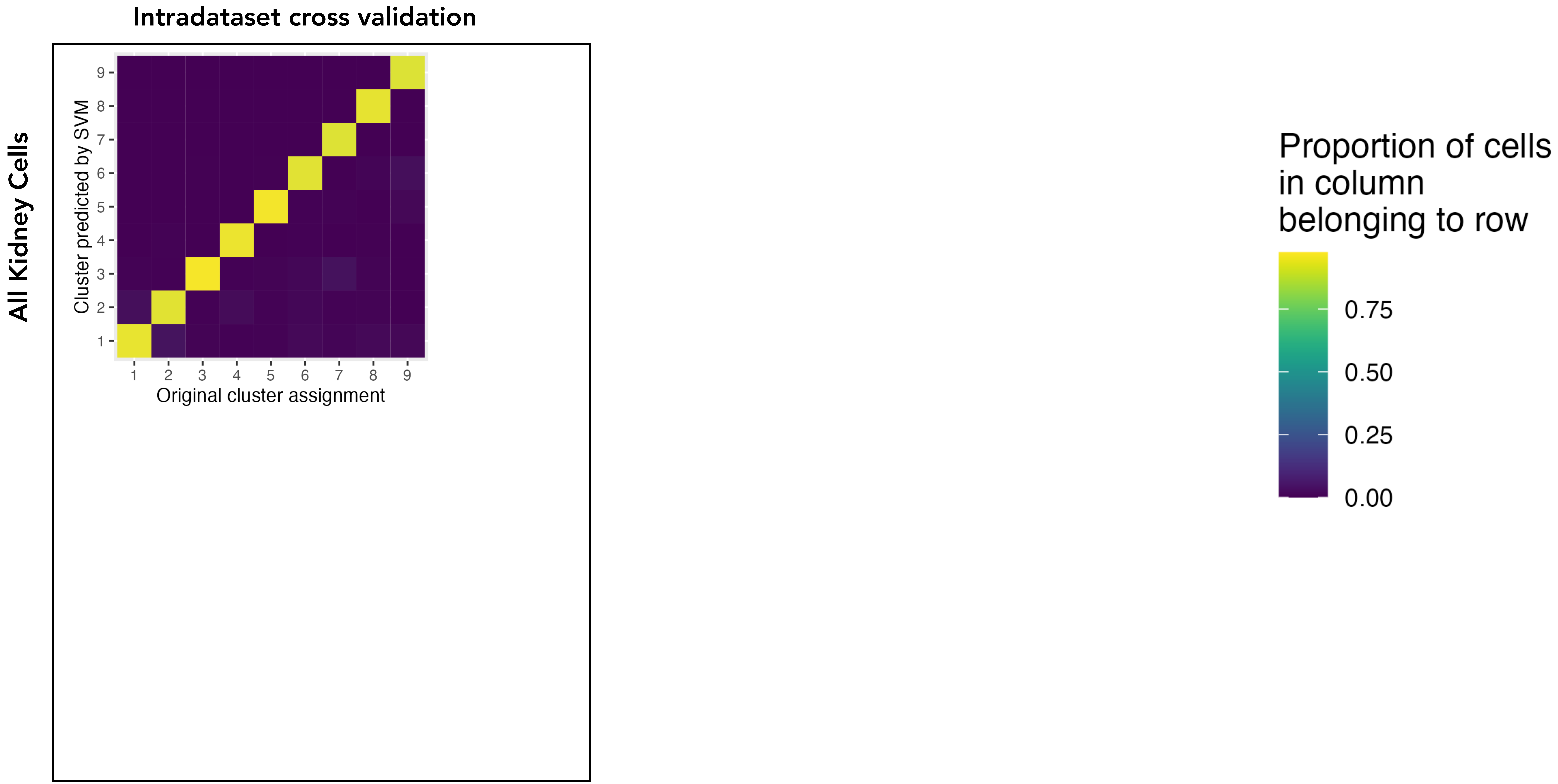
Adjusted Rand Index ≈ 0.01

Re-analysis of Kidney cell data from fetal cell atlas

Re-analysis of Kidney cell data from fetal cell atlas

Intradataset cross validation

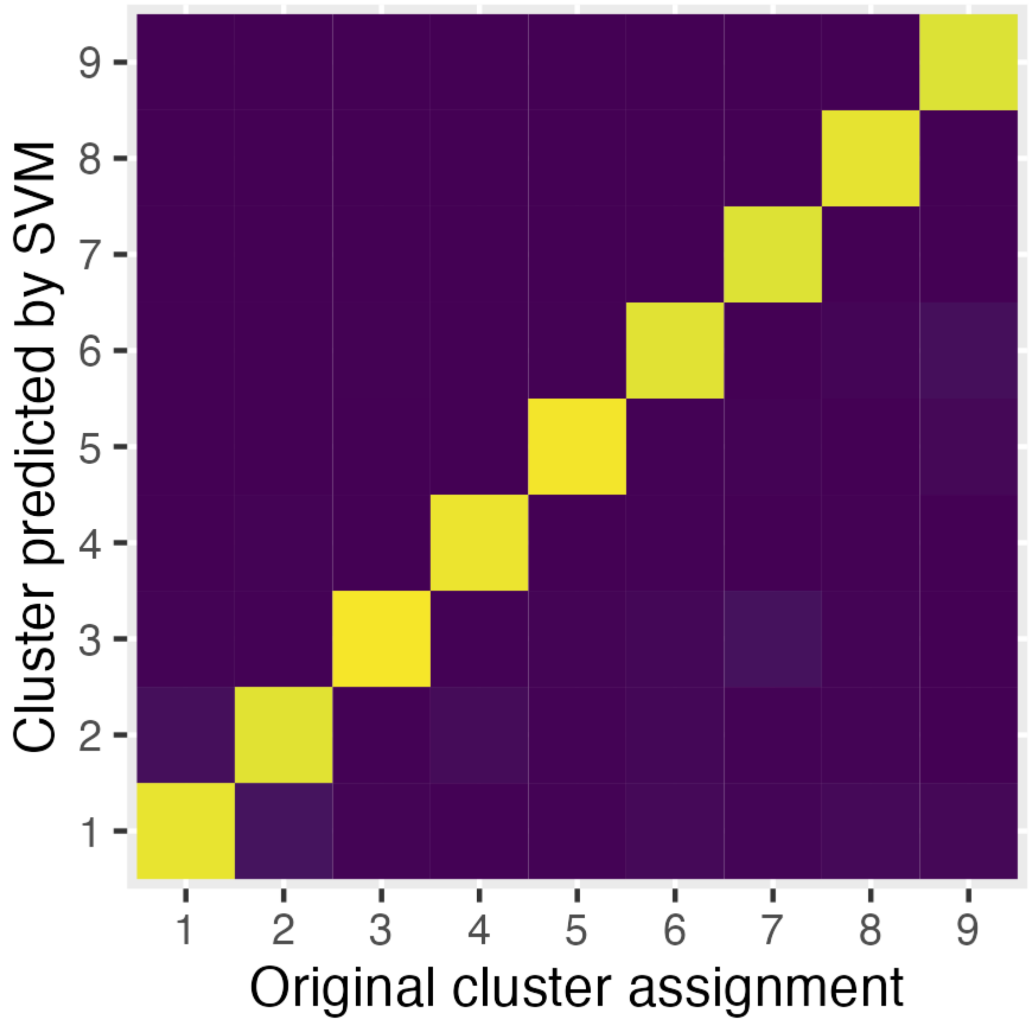
Re-analysis of Kidney cell data from fetal cell atlas



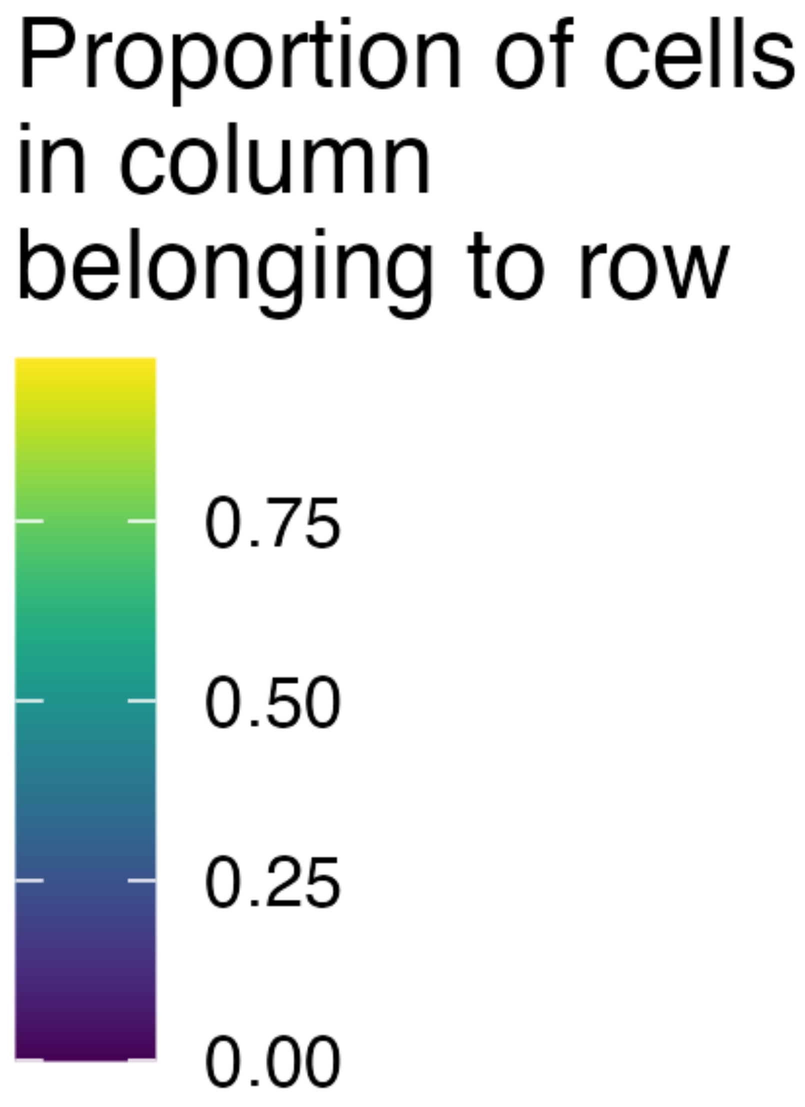
Re-analysis of Kidney cell data from fetal cell atlas

