Defect Content Estimation for Inspections:

Empirical Interval Estimates

Frank Padberg

Universität Karlsruhe

Germany

Inspection Outcome

- list of detected defects
- zero-one matrix: shows which reviewer detected which defect
- classification of the defects

Our Task

reliably estimate

the number of defects in a software document

from the outcome of an inspection!

Existing Estimation Methods

- capture–recapture methods (Eick ea. ICSE 1992)
- curve–fitting methods (Wohlin ea. ICSE 1998)
- studies show that estimates are far too unreliable (Briand ea. TSE 2000, Biffl ea. ICSE 2001)

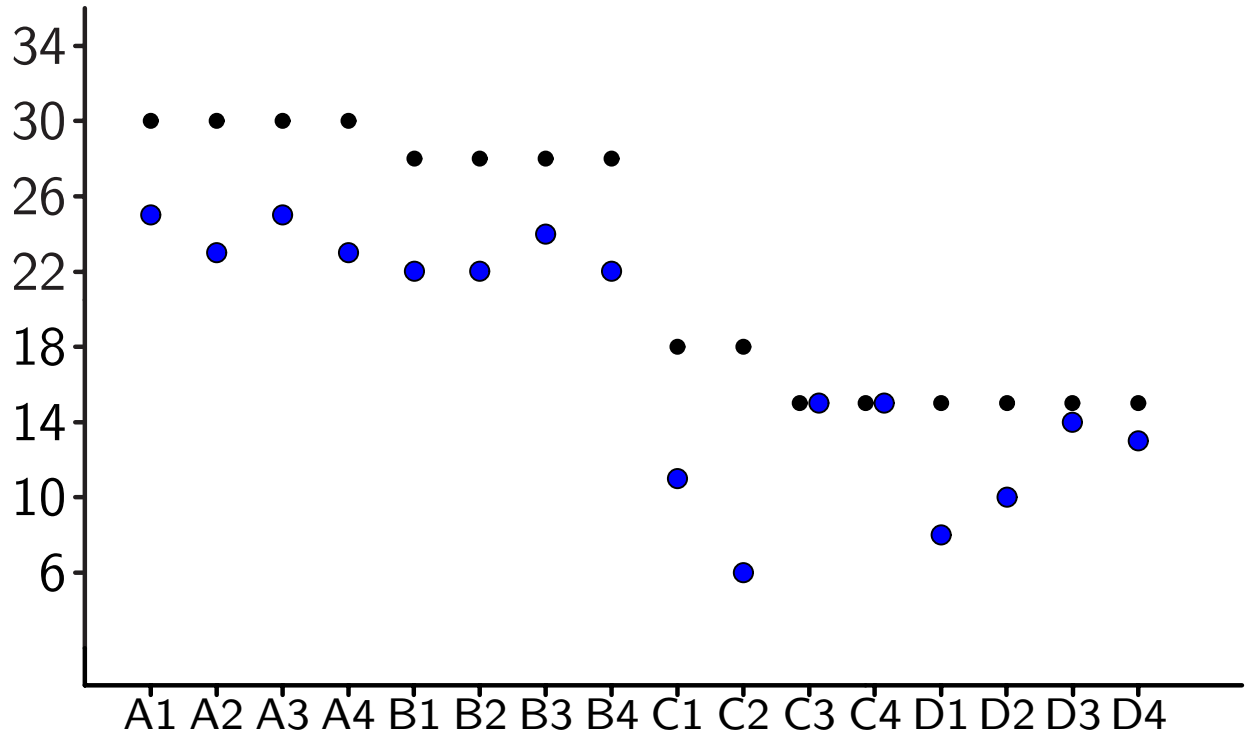
Sample Database

- 16 inspections from controlled experiments at NASA SEL (Basili e.a. 1994/1995)
- four specification documents of varying size
- between 6 and 8 reviewers
- two reading techniques
- true number of defects known exactly

Input Data for Capture–Recapture

- number w_k of defects detected by reviewer k
- total number d of different defects detected
- example: (9, 7, 6, 13, 9, 6) and $d = 23$

