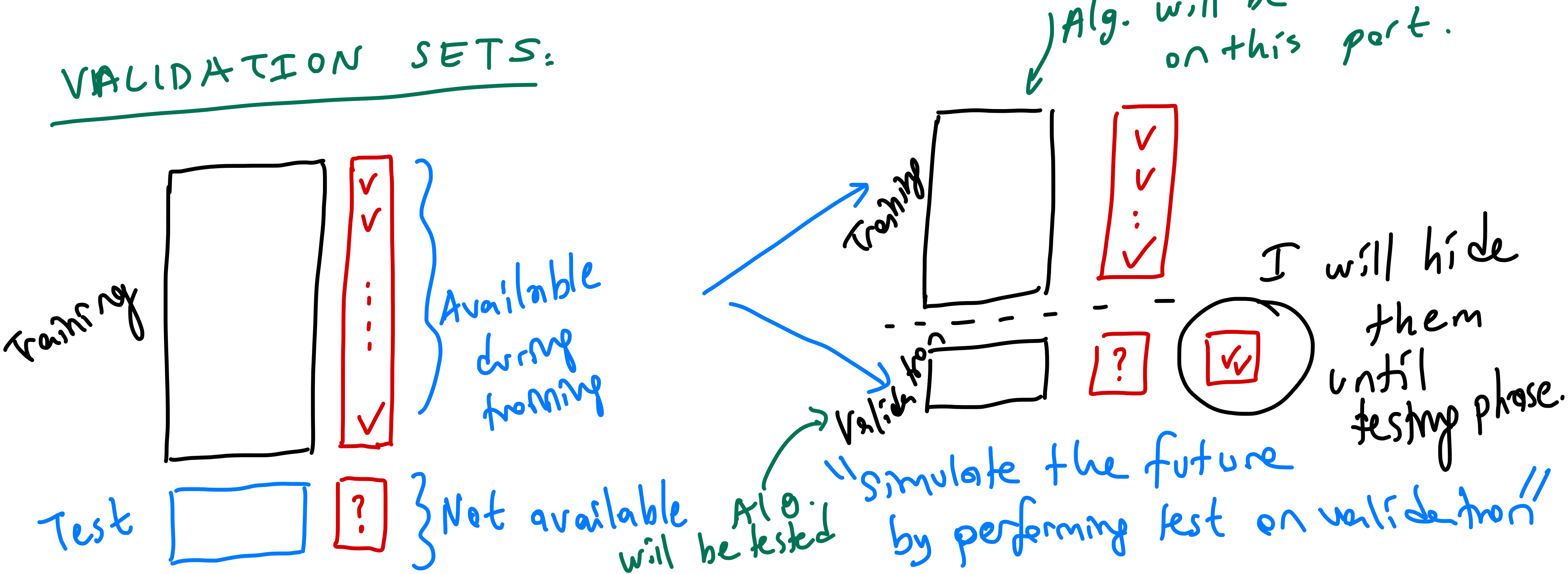
Design and Analysis of Machine Learning Experiments

1. How can we assess the expected error of a learning algorithm for a given problem?
2. Given two or more algorithms, how can we say that one is better than the other(s) for a given problem?

WE CAN NOT USE THE TRAINING SET TO ANSWER 1 & 2!

training set error < test set error

VALIDATION SETS:



"Simulate the future by performing test on validation"

$$\begin{pmatrix} A1 \\ A2 \\ A3 \end{pmatrix} \longrightarrow \begin{array}{c} R_1 \\ R_2 \\ R_3 \\ \vdots \\ R_R \end{array} \begin{pmatrix} T1 \\ T2 \\ T3 \\ \vdots \\ T_R \end{pmatrix} \begin{pmatrix} V1 \\ V2 \\ V3 \\ \vdots \\ V_R \end{pmatrix}$$

R = # of replications

$e_{1,1}$ = misclassification error of A1 on R1 :