Intradataset cross validation

All Kidney Cells

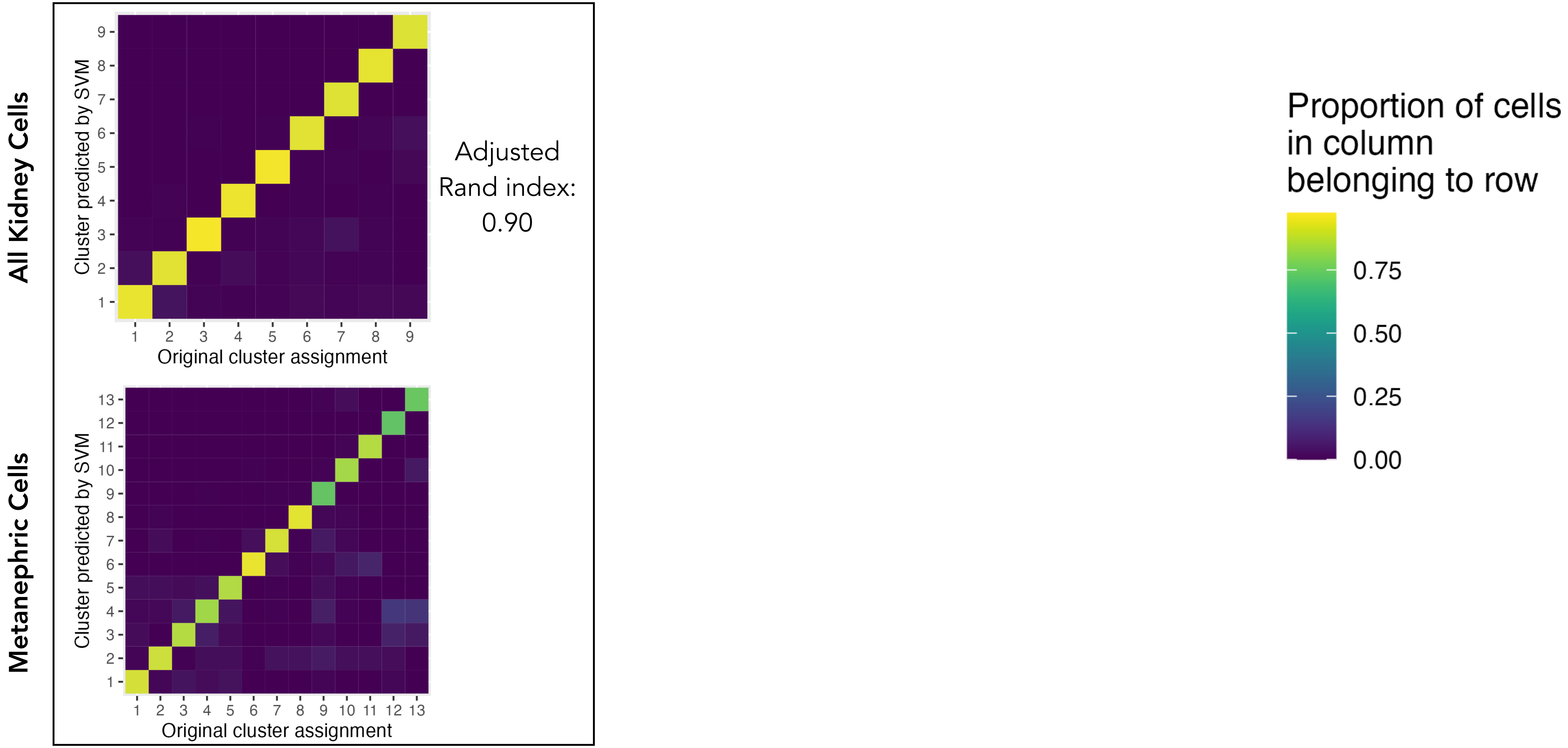


Adjusted
Rand index:
0.90



Re-analysis of Kidney cell data from fetal cell atlas

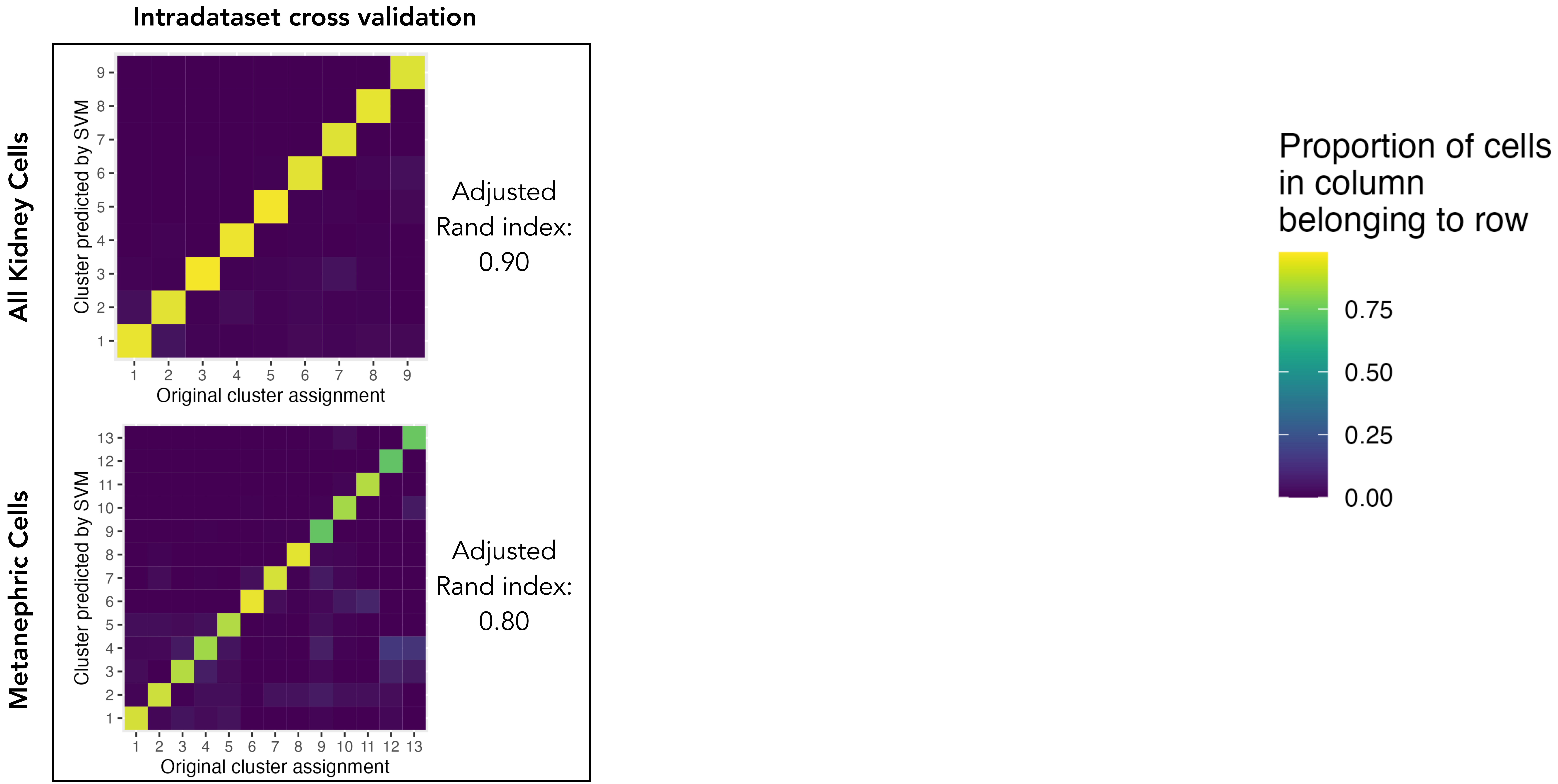
Intradataset cross validation



Adjusted Rand index: 0.90

Re-analysis of Kidney cell data from fetal cell atlas

Intradataset cross validation



Adjusted Rand index: 0.90

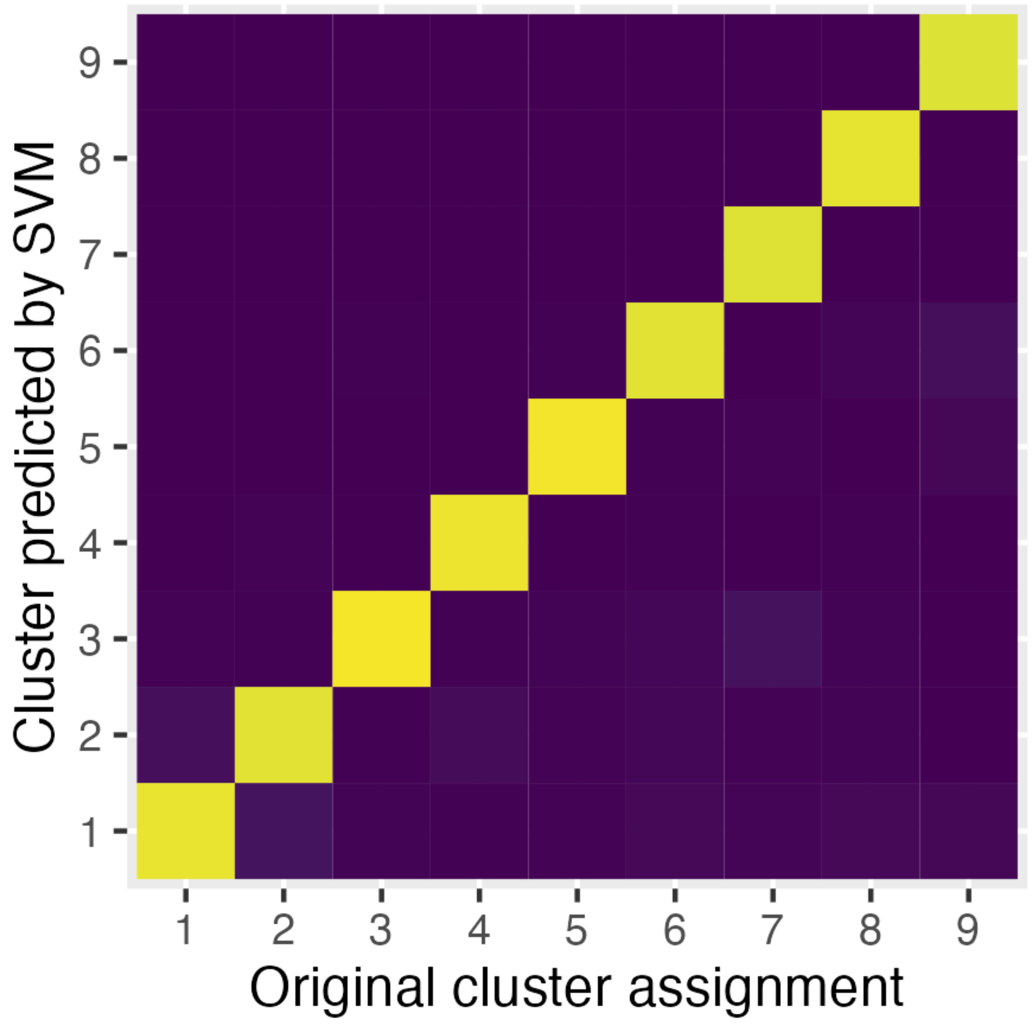
Adjusted Rand index: 0.80

Re-analysis of Kidney cell data from fetal cell atlas

Intradataset cross validation

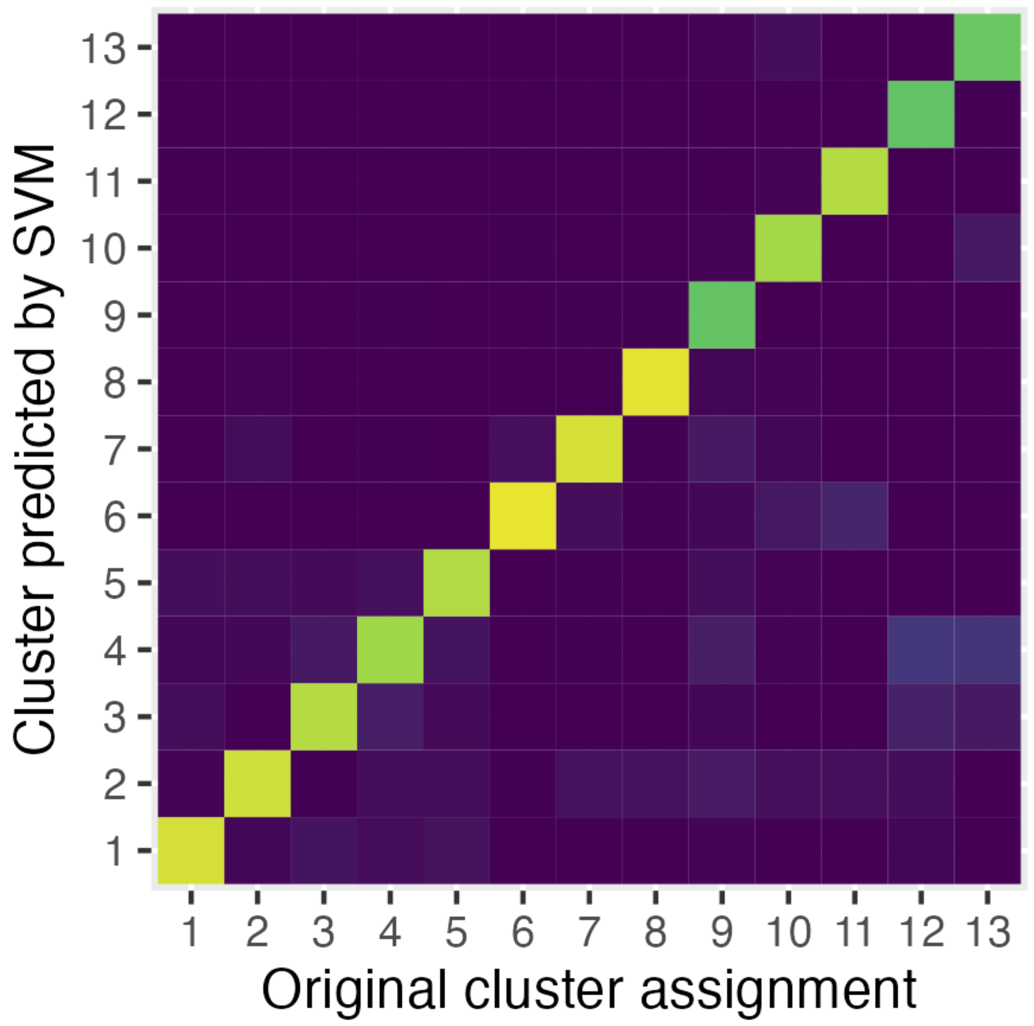
Data thinning

All Kidney Cells



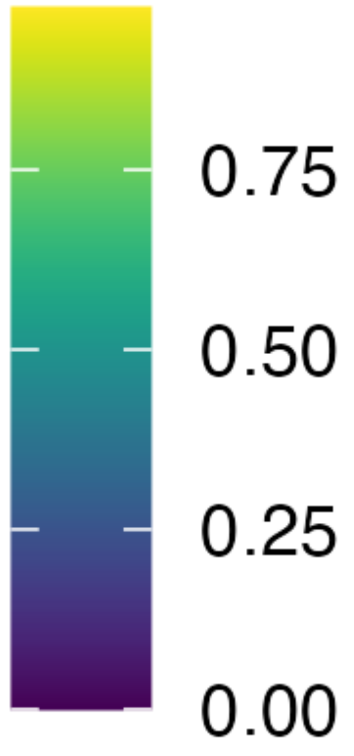
Adjusted Rand index: 0.90

Metanephric Cells

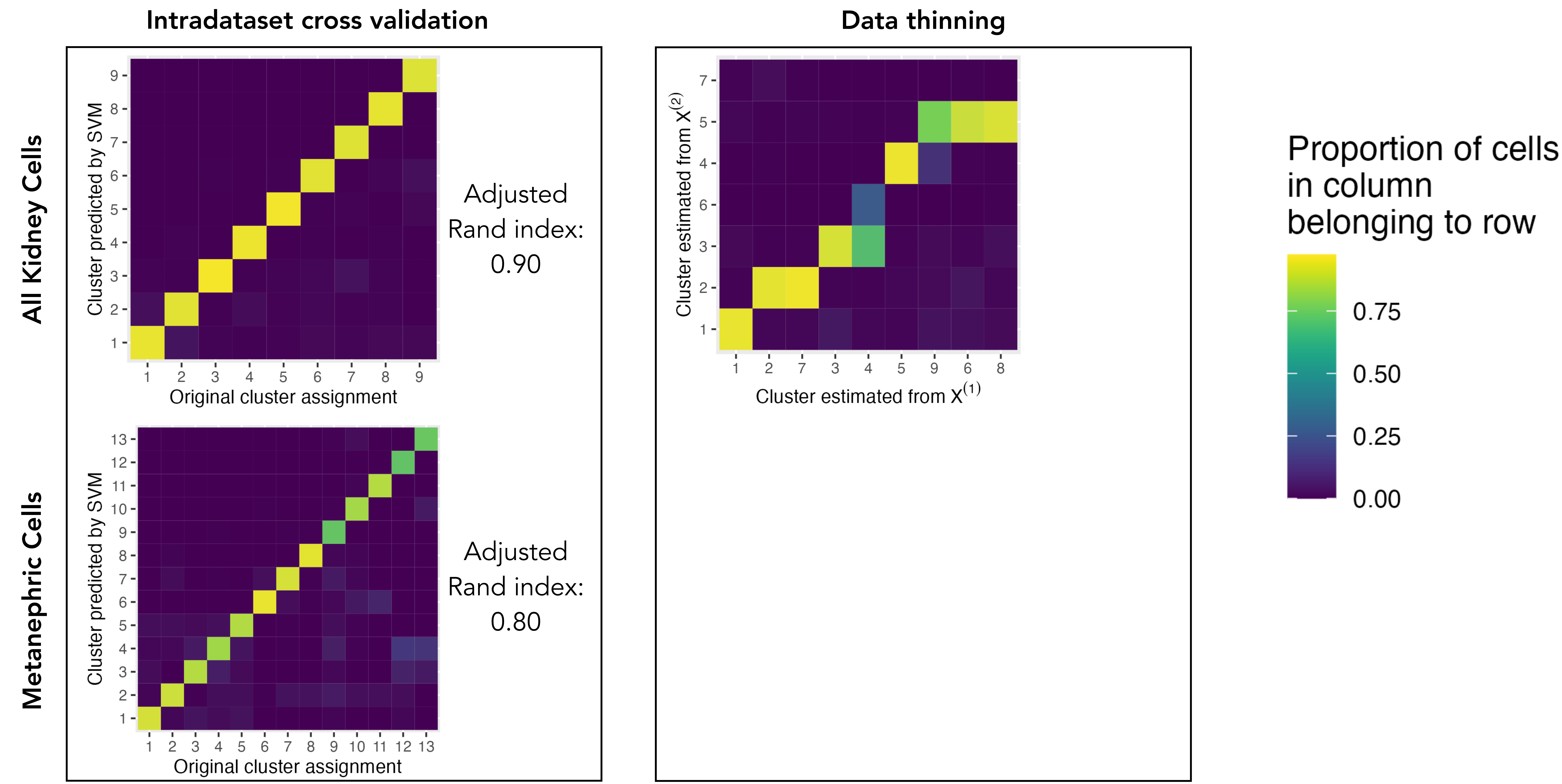


Adjusted Rand index: 0.80

Proportion of cells in column belonging to row



Re-analysis of Kidney cell data from fetal cell atlas



Adjusted Rand index: 0.90

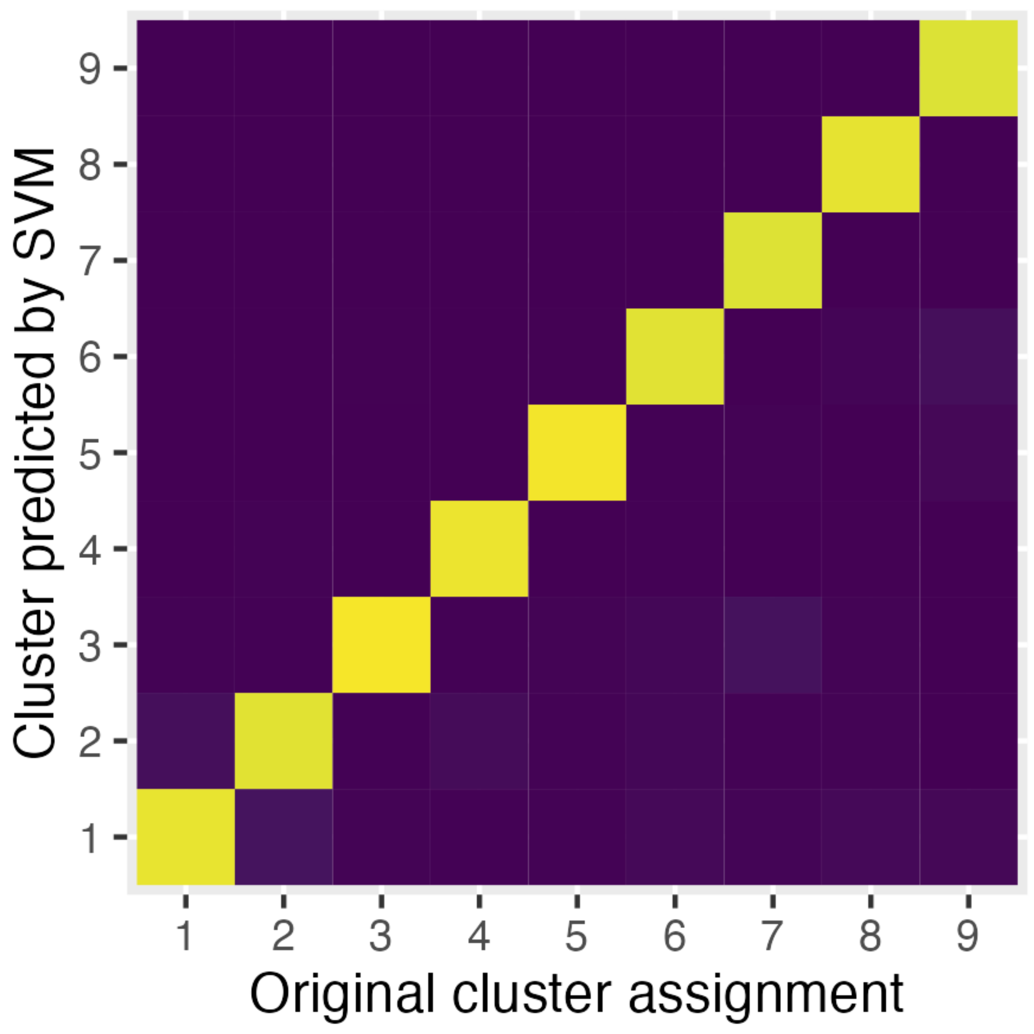
Adjusted Rand index: 0.80