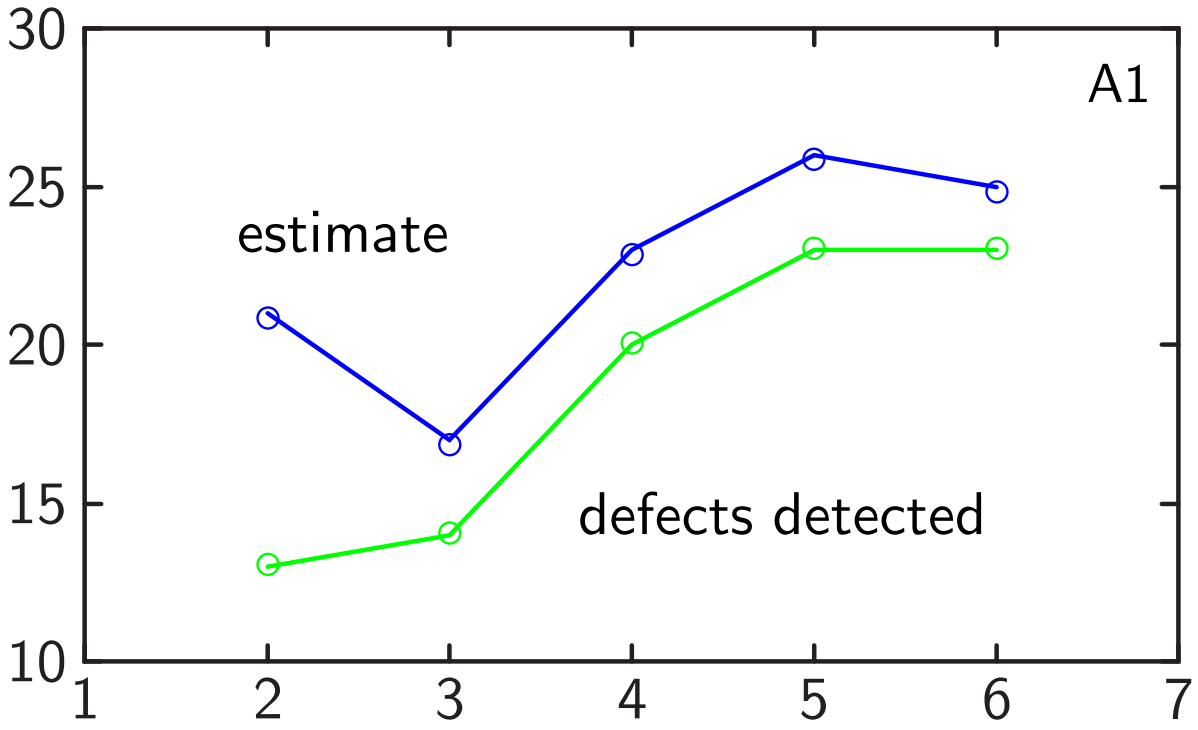
Capture–Recapture Estimates



average error of 24 percent

tendency to underestimate

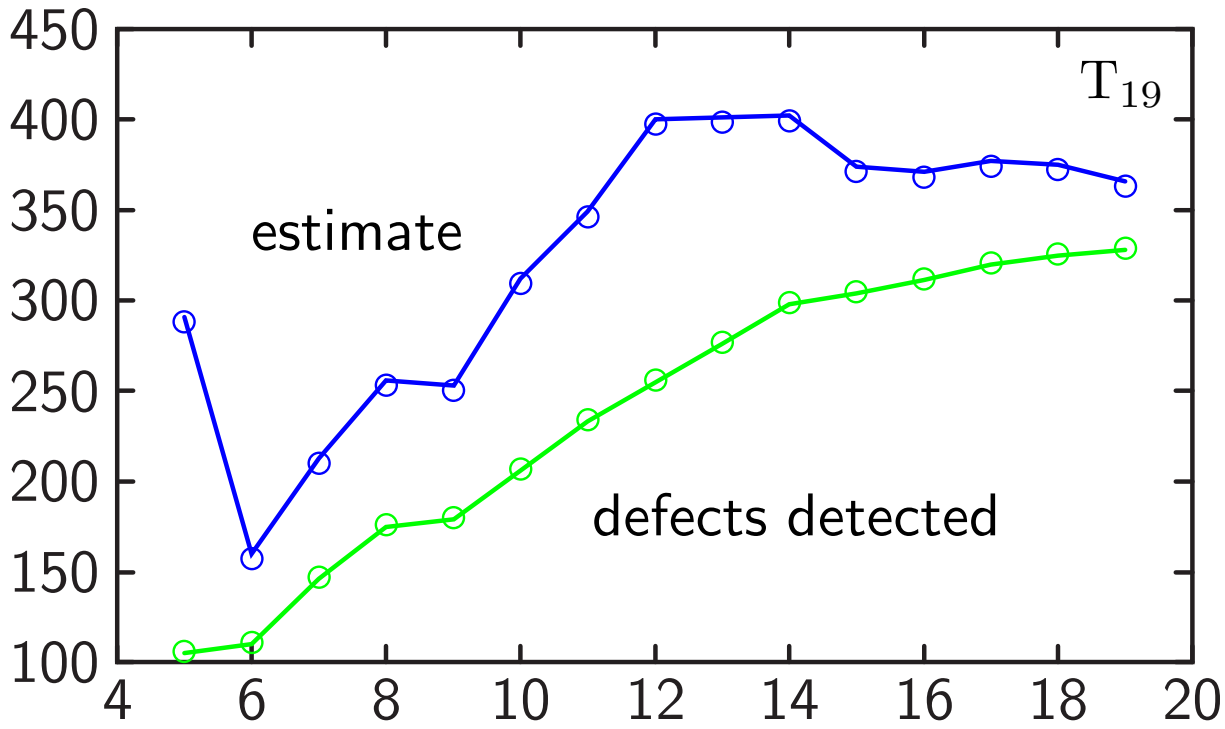
CR–Estimate versus Number of Reviewers



estimates vary with the number of reviewers

final estimate too low (25 instead of 30)

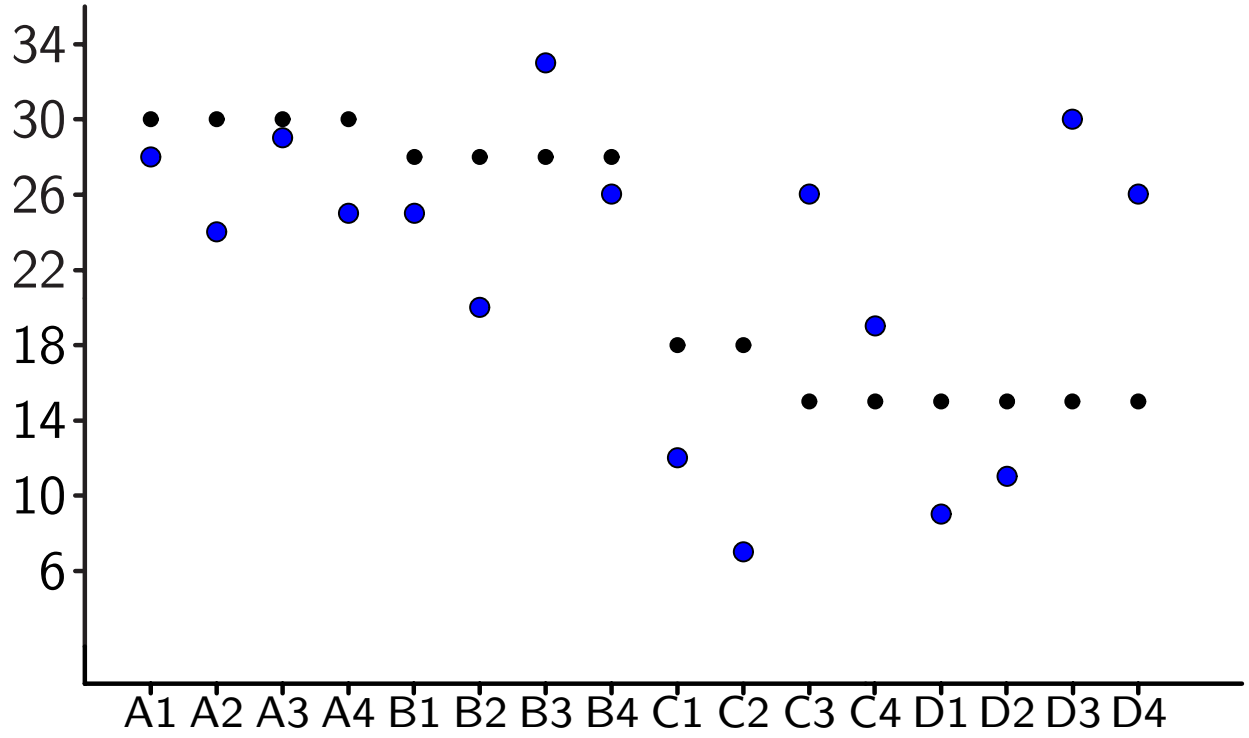
CR–Estimate versus Length of Test Series



estimate "stabilizes" for long test series

high variation of estimate over first few tests

Estimates for Detection Profile Method



average error of 36 percent

extremely high variation

Why Capture–Recapture Fails

- mathematics: "test series" is too short

Why Capture–Recapture Fails

- mathematics: "test series" is too short
- only the outcome of the current inspection enters the estimation