$e_{3,2}$ = misclassification error of A3 on R_2

Measures

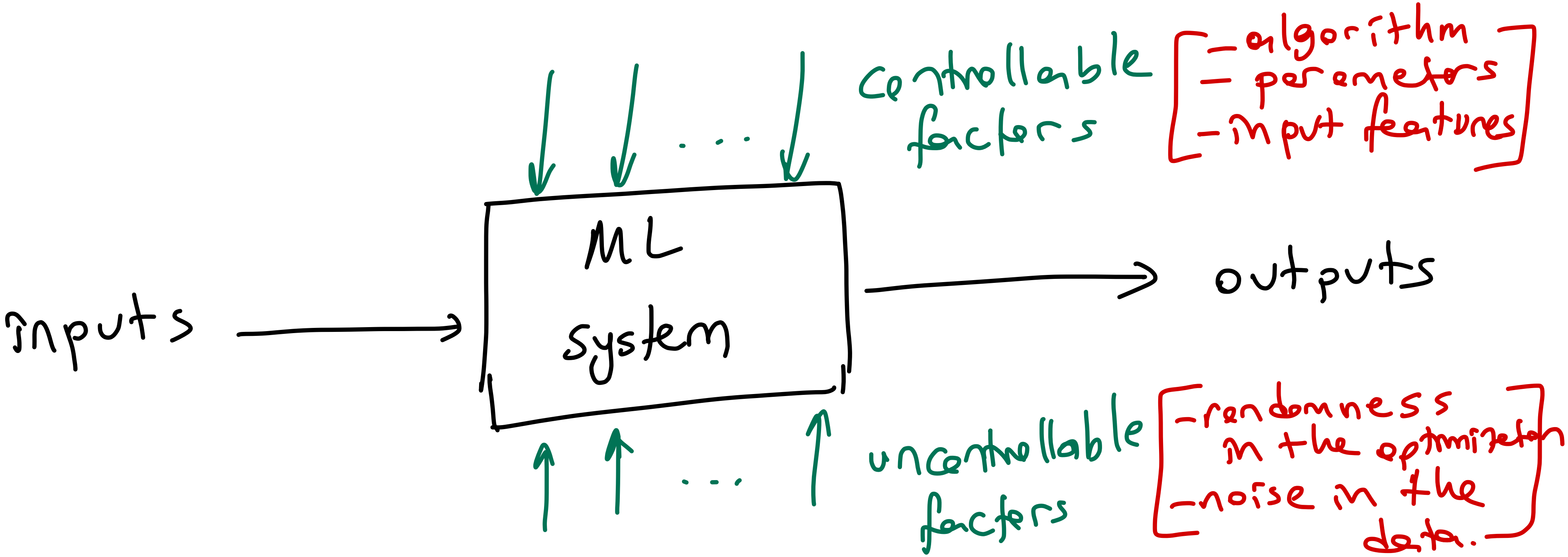
- time complexity
- space complexity
- interpretability
- easy programmability
- loss functions
  - ↳ minimizing FPs
  - ↳ minimizing FNs