Re-analysis of Kidney cell data from fetal cell atlas

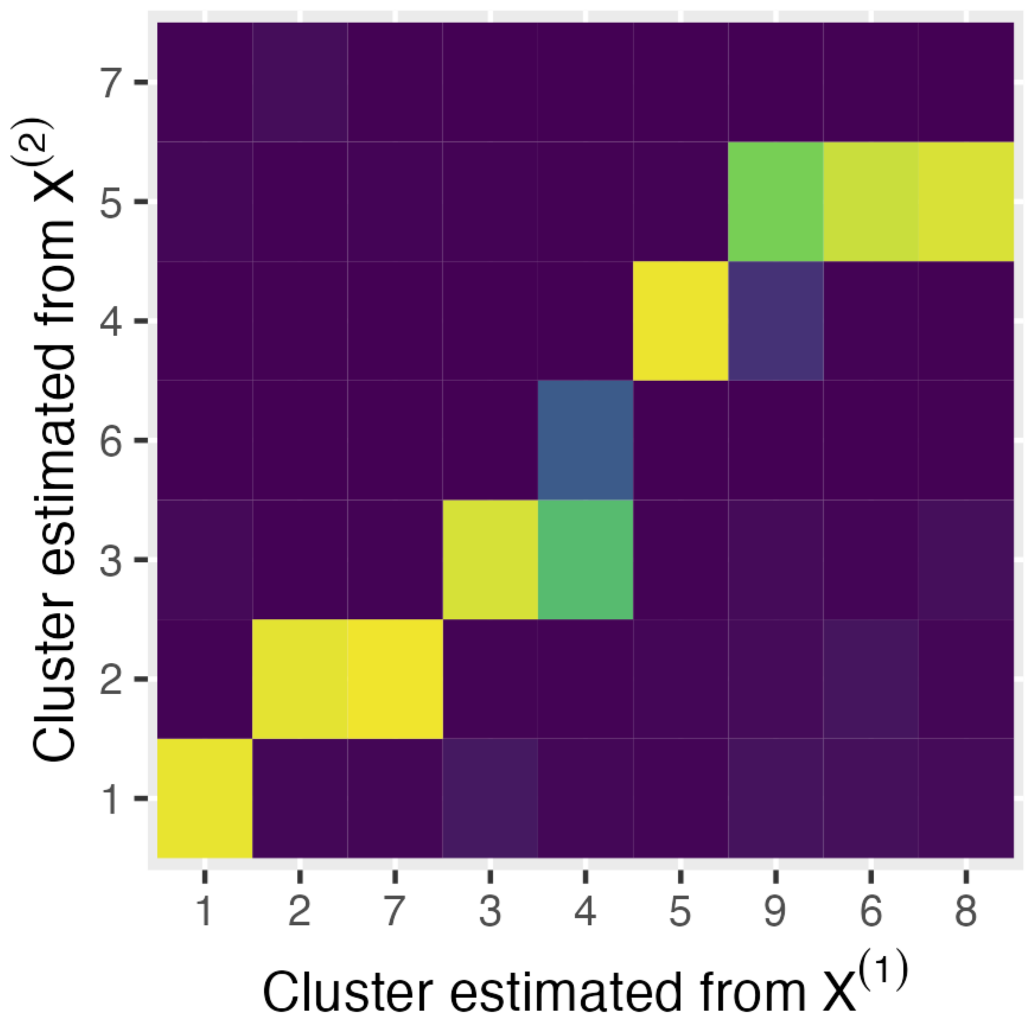
Intradataset cross validation

Data thinning

All Kidney Cells

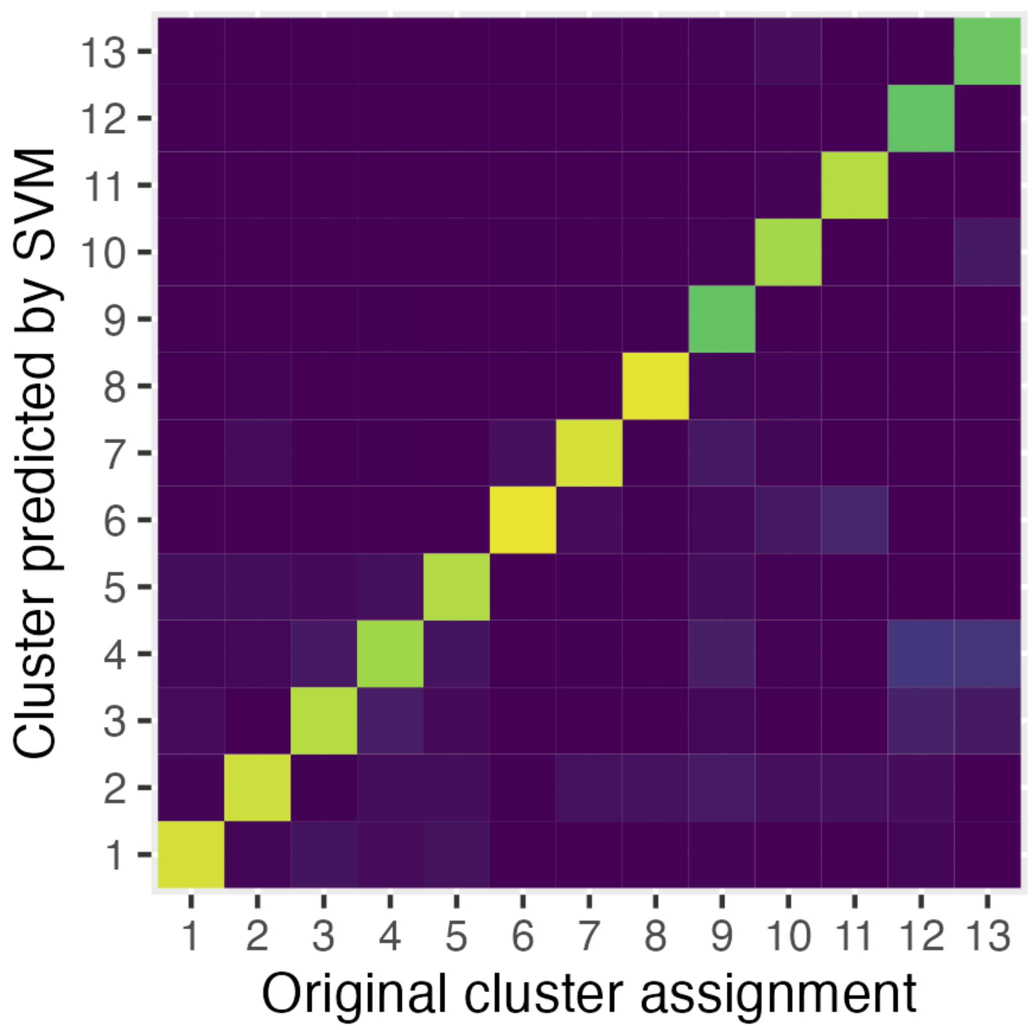


Adjusted Rand index: 0.90

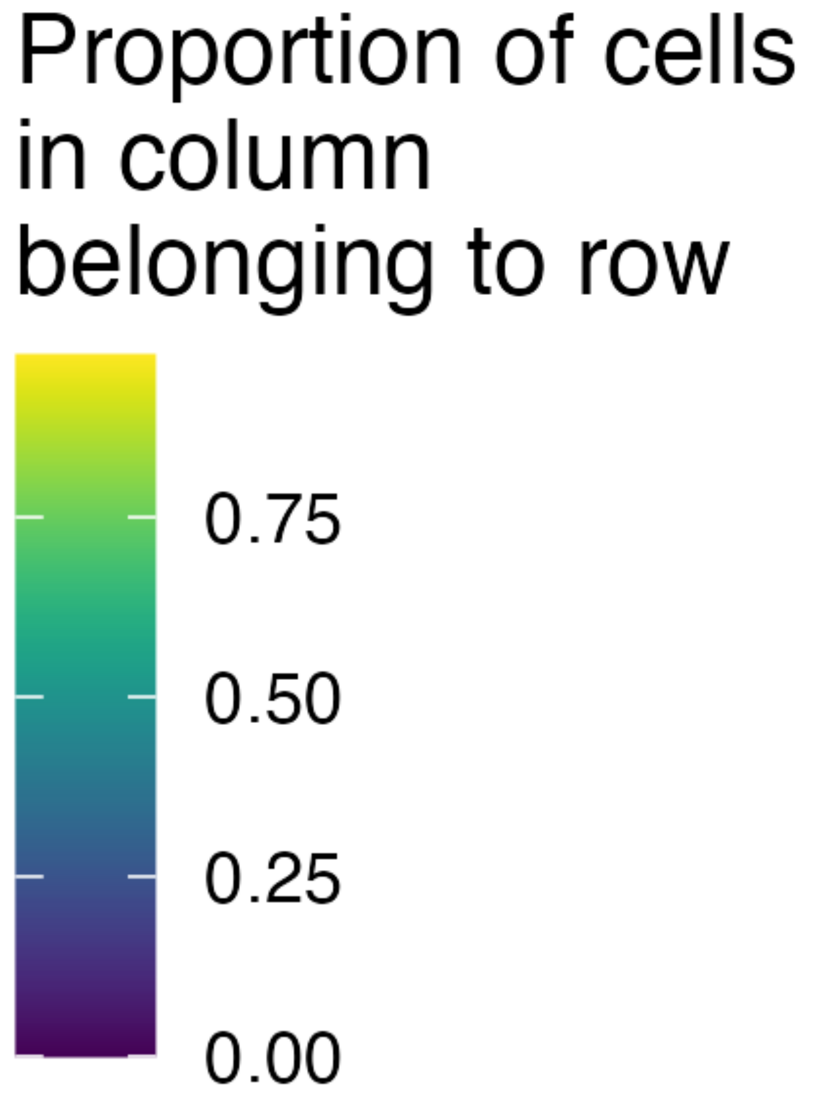


Adjusted Rand index: 0.87

Metanephric Cells

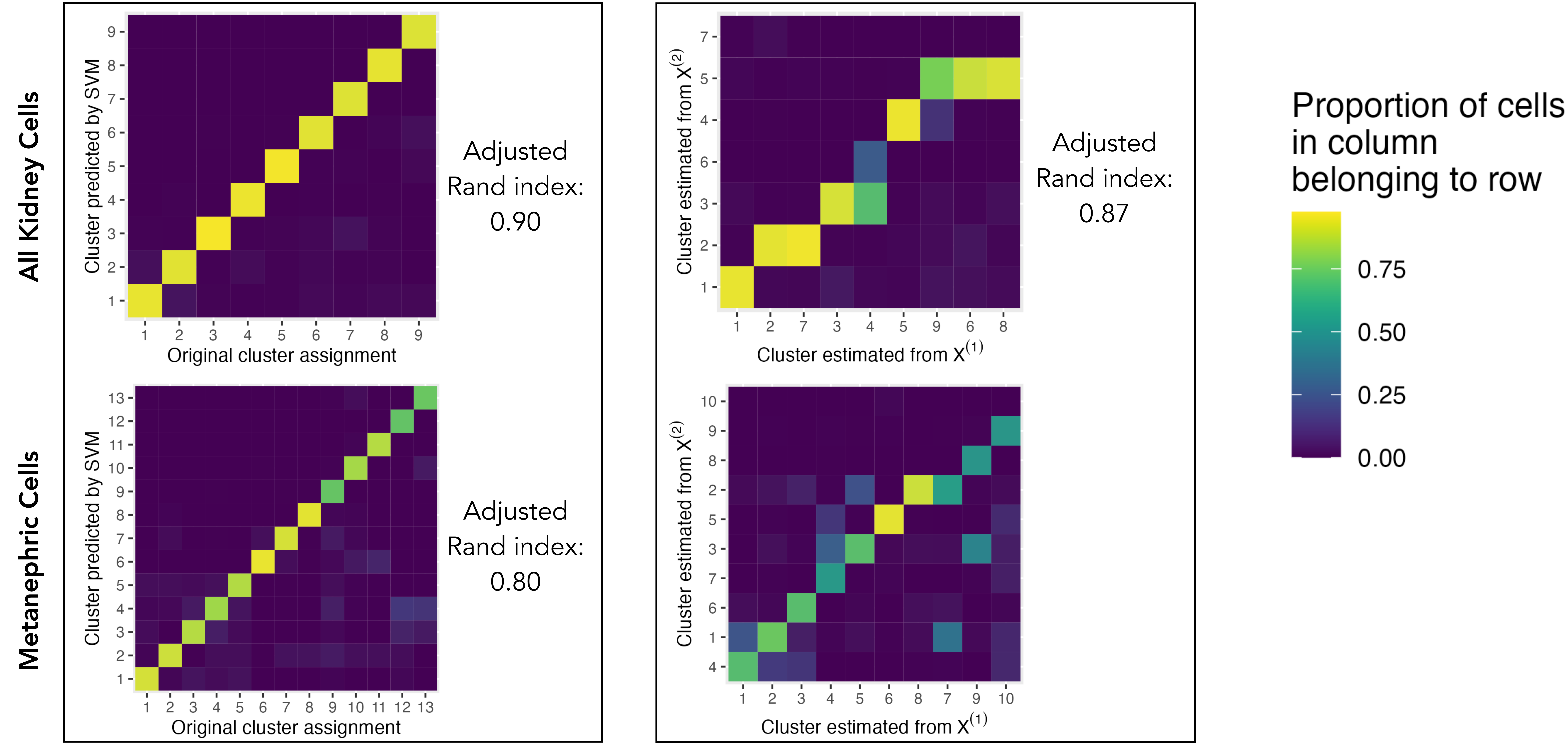


Adjusted Rand index: 0.80



Re-analysis of Kidney cell data from fetal cell atlas

Intradataset cross validation **Data thinning**

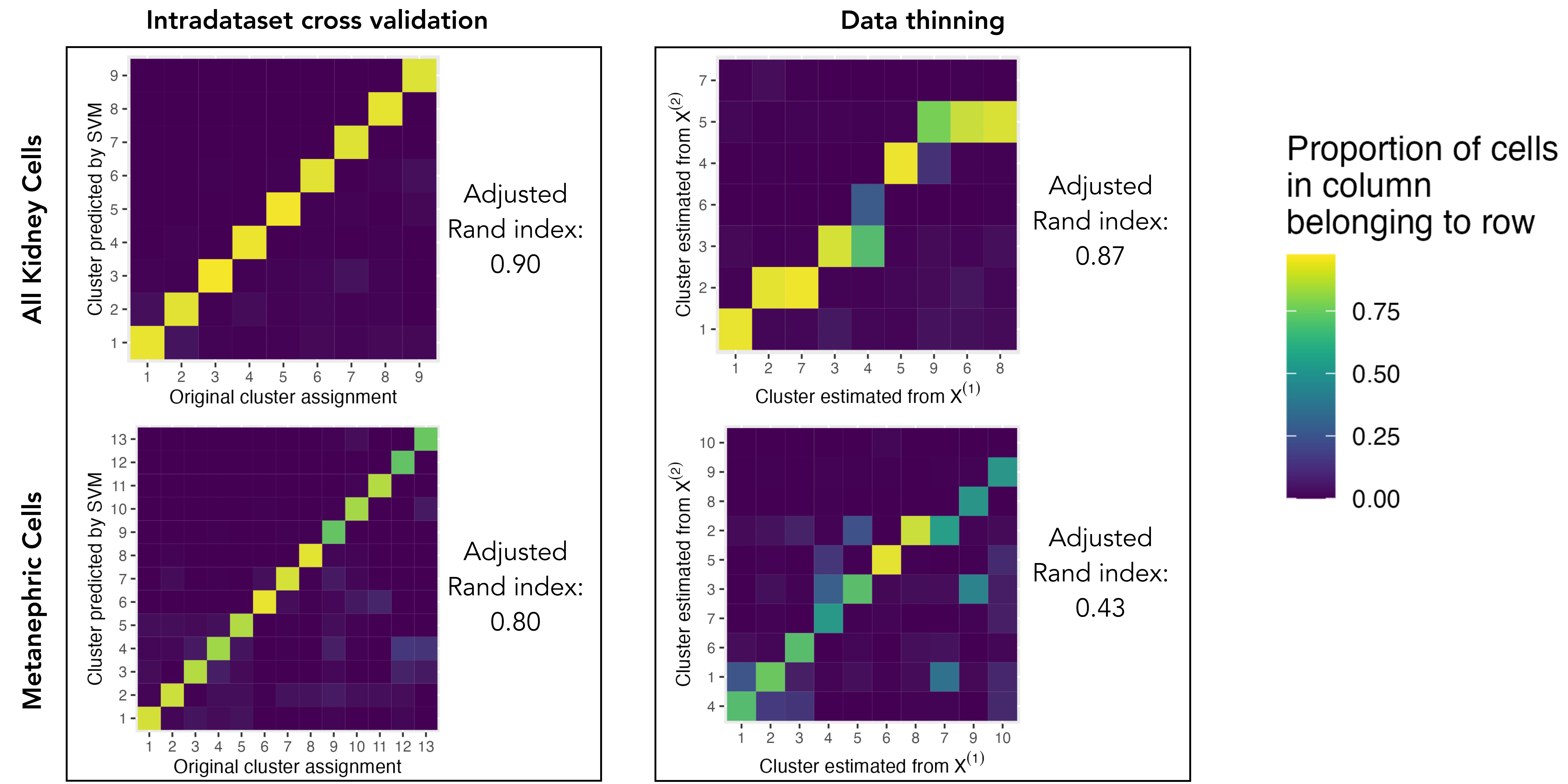


All Kidney Cells, intradataset cross validation: Adjusted Rand index: 0.90

All Kidney Cells, data thinning: Adjusted Rand index: 0.87

Metanephric Cells, intradataset cross validation: Adjusted Rand index: 0.80

Re-analysis of Kidney cell data from fetal cell atlas



Intradataset cross validation

Data thinning

All Kidney Cells

Adjusted Rand index: 0.90

Adjusted Rand index: 0.87

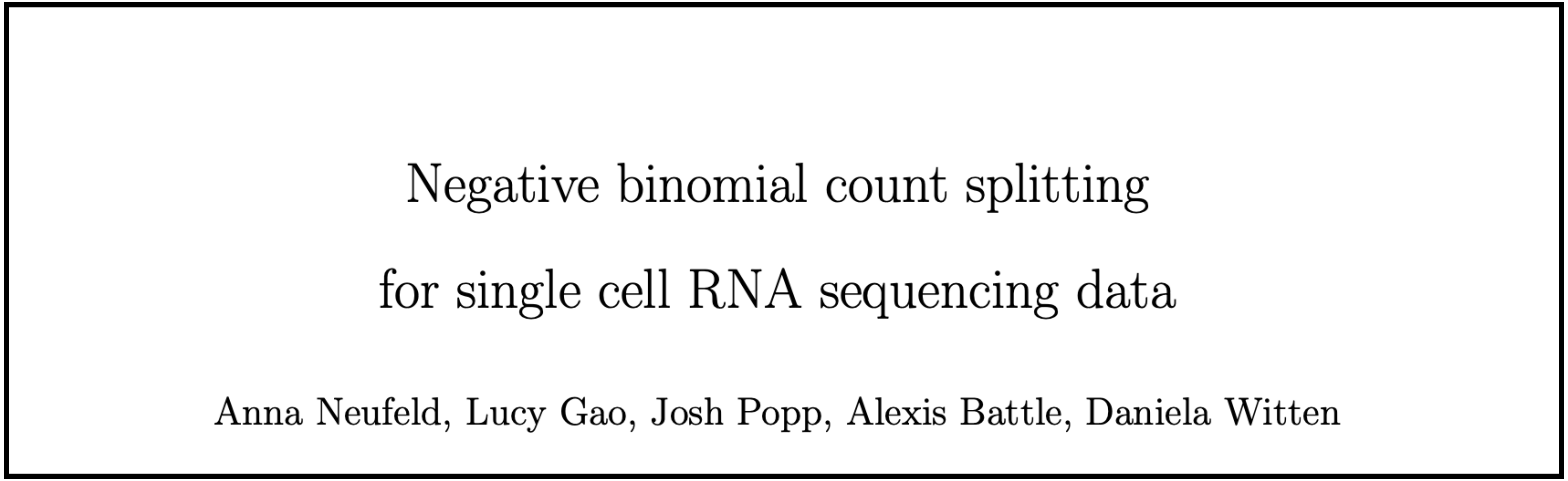
Metanephric Cells

Adjusted Rand index: 0.80

Adjusted Rand index: 0.43

Negative binomial data thinning is useful in the analysis of single-cell RNA sequencing data

Project 4



Negative binomial count splitting
for single cell RNA sequencing data

Anna Neufeld, Lucy Gao, Josh Popp, Alexis Battle, Daniela Witten

Arxiv preprint will be posted soon!

Outline

1. Motivation: settings where sample splitting doesn’t work
2. Poisson thinning
3. Data thinning
4. Application to single-cell RNA sequencing data
5. **Ongoing work**

Three ways to avoid double dipping

1. Specialized methods, such as selective inference.

2. Sample splitting.

3. Data thinning.

Three ways to avoid double dipping

1. Specialized methods, such as selective inference.

   Requires a bespoke solution for every problem at hand.

2. Sample splitting.

3. Data thinning.

Three ways to avoid double dipping

1. Specialized methods, such as selective inference.

   Requires a bespoke solution for every problem at hand.