Why Capture–Recapture Fails

- mathematics: "test series" is too short
- only the outcome of the current inspection enters the estimation
- in other words: no learning from experience

Interval Estimate Method

- use empirical data from past inspections for estimating, besides the outcome of the current inspection

Interval Estimate Method

- use empirical data from past inspections for estimating, besides the outcome of the current inspection

- construct a stochastic model for the outcome of an inspection from the empirical data

Interval Estimate Method

- use empirical data from past inspections for estimating, besides the outcome of the current inspection
- construct a stochastic model for the outcome of an inspection from the empirical data
- maximum likelihood estimation of the defect content of the currently inspected document

Empirical Data About Past Inspections

- number w_k of defects detected by reviewer k
- total number d of different defects detected
- true number N of defects $(N = 30)$

Stochastic Modeling

- relate inspection outcome (the w_k and d) to the true number N of defects

- bundle up datapoints with an equivalence relation ("signature") to avoid isolated points

Signature of an Inspection

- signature $=$ (efficiency class, span)
- the efficiency class is a measure for the overall efficiency of the inspection
- the span is a measure for the variation among the reviewers' inspection results
- by construction, the signature depends on the number N of defects in the document

Efficiency Class of an Inspection

- compute overall detection ratio $r = \frac{d}{N}$
- subdivide range of $0 \ldots 100$ percent into classes
- determine efficiency class $c = \text{class}(r)$
- example: subdivision in steps of 20 percent

$$\text{yields } \text{class}\left(\frac{23}{30}\right) = 4$$

Span of an Inspection

- compute individual detection ratios $r_k = \frac{w_k}{N}$
- subdivide range of $0 \ldots 100$ percent into classes
- determine detection ratio classes $c_k = \text{class}(r_k)$
- compute span $s = \max c_k - \min c_k + 1$

Span Computation Example

- $N = 30$, inspection result $(9, 7, 6, 13, 9, 6)$
- detection ratios $(\frac{9}{30}, \frac{7}{30}, \frac{6}{30}, \frac{13}{30}, \frac{9}{30}, \frac{6}{30})$
- subdivision in steps of 10 percent
- detection ratio classes $(3, 3, 2, 5, 3, 2)$
- span $= 5 - 2 + 1 = 4$