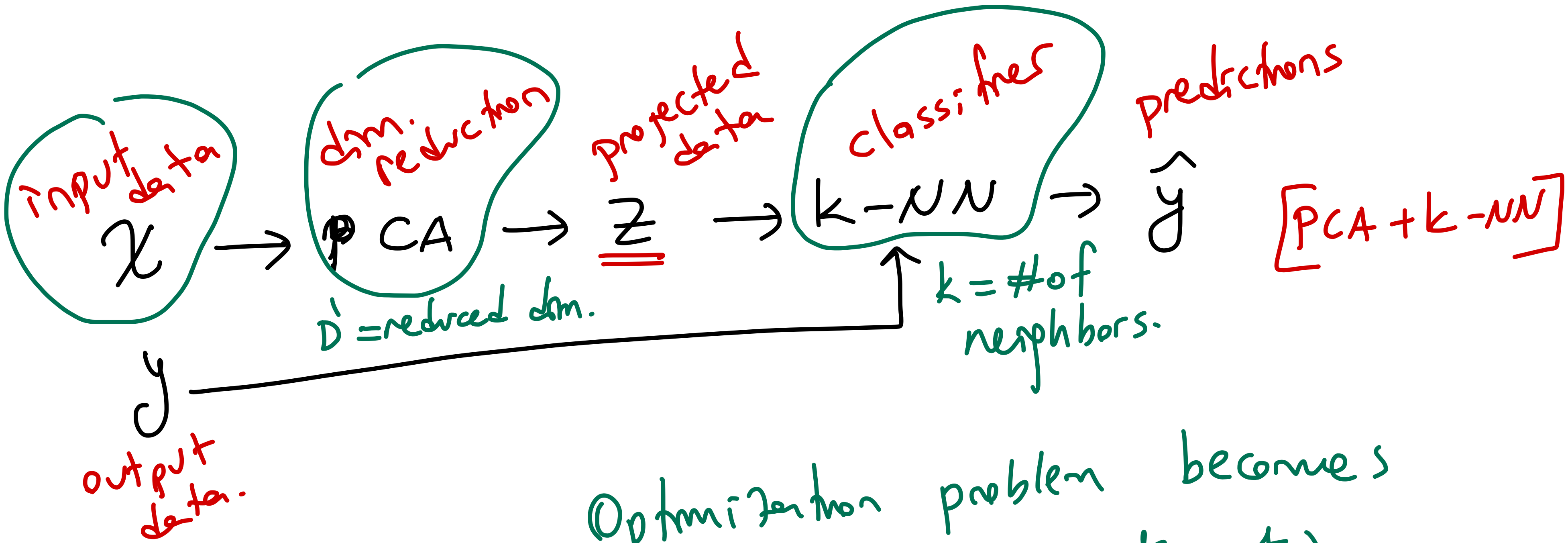
	A1	A2	A3
R_1	$e_{1,1}$	$e_{2,1}$	$e_{3,1}$
R_2	$e_{1,2}$	$e_{2,2}$	$e_{3,2}$
	$\vdots$	$\vdots$	$\vdots$
R_R	$e_{1,R}$	$e_{2,R}$	$e_{3,R}$
	e_1	e_2	e_3

e_1 = average performance of A 1.

A^* = algorithm with the minimum average error.

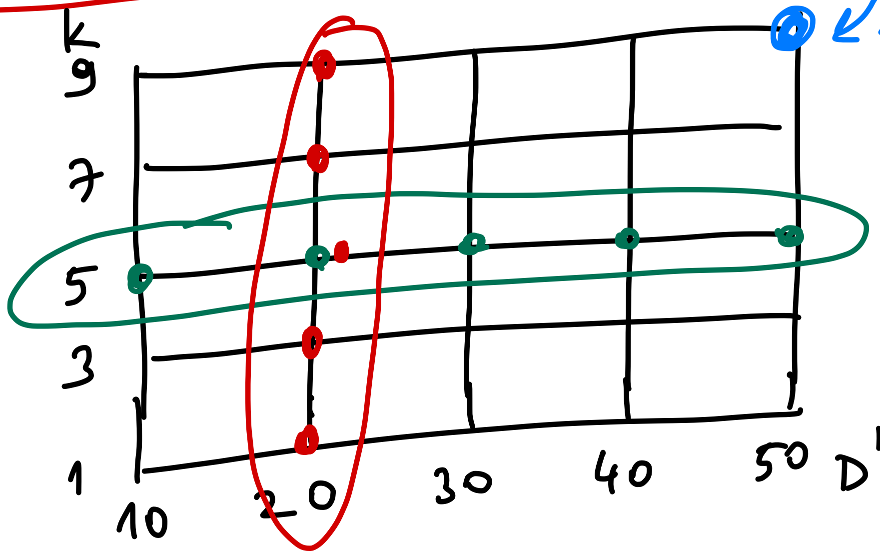
↳ if e_2 is the minimum, A_2 is the best algorithm,





Optimization problem becomes finding (D'^{*}, k^{*})

One factor at a time:



assume this is the best position.