2. Sample splitting.

   Super flexible!

3. Data thinning.

Three ways to avoid double dipping

1. Specialized methods, such as selective inference.

   Requires a bespoke solution for every problem at hand.

2. Sample splitting.

   Super flexible!

   Not an option in some unsupervised settings; unsatisfying in other settings.

3. Data thinning.

Three ways to avoid double dipping

1. Specialized methods, such as selective inference.

   Requires a bespoke solution for every problem at hand.

2. Sample splitting.

   Super flexible!

   Not an option in some unsupervised settings; unsatisfying in other settings.

3. Data thinning.

   No bespoke solutions needed; works in supervised and unsupervised settings.

Three ways to avoid double dipping

1. Specialized methods, such as selective inference.

   Requires a bespoke solution for every problem at hand.

2. Sample splitting.

   Super flexible!

   Not an option in some unsupervised settings; unsatisfying in other settings.

3. Data thinning.

   No bespoke solutions needed; works in supervised and unsupervised settings.

   Requires distributional assumptions and knowledge of nuisance parameters.

Three ways to avoid double dipping

1. Specialized methods, such as selective inference.

   Requires a bespoke solution for every problem at hand.

2. Sample splitting.

   Super flexible!

   Not an option in some unsupervised settings; unsatisfying in other settings.

3. Data thinning.

   No bespoke solutions needed; works in supervised and unsupervised settings.

   Requires distributional assumptions and knowledge of nuisance parameters.

   Limited to convolution-closed distributions?

Revisiting the goals of data thinning

Goal: split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

(1) $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.

(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

Revisiting the goals of data thinning

Goal: split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

(1) $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.

(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

In our previous recipe:

(3) $X = X^{(1)} + X^{(2)}$.

Revisiting the goals of data thinning

Goal: split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

(1) $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.

(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

In our previous recipe:

~~**(3)** $X = X^{(1)} + X^{(2)}$~~.

Revisiting the goals of data thinning

Goal: split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

(1) $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.

(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

In our previous recipe:

~~**(3)** $X = X^{(1)} + X^{(2)}$~~. **(3)** $X = T(X^{(1)}, X^{(2)})$.

Revisiting the goals of data thinning

Goal: split a single observation X into $X^{(1)}$ and $X^{(2)}$ such that:

~~**(1)** $X^{(1)}$ and $X^{(2)}$ have the same distribution as X, up to a parameter scaling.~~

(2) $X^{(1)} \perp\!\!\!\perp X^{(2)}$.

In our previous recipe:

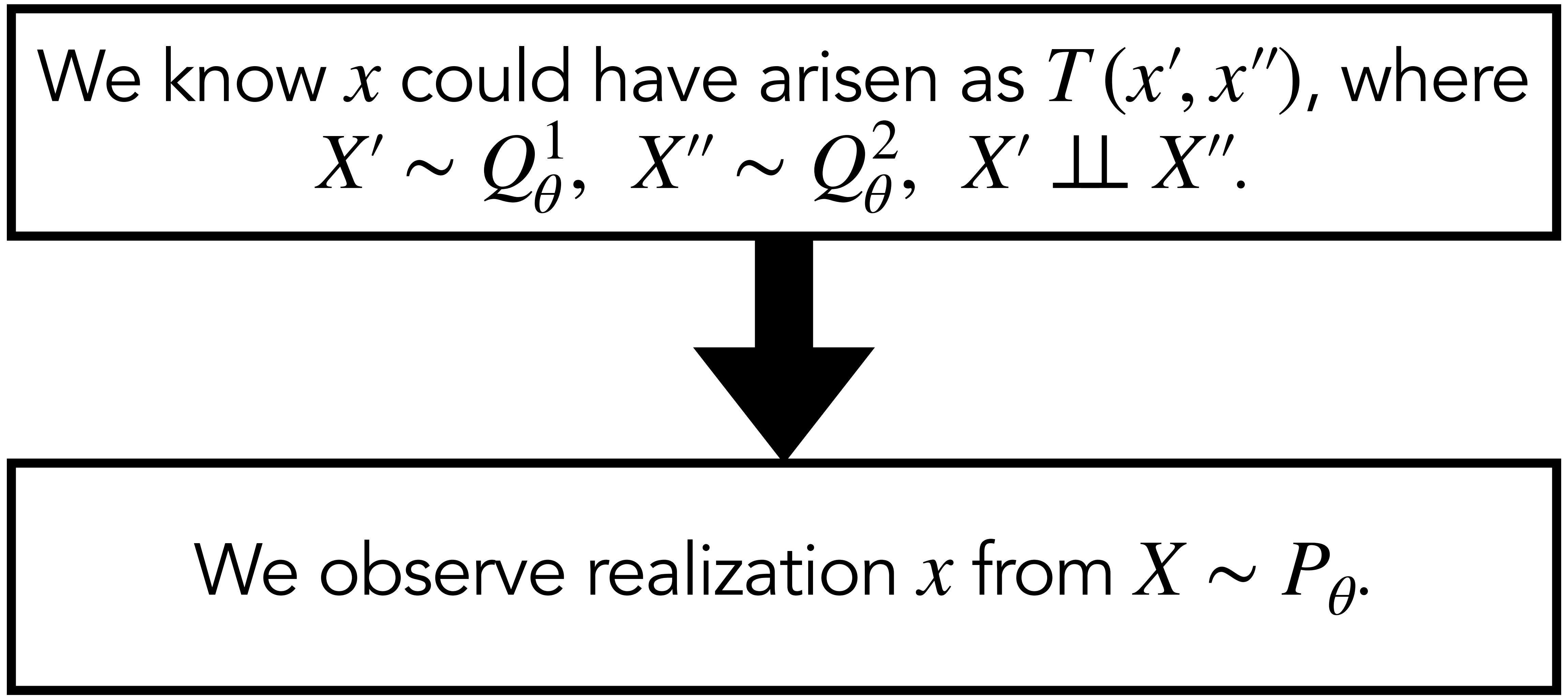
~~**(3)** $X = X^{(1)} + X^{(2)}$~~. **(3)** $X = T(X^{(1)}, X^{(2)})$.

Generalized thinning with non-additive decompositions

Generalized thinning with non-additive decompositions

We observe realization x from $X \sim P_\theta$.

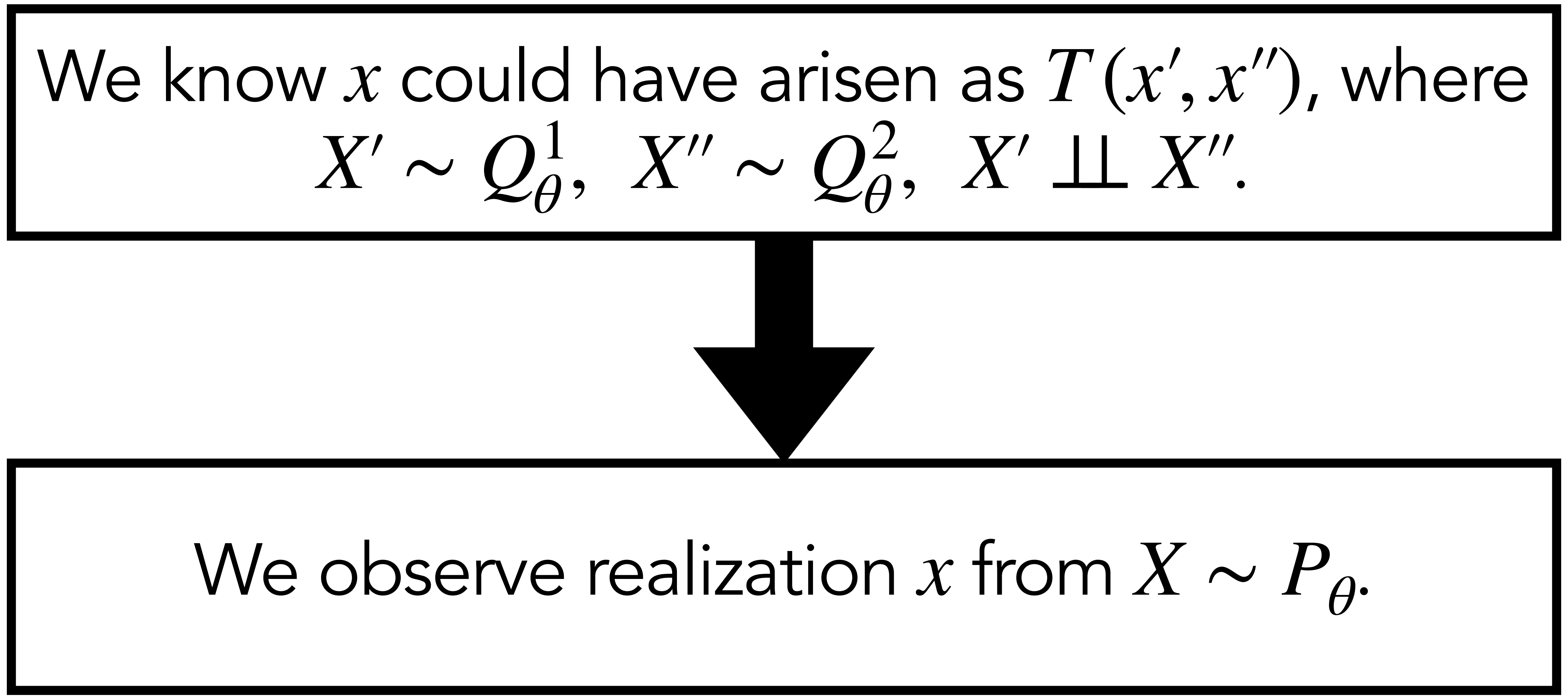
Generalized thinning with non-additive decompositions



We know x could have arisen as $T(x', x'')$, where $X' \sim Q^1_\theta,\ \ X'' \sim Q^2_\theta,\ \ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim P_\theta$.

Generalized thinning with non-additive decompositions

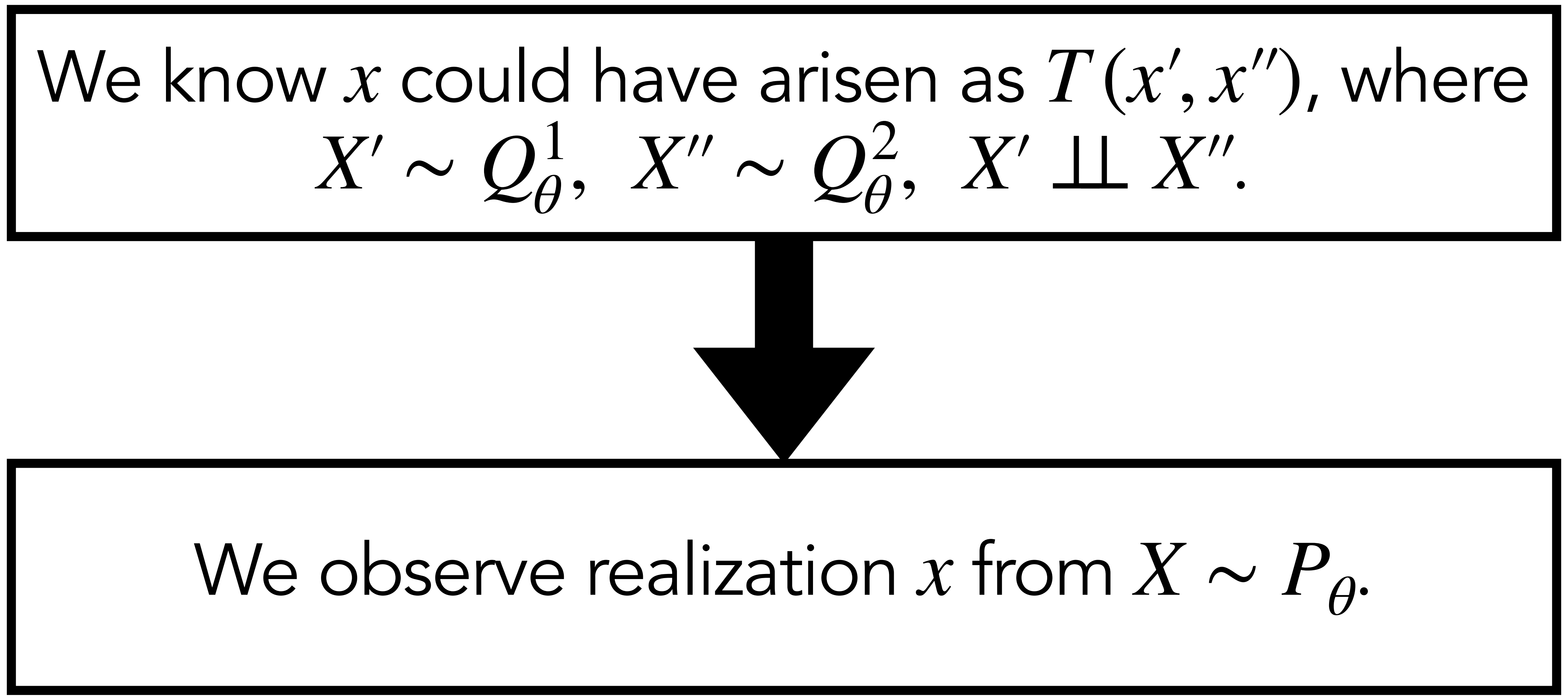


We know x could have arisen as $T(x', x'')$, where $X' \sim Q^1_\theta,\ X'' \sim Q^2_\theta,\ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim P_\theta$.

Can we work backwards to recover x' and x''?