Pre-Processed Sample Database

	c	s
A1	4	4
A2	4	4
A3	4	3
A4	4	4

	c	s
B1	4	3
B2	4	3
B3	5	4
B4	4	5

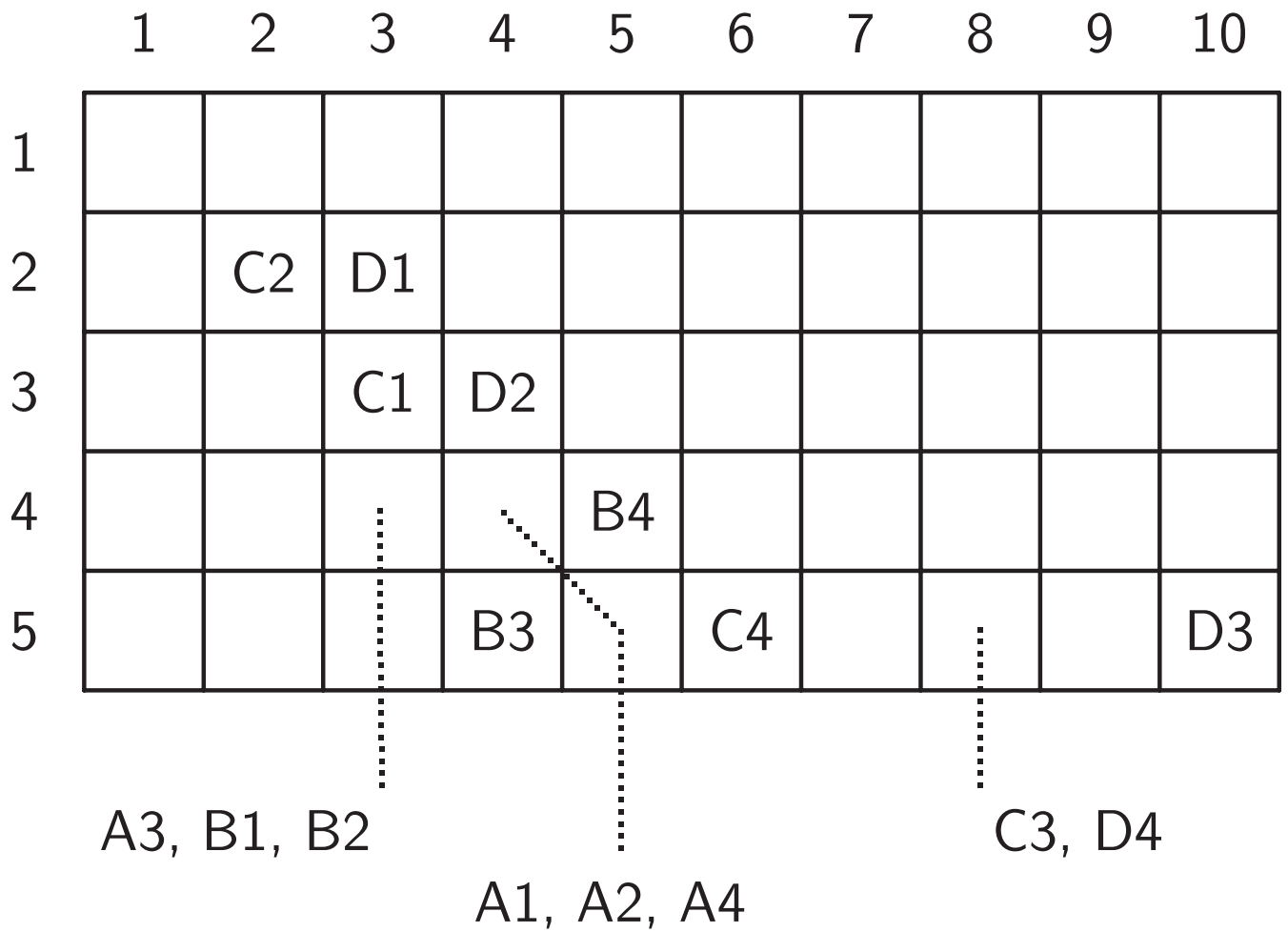
	c	s
C1	3	3
C2	2	2
C3	5	8
C4	5	6

	c	s
D1	2	3
D2	3	4
D3	5	10
D4	5	8

subdivision in steps of 20 percent for the efficiency class

subdivision in steps of 10 percent for the span

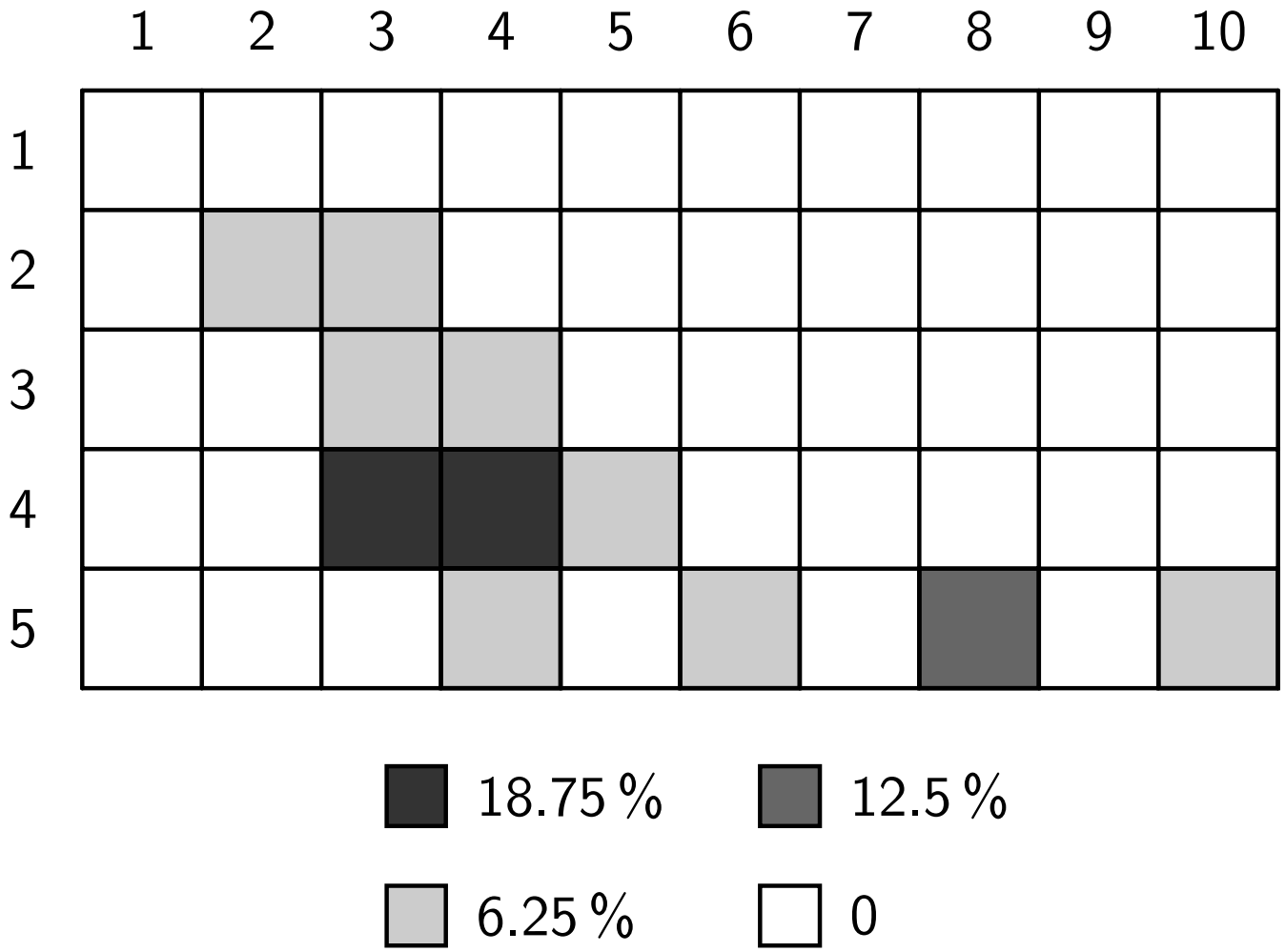
Pre-Processed Sample Database



Constructing the Stochastic Model

- compute the signature for each inspection in the database
- compute the relative frequency of each signature
- assign to each signature its relative frequency as its probability