① Find best D' by setting k to a specified value.

Assume k=5 ⇒ D' = ?

k=9

A1	A2	A3	A4	A5
(10,5)	$\boxed{(20,5)}$ ★	(30,5)	(40,5)	(50,5)

② Find best k by using D' from step #1.

Assume D'=20 ⇒ k = ?

A1	A2	A3	A4	A5
(20,1)	(20,3)	(20,5)	$\boxed{(20,7)}$ ★	(20,9)

$$\boxed{(D'^{*}, k^{*}) = (20, 7)}$$

25 possible positions

⇓

I tried only 9 out of 25 positions.

Exhaustive Enumeration

Try all possible positions

not possible due to "computational complexity"

Guidelines for ML Experiments

1. Aim of the study
   - → evaluate a single algorithm.
   - → pick the best algorithm for a specific problem.
   - → pick the best algorithm for a set of problems.

2. Selection of the response variable → performance criteria

3. Choice of factors and their levels
   - → algorithms
   - → parameters
   - → ....

4. Choice of experimental design
   - → factorial design
   - → one factor at a time
   - → response surface design
   - → ....

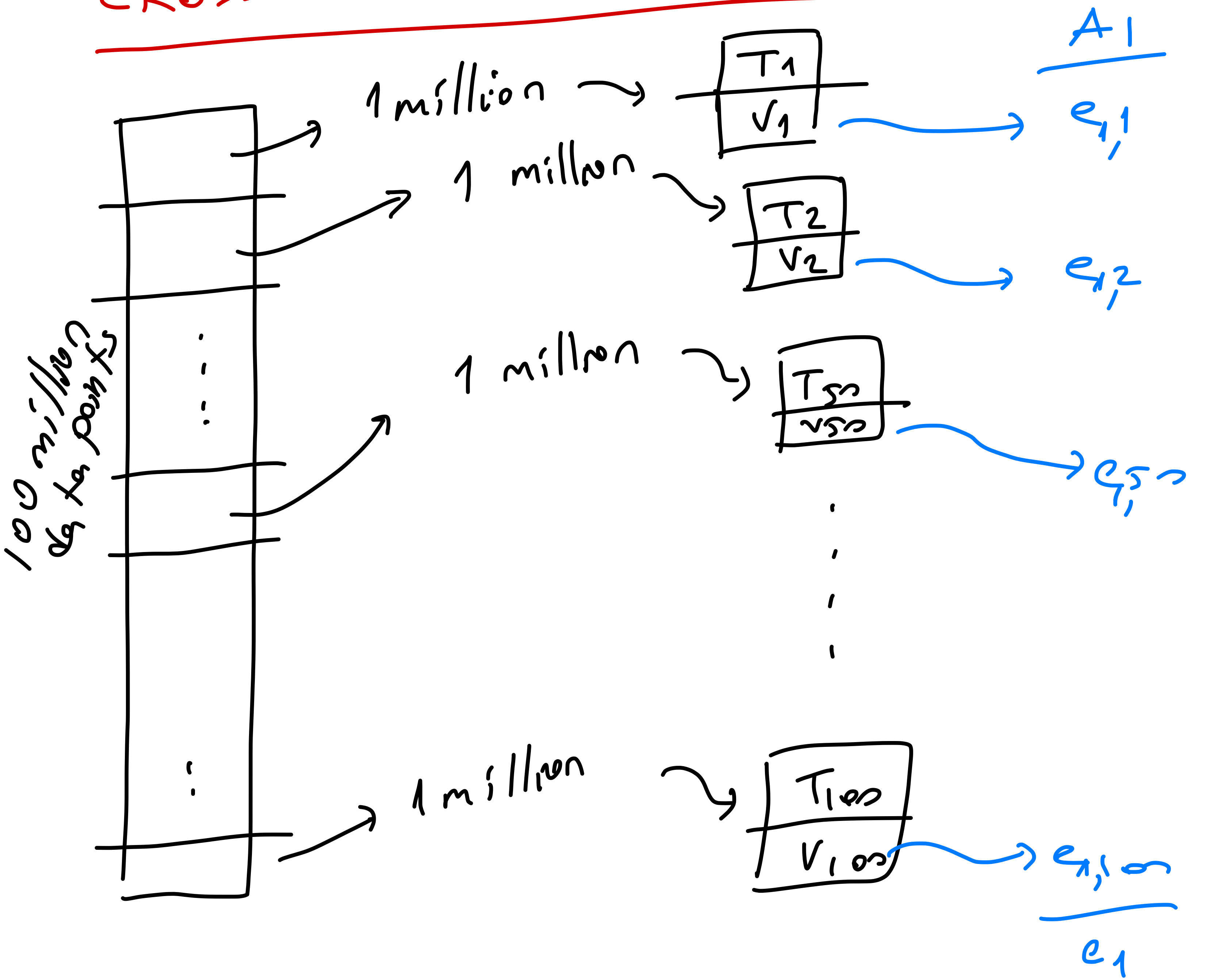
5. Run experiments → use [parallel / cloud] computing if possible.