Generalized thinning with non-additive decompositions



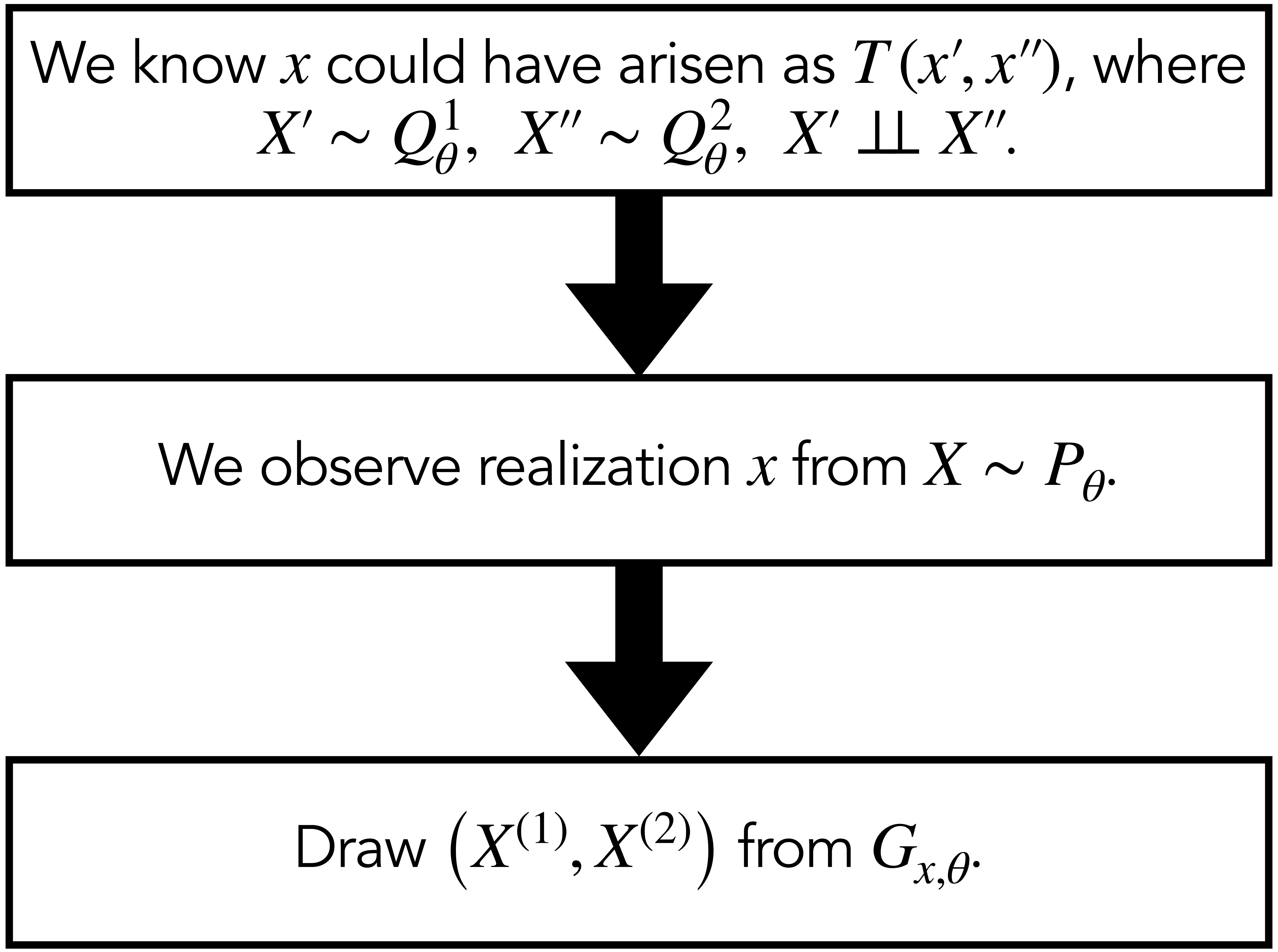
We know x could have arisen as $T(x', x'')$, where $X' \sim Q_\theta^1,\ \ X'' \sim Q_\theta^2,\ \ X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim P_\theta$.

Can we work backwards to recover x' and x''?

Let $G_{x,\theta}$ be the conditional distribution of $(X', X'') \mid X = x$.

Generalized thinning with non-additive decompositions



We know x could have arisen as $T(x', x'')$, where $X' \sim Q^1_\theta, \;\; X'' \sim Q^2_\theta, \;\; X' \perp\!\!\!\perp X''$.

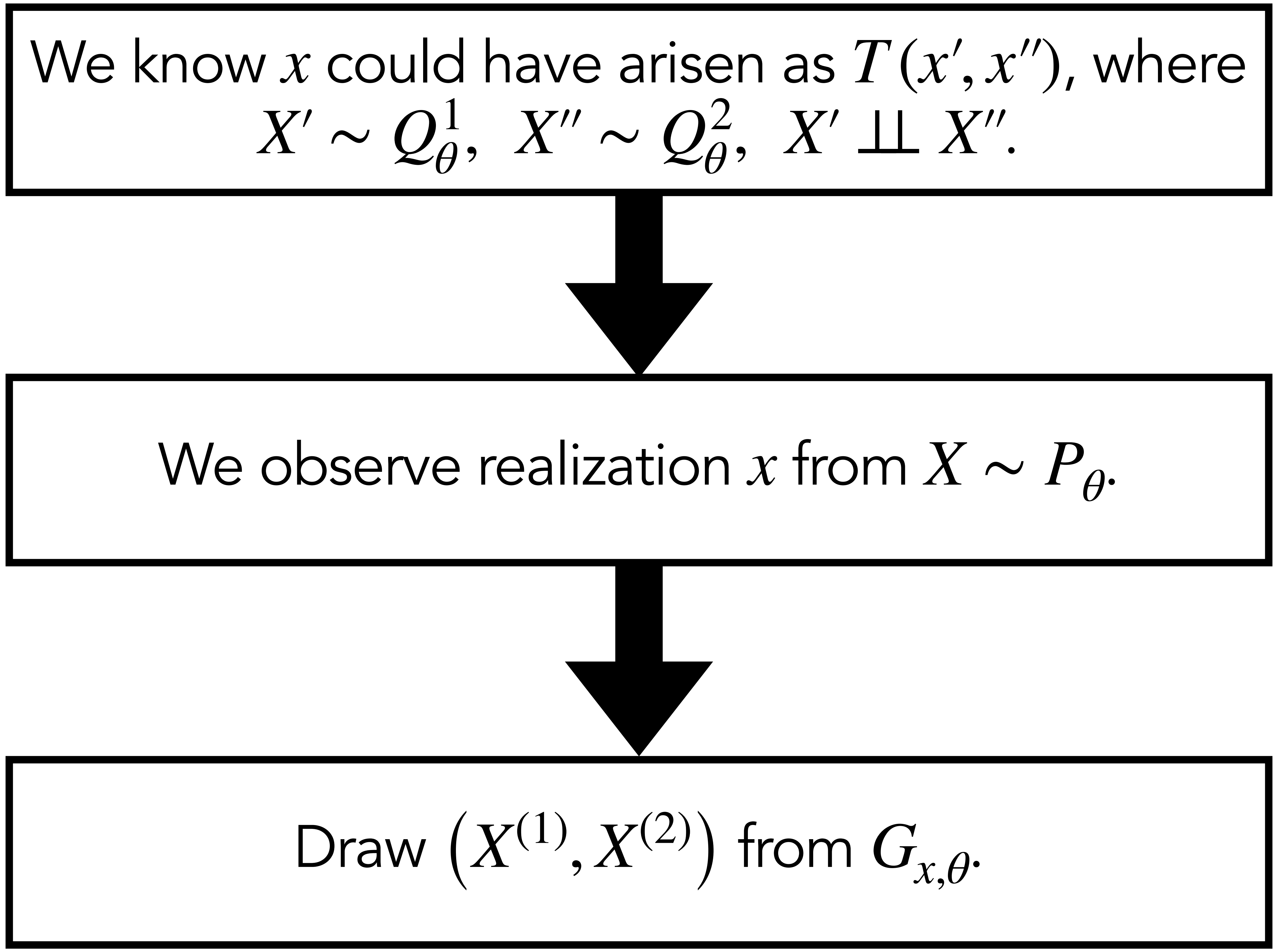
We observe realization x from $X \sim P_\theta$.

Draw $\left(X^{(1)}, X^{(2)}\right)$ from $G_{x,\theta}$.

Can we work backwards to recover x' and x''?

Let $G_{x,\theta}$ be the conditional distribution of $(X', X'') \mid X = x$.

Generalized thinning with non-additive decompositions



We know x could have arisen as $T(x', x'')$, where $X' \sim Q_\theta^1, \;\; X'' \sim Q_\theta^2, \;\; X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim P_\theta$.

Draw $\left(X^{(1)}, X^{(2)}\right)$ from $G_{x,\theta}$.

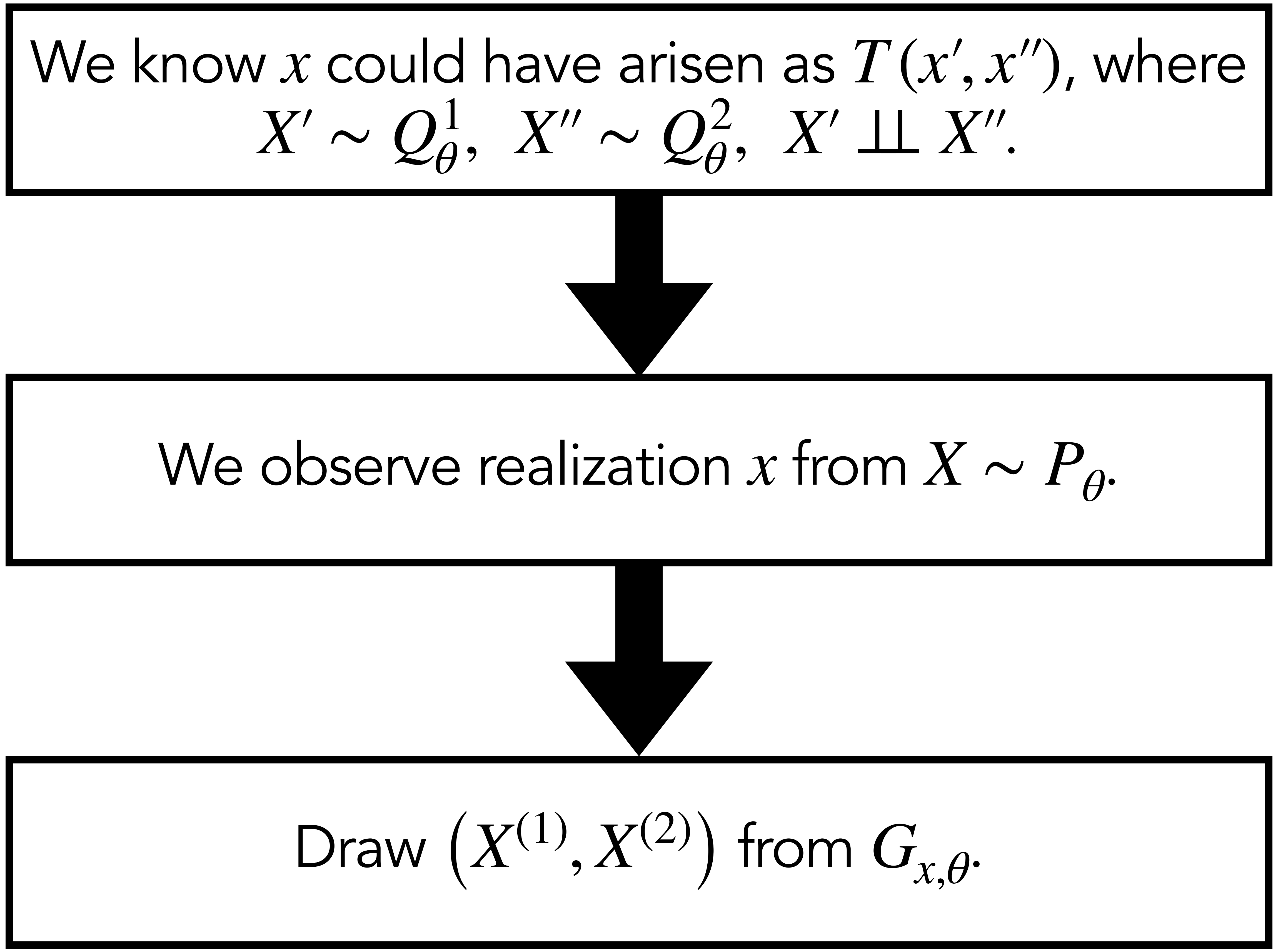
Can we work backwards to recover x' and x''?

Let $G_{x,\theta}$ be the conditional distribution of $(X', X'') \mid X = x$.

Theorem:

$$X^{(1)} \sim Q_\theta^1, \qquad X^{(2)} \sim Q_\theta^2, \qquad X^{(1)} \perp\!\!\!\perp X^{(2)}.$$

Generalized thinning with non-additive decompositions



We know x could have arisen as $T(x', x'')$, where $X' \sim Q^1_\theta, \;\; X'' \sim Q^2_\theta, \;\; X' \perp\!\!\!\perp X''$.

We observe realization x from $X \sim P_\theta$.

Draw $\left(X^{(1)}, X^{(2)}\right)$ from $G_{x,\theta}$.