Full Sample Probability Distribution



Likelihood Function

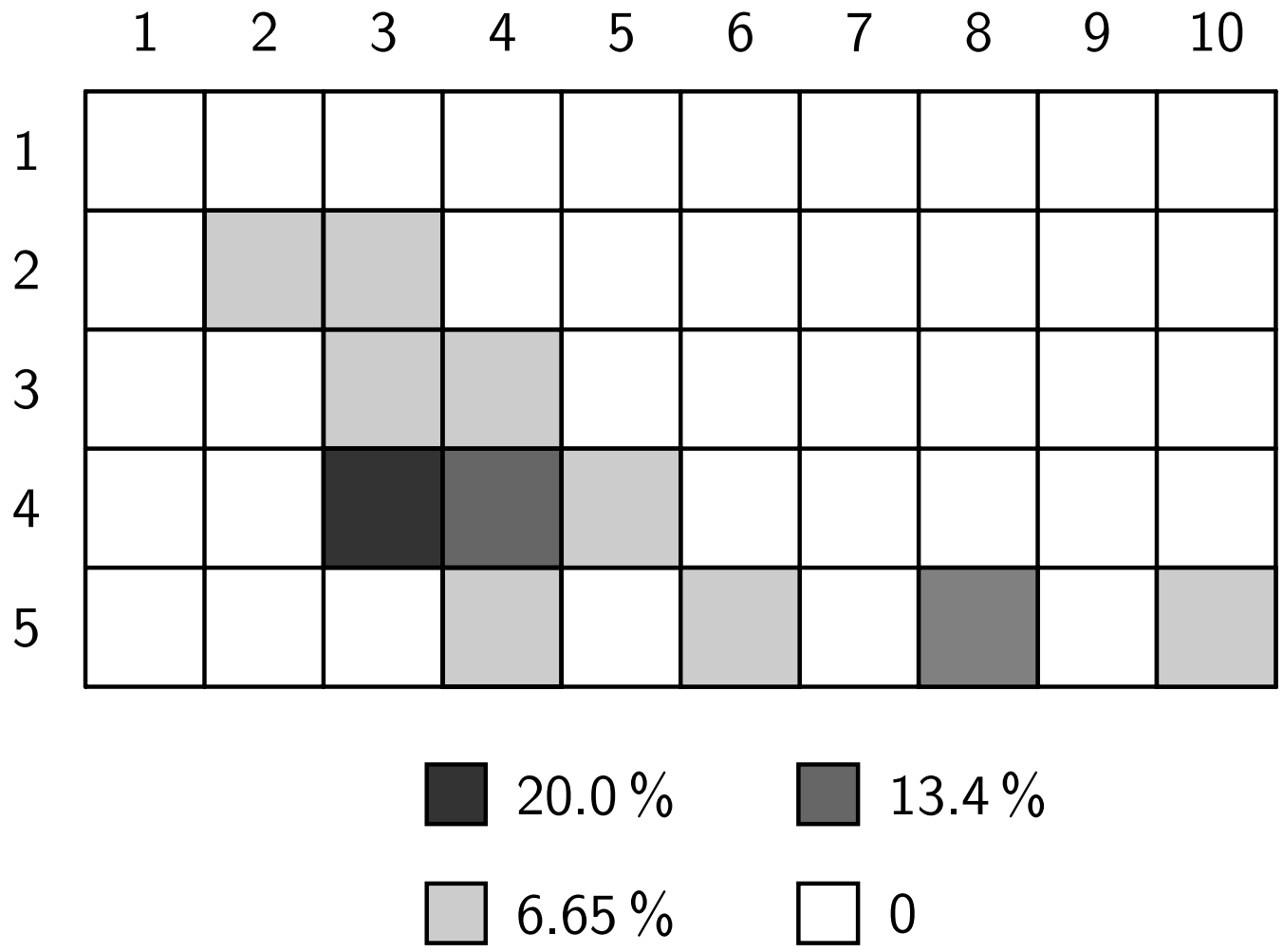
- have result vector for current inspection, but do *not* know the value of N
- signature of the inspection depends on N
- compute signature for all possible values of N
- get the likelihood function

$$\mathrm{L}: \; N \; \mapsto \; \mathrm{P}\,(\,c\,(\,N\,),\, s\,(\,N\,)\,)$$

Example

- assume result vector (9, 7, 6, 13, 9, 6; 23)
- forget about known number of 30 defects
- use inspections A2 through D4 as empirical database
- re-compute probability distribution

Probability Distribution with A1 Left Out



Example's Likelihood Function

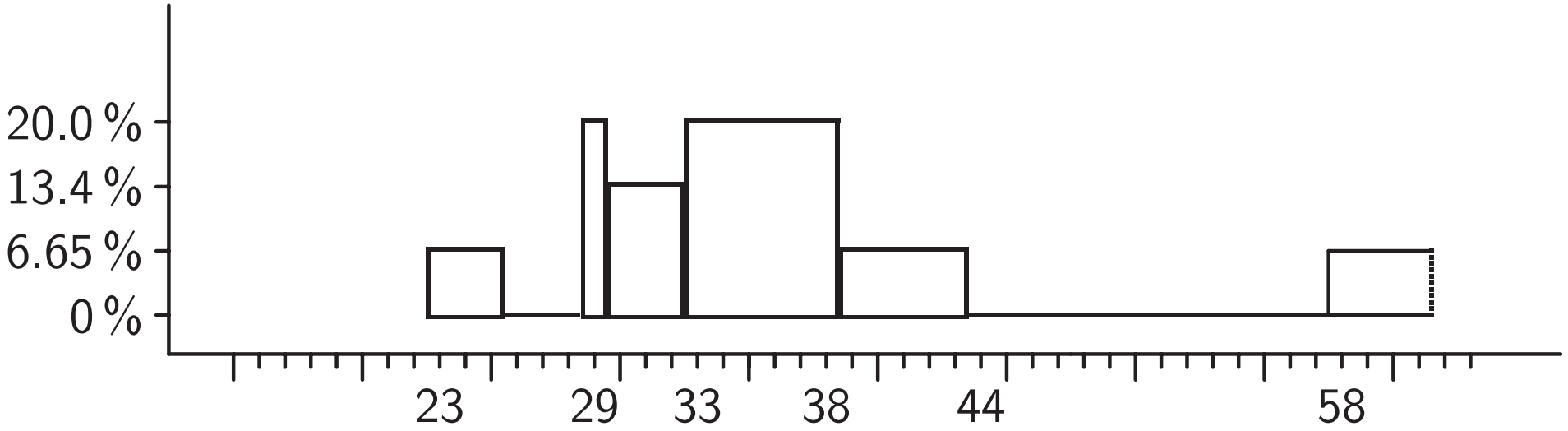
N	$c\,(N)$	$s\,(N)$	$\mathrm{L}\,(N)$
23 – 25	5	4	6.65 %
26 – 28	5	3	0
29	4	3	20.0 %
30 – 32	4	4	13.4 %
33 – 38	4	3	20.0 %
39 – 43	3	3	6.65 %

N	$c\,(N)$	$s\,(N)$	$\mathrm{L}\,(N)$
44 – 57	3	2	0
58 – 59	2	2	6.65 %
60 – 64	2	3	6.65 %
65 – 114	2	2	6.65 %
115 – 129	1	2	0
130 – ...	1	1	0