6. Statistical analysis of the results → $AlgA_1 \overset{?}{>} AlgA_2$

   hypothesis testing.

7. Conclusions & Recommendations.

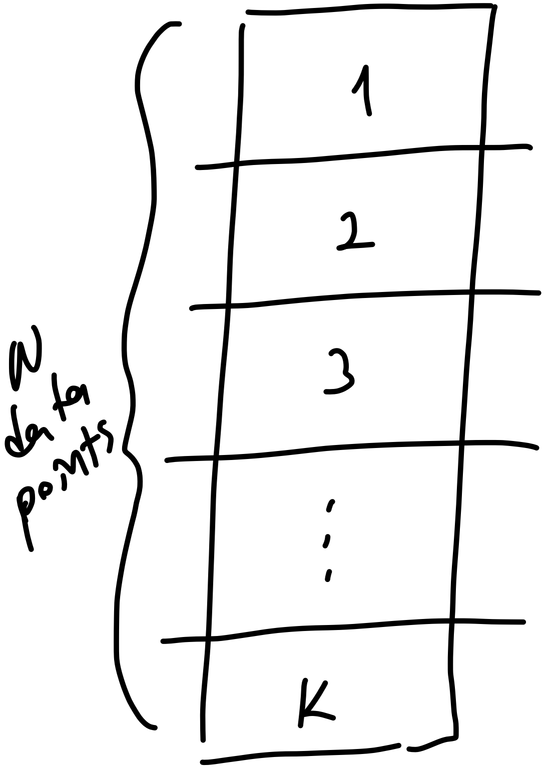
CROSS-VALIDATION & RESAMPLING

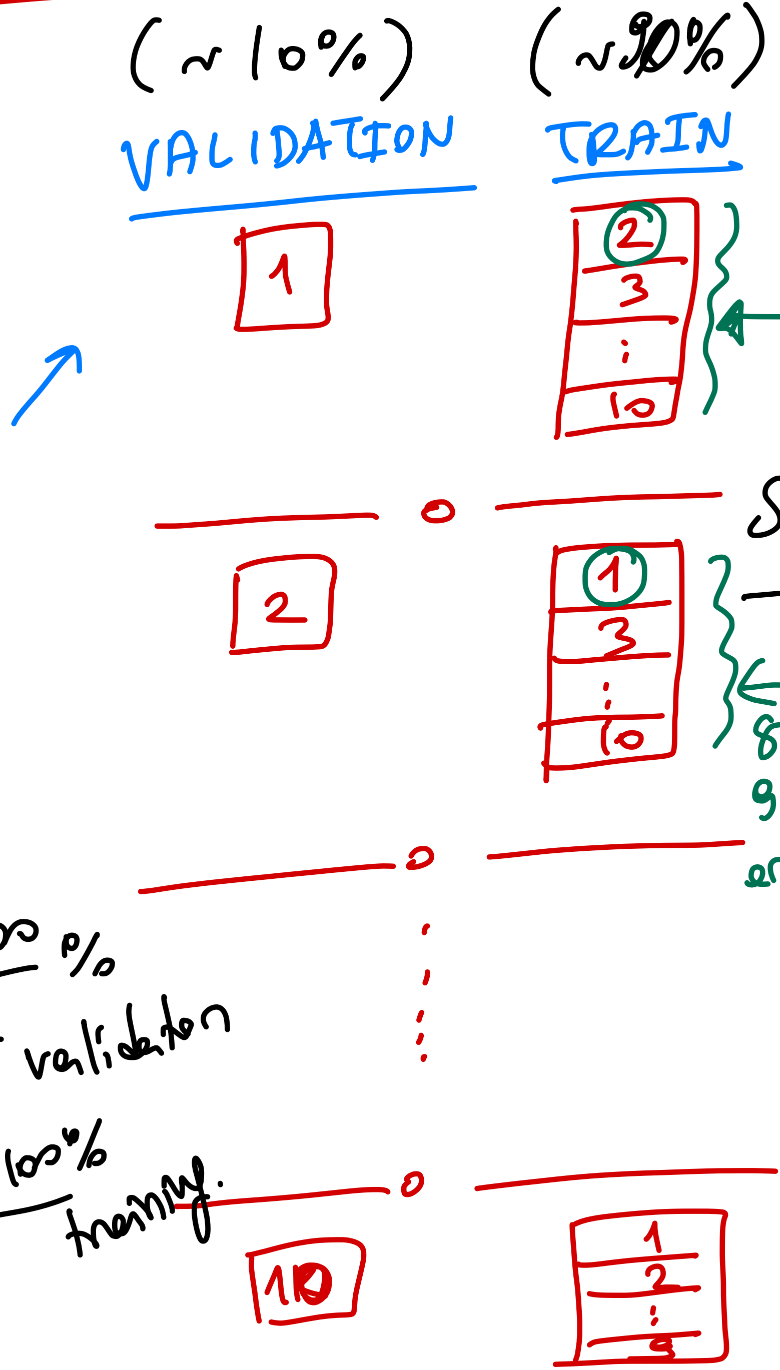