Theorem:

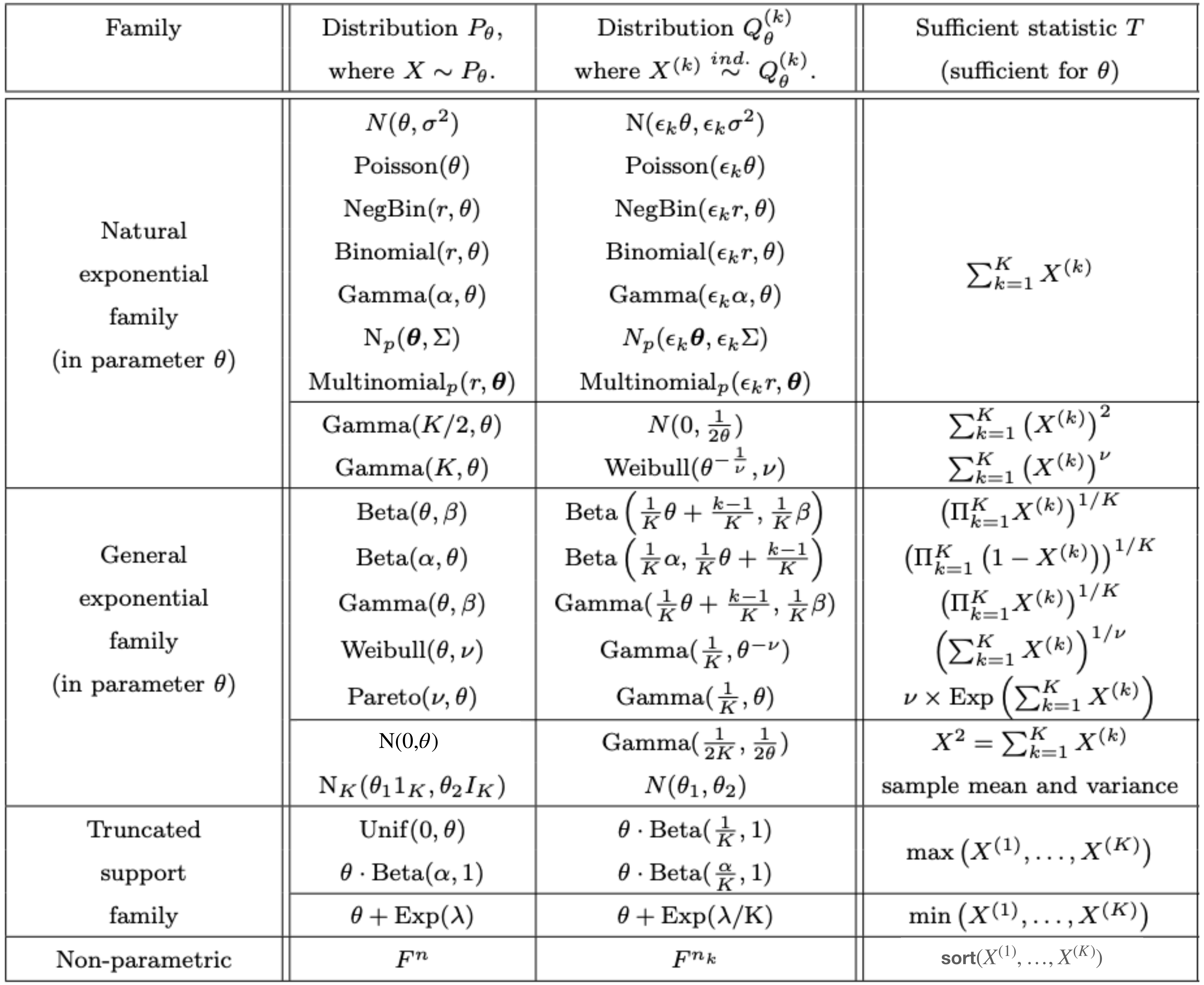
$$X^{(1)} \sim Q^1_\theta, \quad X^{(2)} \sim Q^2_\theta, \quad X^{(1)} \perp\!\!\!\perp X^{(2)}.$$

Can we work backwards to recover x' and x''?

Let $G_{x,\theta}$ be the conditional distribution of $(X', X'') \mid X = x$.

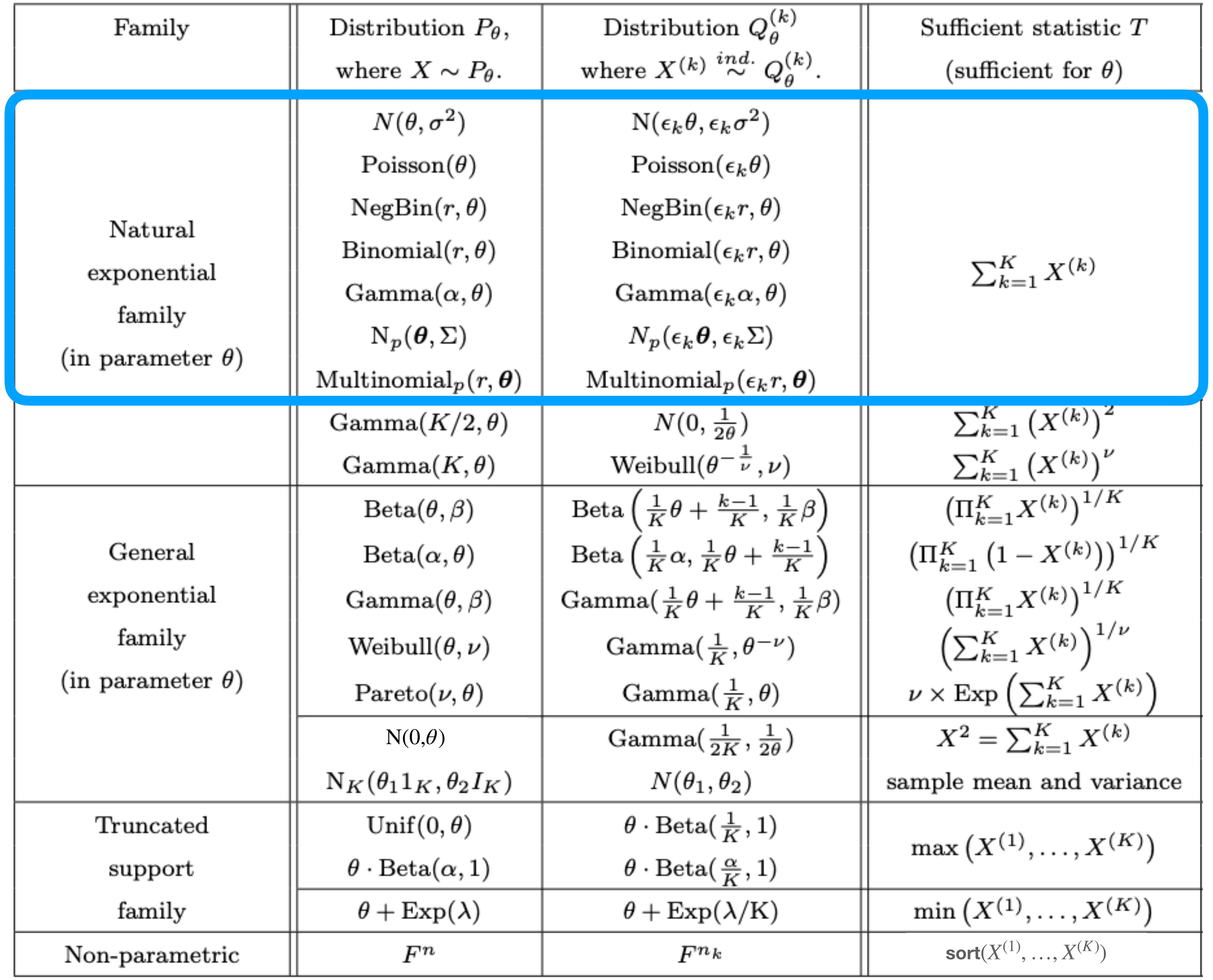
Key idea: If $X = T(X', X'')$ is sufficient for θ in the joint of (X', X''), then $G_{x,\theta}$ does not depend on θ.

The list of distributions we can thin is extensive



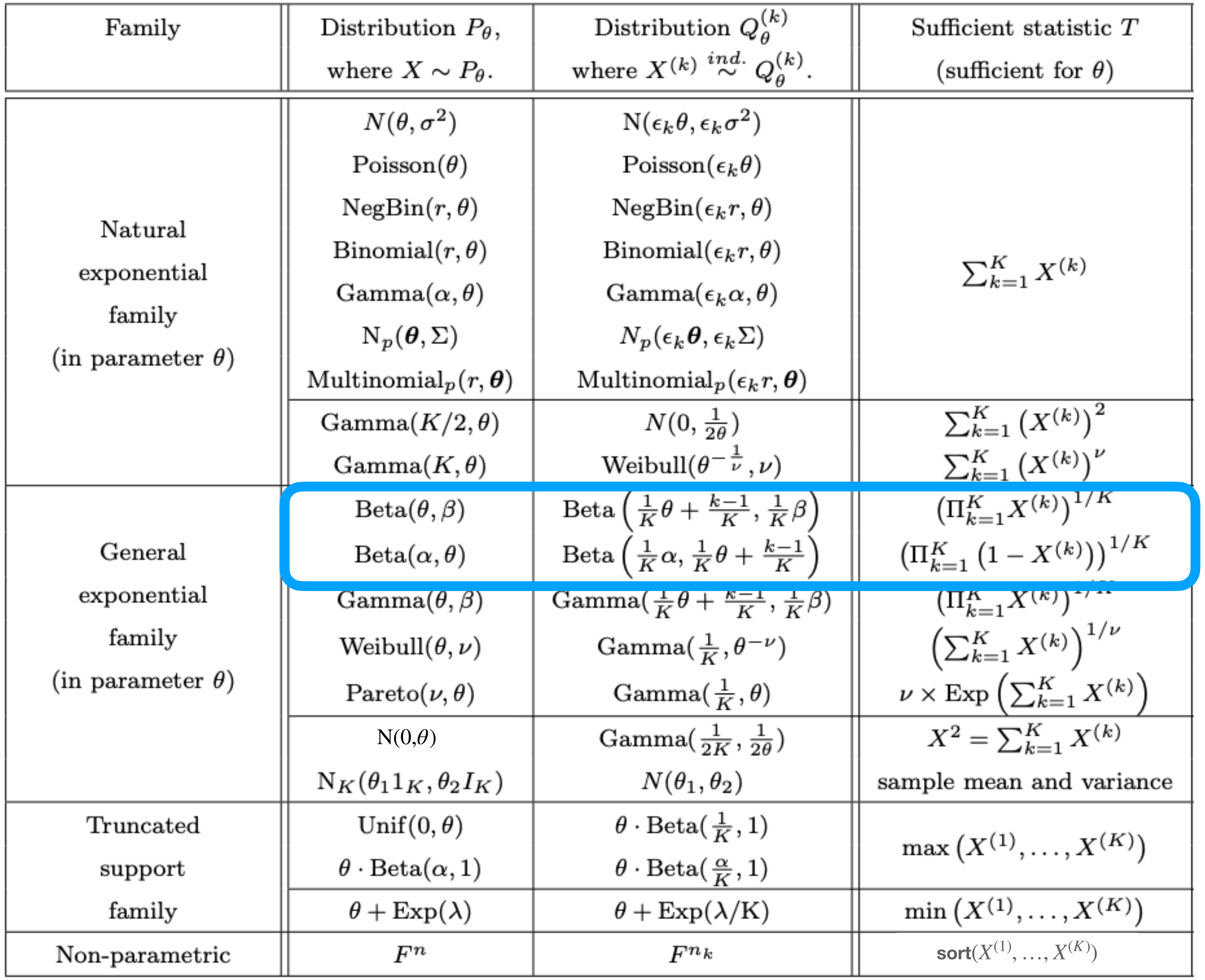
Family	Distribution P_θ, where $X \sim P_\theta$.	Distribution $Q_\theta^{(k)}$ where $X^{(k)} \overset{ind.}{\sim} Q_\theta^{(k)}$.	Sufficient statistic T (sufficient for θ)
Natural exponential family (in parameter θ)	$N(\theta, \sigma^2)$ Poisson(θ) NegBin(r, θ) Binomial(r, θ) Gamma(α, θ) $\mathrm{N}_p(\boldsymbol{\theta}, \Sigma)$ Multinomial$_p(r, \boldsymbol{\theta})$	$\mathrm{N}(\epsilon_k\theta, \epsilon_k\sigma^2)$ Poisson$(\epsilon_k\theta)$ NegBin$(\epsilon_k r, \theta)$ Binomial$(\epsilon_k r, \theta)$ Gamma$(\epsilon_k\alpha, \theta)$ $N_p(\epsilon_k\boldsymbol{\theta}, \epsilon_k\Sigma)$ Multinomial$_p(\epsilon_k r, \boldsymbol{\theta})$	$\sum_{k=1}^K X^{(k)}$
	Gamma$(K/2, \theta)$ Gamma(K, θ)	$N(0, \frac{1}{2\theta})$ Weibull$(\theta^{-\frac{1}{\nu}}, \nu)$	$\sum_{k=1}^K \left(X^{(k)}\right)^2$ $\sum_{k=1}^K \left(X^{(k)}\right)^\nu$
General exponential family (in parameter θ)	Beta(θ, β) Beta(α, θ) Gamma(θ, β) Weibull(θ, ν) Pareto(ν, θ)	Beta $\left(\frac{1}{K}\theta + \frac{k-1}{K}, \frac{1}{K}\beta\right)$ Beta $\left(\frac{1}{K}\alpha, \frac{1}{K}\theta + \frac{k-1}{K}\right)$ Gamma$(\frac{1}{K}\theta + \frac{k-1}{K}, \frac{1}{K}\beta)$ Gamma$(\frac{1}{K}, \theta^{-\nu})$ Gamma$(\frac{1}{K}, \theta)$	$\left(\Pi_{k=1}^K X^{(k)}\right)^{1/K}$ $\left(\Pi_{k=1}^K \left(1 - X^{(k)}\right)\right)^{1/K}$ $\left(\Pi_{k=1}^K X^{(k)}\right)^{1/K}$ $\left(\sum_{k=1}^K X^{(k)}\right)^{1/\nu}$ $\nu \times \mathrm{Exp}\left(\sum_{k=1}^K X^{(k)}\right)$
	N$(0,\theta)$ $\mathrm{N}_K(\theta_1 1_K, \theta_2 I_K)$	Gamma$(\frac{1}{2K}, \frac{1}{2\theta})$ $N(\theta_1, \theta_2)$	$X^2 = \sum_{k=1}^K X^{(k)}$ sample mean and variance
Truncated support family	Unif$(0, \theta)$ $\theta \cdot$ Beta$(\alpha, 1)$	$\theta \cdot$ Beta$(\frac{1}{K}, 1)$ $\theta \cdot$ Beta$(\frac{\alpha}{K}, 1)$	max $\left(X^{(1)}, \ldots, X^{(K)}\right)$
	$\theta + \mathrm{Exp}(\lambda)$	$\theta + \mathrm{Exp}(\lambda/\mathrm{K})$	min $\left(X^{(1)}, \ldots, X^{(K)}\right)$
Non-parametric	F^n	F^{n_k}	sort$(X^{(1)}, \ldots, X^{(K)})$