A1 left out from the database

probability distribution re-computed

Example's Likelihood Function



A1 left out from the database

probability distribution re-computed

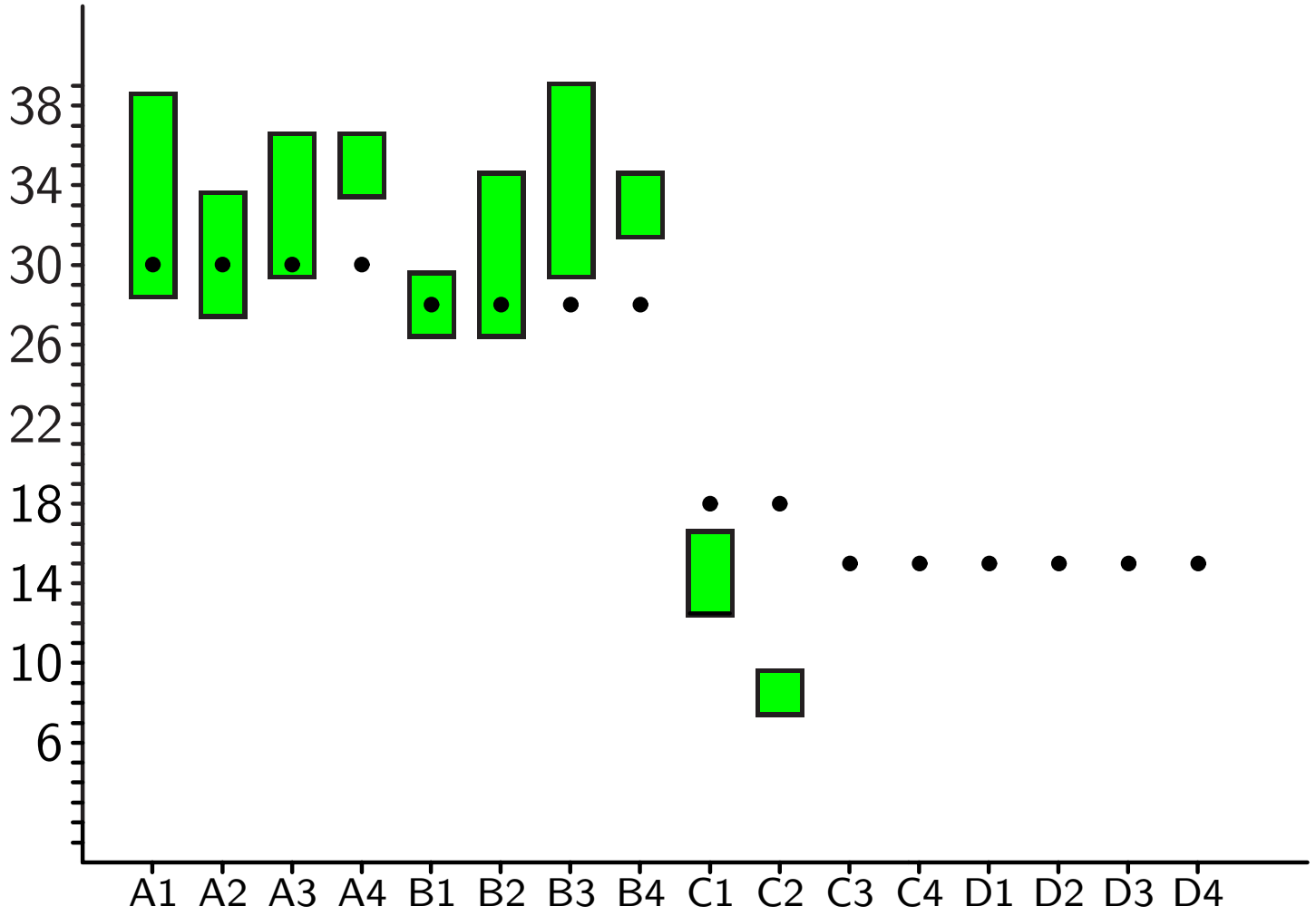
Interval Estimates

- likelihood function assigns to each N the probability of the corresponding signature
- determine values of N where the likelihood is maximal
- get whole intervals as estimates
- previous example: N is most likely to range between 29 and 38 (true value: 30)

Jackknife Validation

- leave out an inspection from the database
- compute the probability measure using the remaining 15 inspections
- compute the interval estimate for the one inspection which was left out
- compare the estimate with the true value of the number of defects