K-FOLD CROSS-VALIDATION:

Step 1: Shuffle the data points

Step 2: Divide the data set into K (almost) equal size portions.





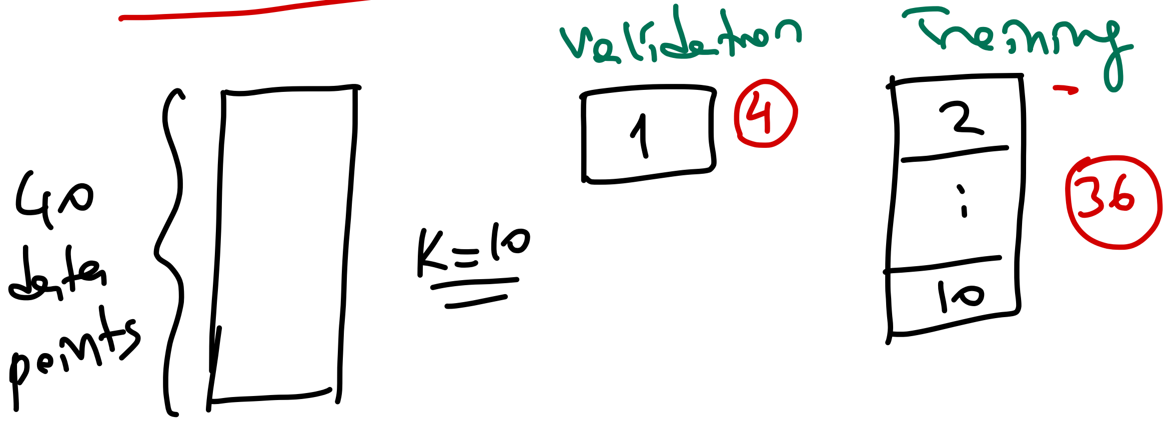
Step 3: Use each portion as the validation data set in one replication.