The list of distributions we can thin is extensive



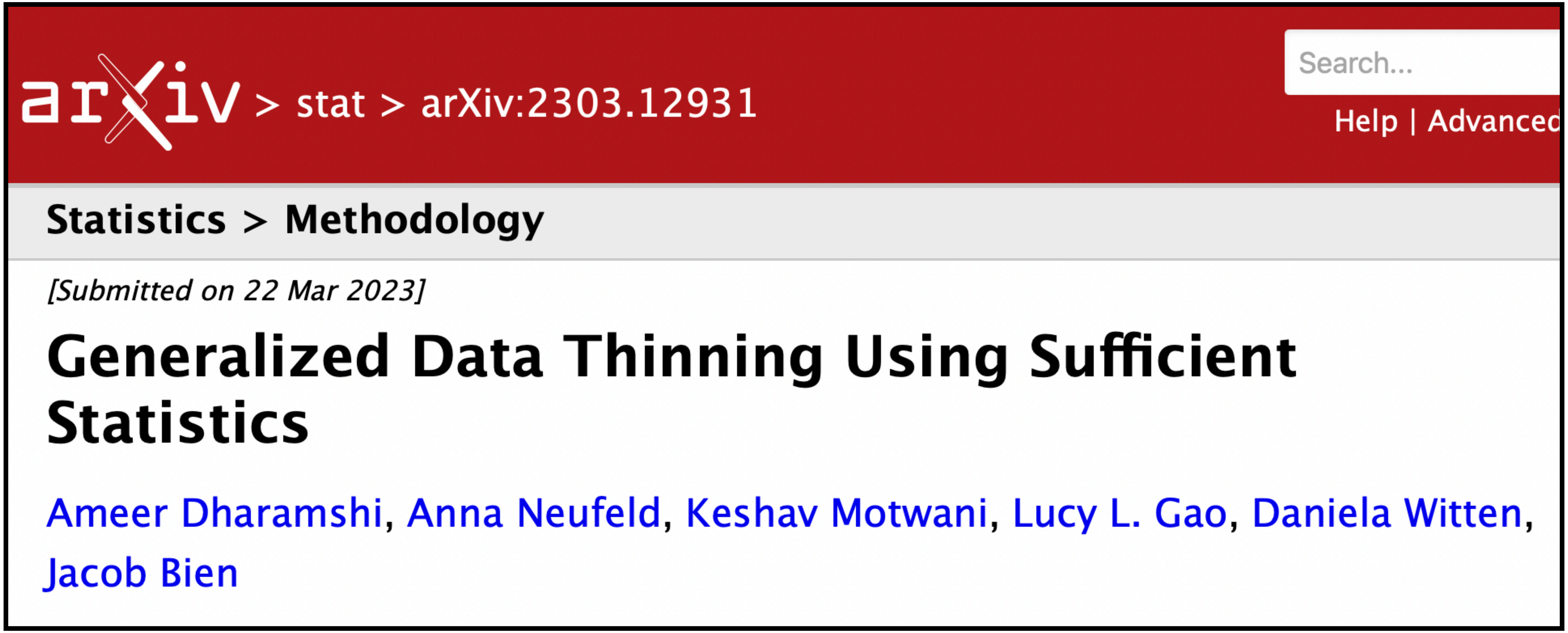
Family	Distribution P_θ, where $X \sim P_\theta$.	Distribution $Q_\theta^{(k)}$ where $X^{(k)} \overset{ind.}{\sim} Q_\theta^{(k)}$.	Sufficient statistic T (sufficient for θ)
Natural exponential family (in parameter θ)	$N(\theta, \sigma^2)$	$\mathrm{N}(\epsilon_k \theta, \epsilon_k \sigma^2)$	$\sum_{k=1}^K X^{(k)}$
	$\mathrm{Poisson}(\theta)$	$\mathrm{Poisson}(\epsilon_k \theta)$	
	$\mathrm{NegBin}(r, \theta)$	$\mathrm{NegBin}(\epsilon_k r, \theta)$	
	$\mathrm{Binomial}(r, \theta)$	$\mathrm{Binomial}(\epsilon_k r, \theta)$	
	$\mathrm{Gamma}(\alpha, \theta)$	$\mathrm{Gamma}(\epsilon_k \alpha, \theta)$	
	$\mathrm{N}_p(\boldsymbol{\theta}, \Sigma)$	$N_p(\epsilon_k \boldsymbol{\theta}, \epsilon_k \Sigma)$	
	$\mathrm{Multinomial}_p(r, \boldsymbol{\theta})$	$\mathrm{Multinomial}_p(\epsilon_k r, \boldsymbol{\theta})$	
	$\mathrm{Gamma}(K/2, \theta)$	$N(0, \frac{1}{2\theta})$	$\sum_{k=1}^K \left(X^{(k)}\right)^2$
	$\mathrm{Gamma}(K, \theta)$	$\mathrm{Weibull}(\theta^{-\frac{1}{\nu}}, \nu)$	$\sum_{k=1}^K \left(X^{(k)}\right)^\nu$
General exponential family (in parameter θ)	$\mathrm{Beta}(\theta, \beta)$	$\mathrm{Beta}\left(\frac{1}{K}\theta + \frac{k-1}{K}, \frac{1}{K}\beta\right)$	$\left(\Pi_{k=1}^K X^{(k)}\right)^{1/K}$
	$\mathrm{Beta}(\alpha, \theta)$	$\mathrm{Beta}\left(\frac{1}{K}\alpha, \frac{1}{K}\theta + \frac{k-1}{K}\right)$	$\left(\Pi_{k=1}^K \left(1 - X^{(k)}\right)\right)^{1/K}$
	$\mathrm{Gamma}(\theta, \beta)$	$\mathrm{Gamma}(\frac{1}{K}\theta + \frac{k-1}{K}, \frac{1}{K}\beta)$	$\left(\Pi_{k=1}^K X^{(k)}\right)^{1/K}$
	$\mathrm{Weibull}(\theta, \nu)$	$\mathrm{Gamma}(\frac{1}{K}, \theta^{-\nu})$	$\left(\sum_{k=1}^K X^{(k)}\right)^{1/\nu}$
	$\mathrm{Pareto}(\nu, \theta)$	$\mathrm{Gamma}(\frac{1}{K}, \theta)$	$\nu \times \mathrm{Exp}\left(\sum_{k=1}^K X^{(k)}\right)$
	$\mathrm{N}(0,\theta)$	$\mathrm{Gamma}(\frac{1}{2K}, \frac{1}{2\theta})$	$X^2 = \sum_{k=1}^K X^{(k)}$
	$\mathrm{N}_K(\theta_1 1_K, \theta_2 I_K)$	$N(\theta_1, \theta_2)$	sample mean and variance
Truncated support family	$\mathrm{Unif}(0, \theta)$	$\theta \cdot \mathrm{Beta}(\frac{1}{K}, 1)$	$\max\left(X^{(1)}, \ldots, X^{(K)}\right)$
	$\theta \cdot \mathrm{Beta}(\alpha, 1)$	$\theta \cdot \mathrm{Beta}(\frac{\alpha}{K}, 1)$	
	$\theta + \mathrm{Exp}(\lambda)$	$\theta + \mathrm{Exp}(\lambda/\mathrm{K})$	$\min\left(X^{(1)}, \ldots, X^{(K)}\right)$
Non-parametric	F^n	F^{n_k}	sort$(X^{(1)}, \ldots, X^{(K)})$

The list of distributions we can thin is extensive



Family	Distribution P_θ, where $X \sim P_\theta$.	Distribution $Q_\theta^{(k)}$ where $X^{(k)} \overset{ind.}{\sim} Q_\theta^{(k)}$.	Sufficient statistic T (sufficient for θ)
Natural exponential family (in parameter θ)	$N(\theta, \sigma^2)$	$\mathrm{N}(\epsilon_k\theta, \epsilon_k\sigma^2)$	$\sum_{k=1}^K X^{(k)}$
	$\mathrm{Poisson}(\theta)$	$\mathrm{Poisson}(\epsilon_k\theta)$	
	$\mathrm{NegBin}(r, \theta)$	$\mathrm{NegBin}(\epsilon_k r, \theta)$	
	$\mathrm{Binomial}(r, \theta)$	$\mathrm{Binomial}(\epsilon_k r, \theta)$	
	$\mathrm{Gamma}(\alpha, \theta)$	$\mathrm{Gamma}(\epsilon_k\alpha, \theta)$	
	$\mathrm{N}_p(\boldsymbol{\theta}, \Sigma)$	$N_p(\epsilon_k\boldsymbol{\theta}, \epsilon_k\Sigma)$	
	$\mathrm{Multinomial}_p(r, \boldsymbol{\theta})$	$\mathrm{Multinomial}_p(\epsilon_k r, \boldsymbol{\theta})$	
	$\mathrm{Gamma}(K/2, \theta)$	$N(0, \frac{1}{2\theta})$	$\sum_{k=1}^K \left(X^{(k)}\right)^2$
	$\mathrm{Gamma}(K, \theta)$	$\mathrm{Weibull}(\theta^{-\frac{1}{\nu}}, \nu)$	$\sum_{k=1}^K \left(X^{(k)}\right)^\nu$
General exponential family (in parameter θ)	$\mathrm{Beta}(\theta, \beta)$	$\mathrm{Beta}\left(\frac{1}{K}\theta + \frac{k-1}{K}, \frac{1}{K}\beta\right)$	$\left(\Pi_{k=1}^K X^{(k)}\right)^{1/K}$
	$\mathrm{Beta}(\alpha, \theta)$	$\mathrm{Beta}\left(\frac{1}{K}\alpha, \frac{1}{K}\theta + \frac{k-1}{K}\right)$	$\left(\Pi_{k=1}^K \left(1 - X^{(k)}\right)\right)^{1/K}$
	$\mathrm{Gamma}(\theta, \beta)$	$\mathrm{Gamma}(\frac{1}{K}\theta + \frac{k-1}{K}, \frac{1}{K}\beta)$	$\left(\Pi_{k=1}^K X^{(k)}\right)^{1/K}$
	$\mathrm{Weibull}(\theta, \nu)$	$\mathrm{Gamma}(\frac{1}{K}, \theta^{-\nu})$	$\left(\sum_{k=1}^K X^{(k)}\right)^{1/\nu}$
	$\mathrm{Pareto}(\nu, \theta)$	$\mathrm{Gamma}(\frac{1}{K}, \theta)$	$\nu \times \mathrm{Exp}\left(\sum_{k=1}^K X^{(k)}\right)$
	$\mathrm{N}(0,\theta)$	$\mathrm{Gamma}(\frac{1}{2K}, \frac{1}{2\theta})$	$X^2 = \sum_{k=1}^K X^{(k)}$
	$\mathrm{N}_K(\theta_1 1_K, \theta_2 I_K)$	$N(\theta_1, \theta_2)$	sample mean and variance
Truncated support family	$\mathrm{Unif}(0, \theta)$	$\theta \cdot \mathrm{Beta}(\frac{1}{K}, 1)$	$\max\left(X^{(1)}, \ldots, X^{(K)}\right)$
	$\theta \cdot \mathrm{Beta}(\alpha, 1)$	$\theta \cdot \mathrm{Beta}(\frac{\alpha}{K}, 1)$	
	$\theta + \mathrm{Exp}(\lambda)$	$\theta + \mathrm{Exp}(\lambda/\mathrm{K})$	$\min\left(X^{(1)}, \ldots, X^{(K)}\right)$
Non-parametric	F^n	F^{n_k}	sort$(X^{(1)}, \ldots, X^{(K)})$

We are working on additional extensions to Project 3



Statistics > Methodology

[Submitted on 22 Mar 2023]

Generalized Data Thinning Using Sufficient Statistics

Ameer Dharamshi, Anna Neufeld, Keshav Motwani, Lucy L. Gao, Daniela Witten, Jacob Bien

Acknowledgements

Acknowledgements

Daniela Witten
University of Washington

Acknowledgements

Daniela Witten
University of Washington

Lucy Gao
University of British Columbia

Acknowledgements

Daniela Witten
University of Washington

Lucy Gao
University of British Columbia

Alexis Battle
Johns Hopkins

Joshua Popp
Johns Hopkins

Acknowledgements

Daniela Witten
University of Washington

Lucy Gao
University of British Columbia

Ameer Dharamshi
University of Washington

Alexis Battle
Johns Hopkins

Joshua Popp
Johns Hopkins

Acknowledgements

Daniela Witten
University of Washington

Lucy Gao
University of British Columbia

Ameer Dharamshi
University of Washington

Keshav Motwani
University of Washington

Alexis Battle
Johns Hopkins

Joshua Popp
Johns Hopkins

Jacob Bien
USC

Questions?