Jackknife Validation Results



reasonable interval estimates on one half of the dataset

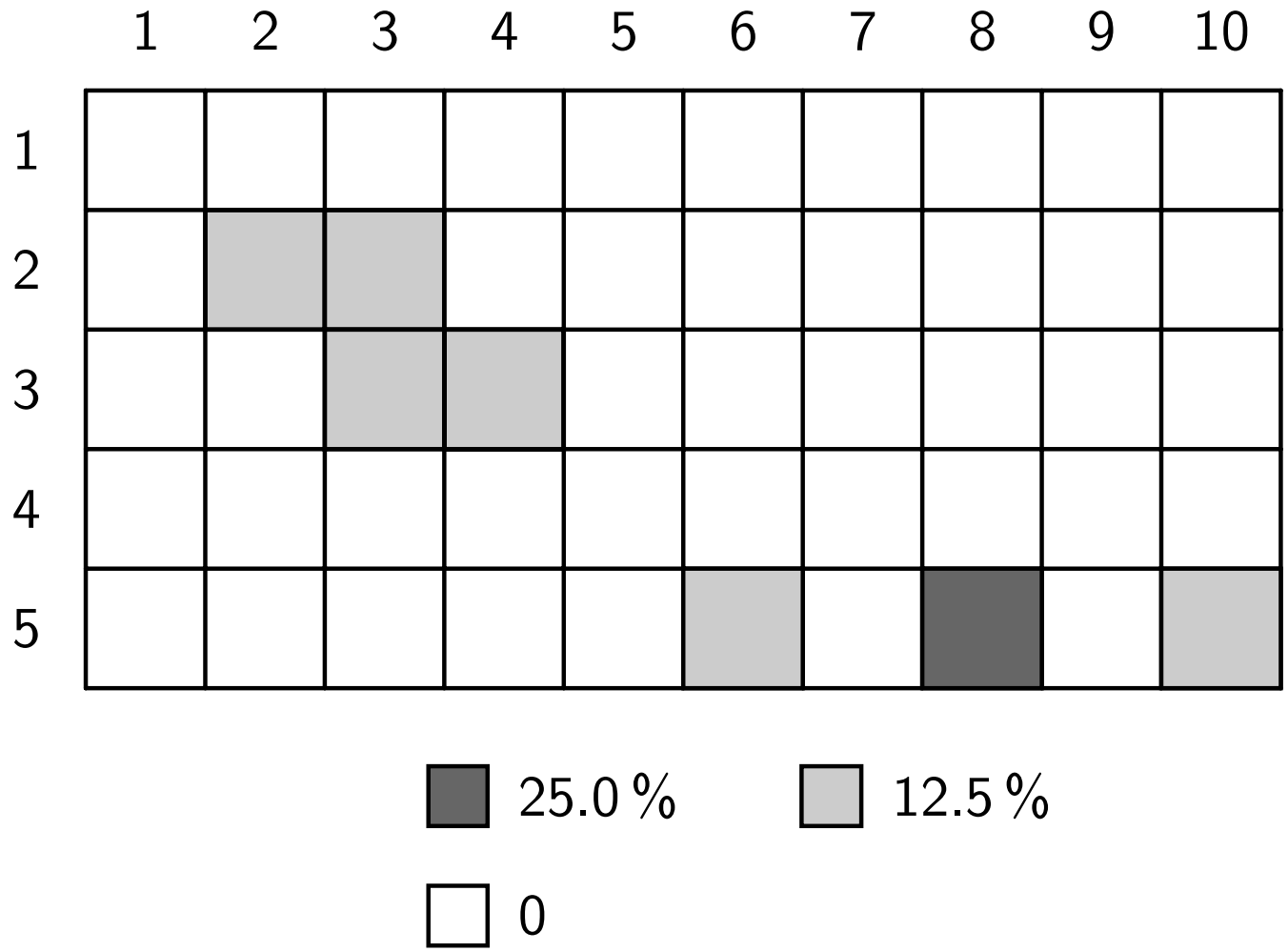
Confidence Levels

- no estimates for C3 to D4 specified
- value of the likelihood function provides a simple confidence level
- discard an estimate if its confidence level is too low (graph of likelihood function is flat)
- discard estimates for C3 to D4

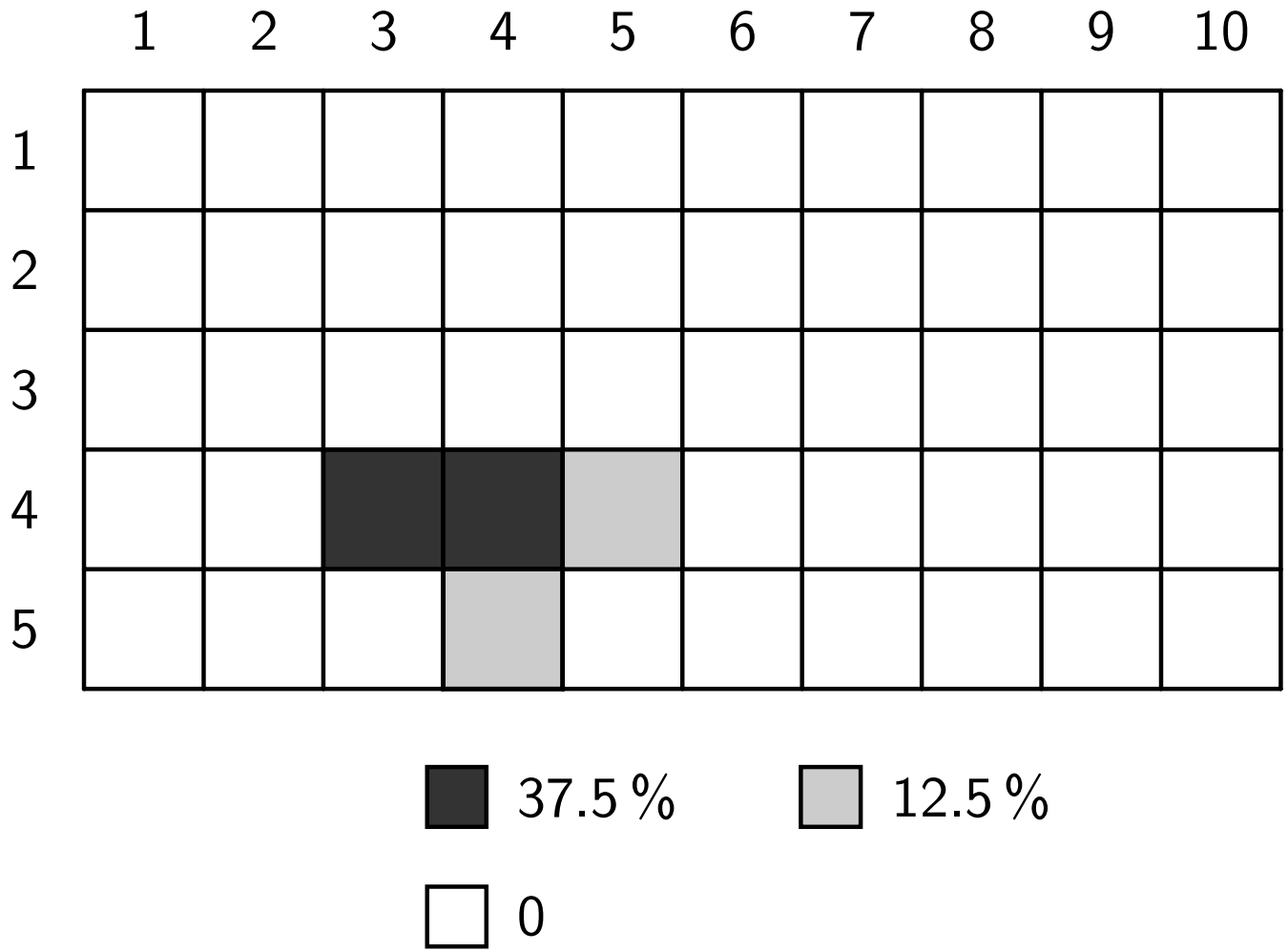
Domain Dependence

- interval estimates for C1 and C2 are outliers
- C1 and C2 belong to different document domain than A1 to B4
- split dataset according to document domain
- re-compute stochastic model on each domain
- outliers vanish, other estimates don't change