overlap between training data is quite large!!

Approximately $\frac{100}{K}$ % validation

$\frac{(K-1)100}{K}$ % training.

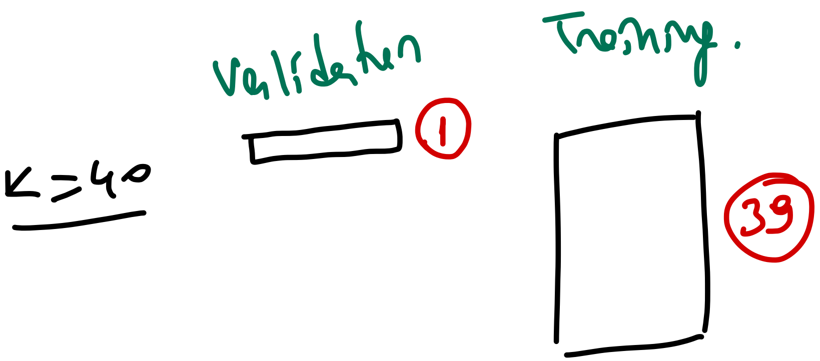
LEAVE-ONE-OUT CROSS-VALIDATION (where K=N)



each replication:

4 validation points

36 training points

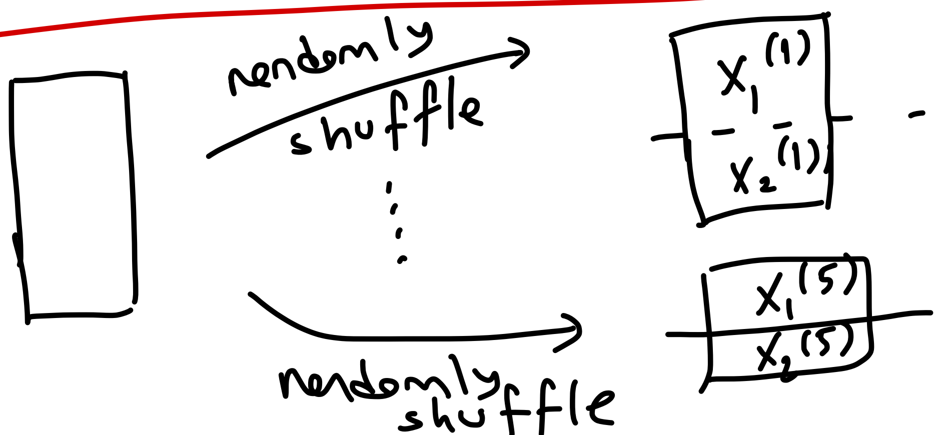


each replication:

1 validation point

39 training points

5x2 CROSS-VALIDATION



Training	Validation
$X_1^{(1)}$	$X_2^{(1)}$
$X_2^{(1)}$	$X_1^{(1)}$
⋮	⋮
$X_1^{(5)}$	$X_2^{(5)}$
$X_2^{(5)}$	$X_1^{(5)}$

10 replications.

No overlap

there is an overlap

No overlap