Probability Distribution for NASA Domain



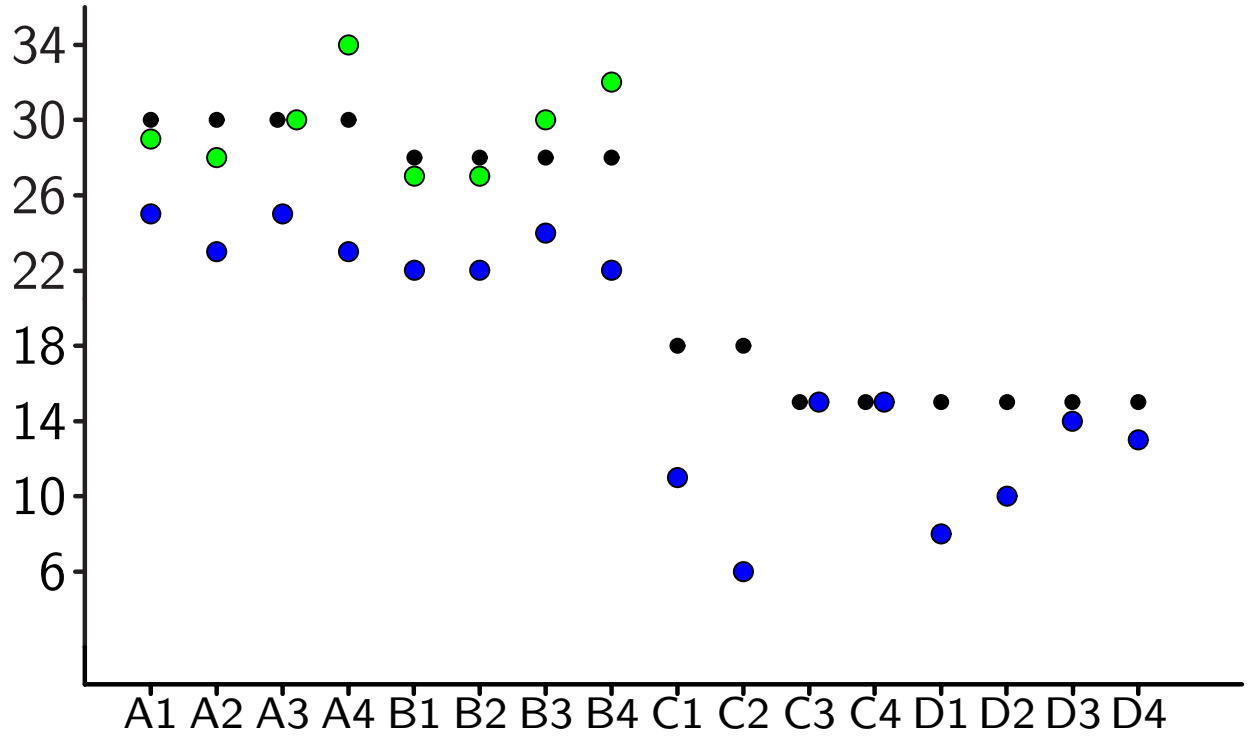
Probability Distribution for Generic Domain



Point Estimates

- derive from interval estimate
- good candidates are lower boundary and median
- previous example:
  lower boundary 29, median 34, true value 30

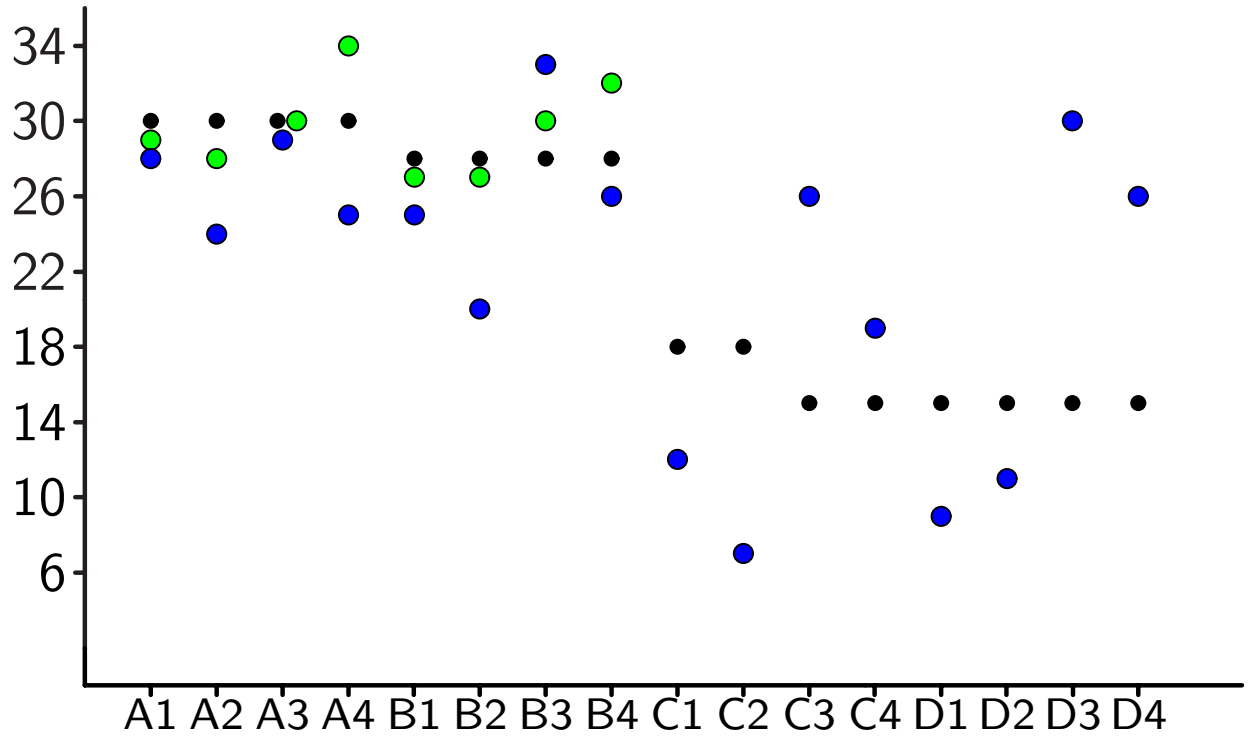
Interval Estimates versus Capture–Recapture



lower boundary as point estimate

clearly outperforms capture–recapture

Interval Estimates versus Curve-Fitting



lower boundary as point estimate

clearly outperforms detection profile method

Estimation Errors

	CRM	DPM	IEM
A1	−20.0 %	−6.7 %	−3.4 %
A2	−26.7 %	−20.0 %	−6.7 %
A3	−16.7 %	−3.4 %	0 %
A4	−23.4 %	−16.7 %	+13.4 %
B1	−21.5 %	−10.8 %	−3.6 %
B2	−25.0 %	−28.6 %	−3.6 %
B3	−14.3 %	+17.9 %	+7.2 %
B4	−25.0 %	−7.2 %	+14.3 %
mean abs	**21.6 %**	**13.9 %**	**6.6 %**

IEM Summary

- uses empirical data from past inspections
- stochastic model and max likelihood estimation
- interval estimates and confidence levels
- outperforms existing methods
- see Padberg ICSE 2002

Required Inspection Data

- zero-one matrix
- document meta-data:

  type, size, module coupling, code complexity, ...
- inspection meta-data:

  reading technique, number of reviewers, ...
- true number of defects

Building a Database

- collect data from as many inspections as possible (inspection outcome and meta-data)
- trace defects which are detected in later phases back to the corresponding document

Validating the Technique

- compute signature for each inspection
- perform a jackknife
- try different subdivisions of the database
- jackknife again on the subsets
- hopefully : reliable estimates

Let's Do It !