COMP2261 ARTIFICIAL INTELLIGENCE / MACHINE LEARNING

Cross-Validation

Dr SHI Lei

Durham University

learning lab

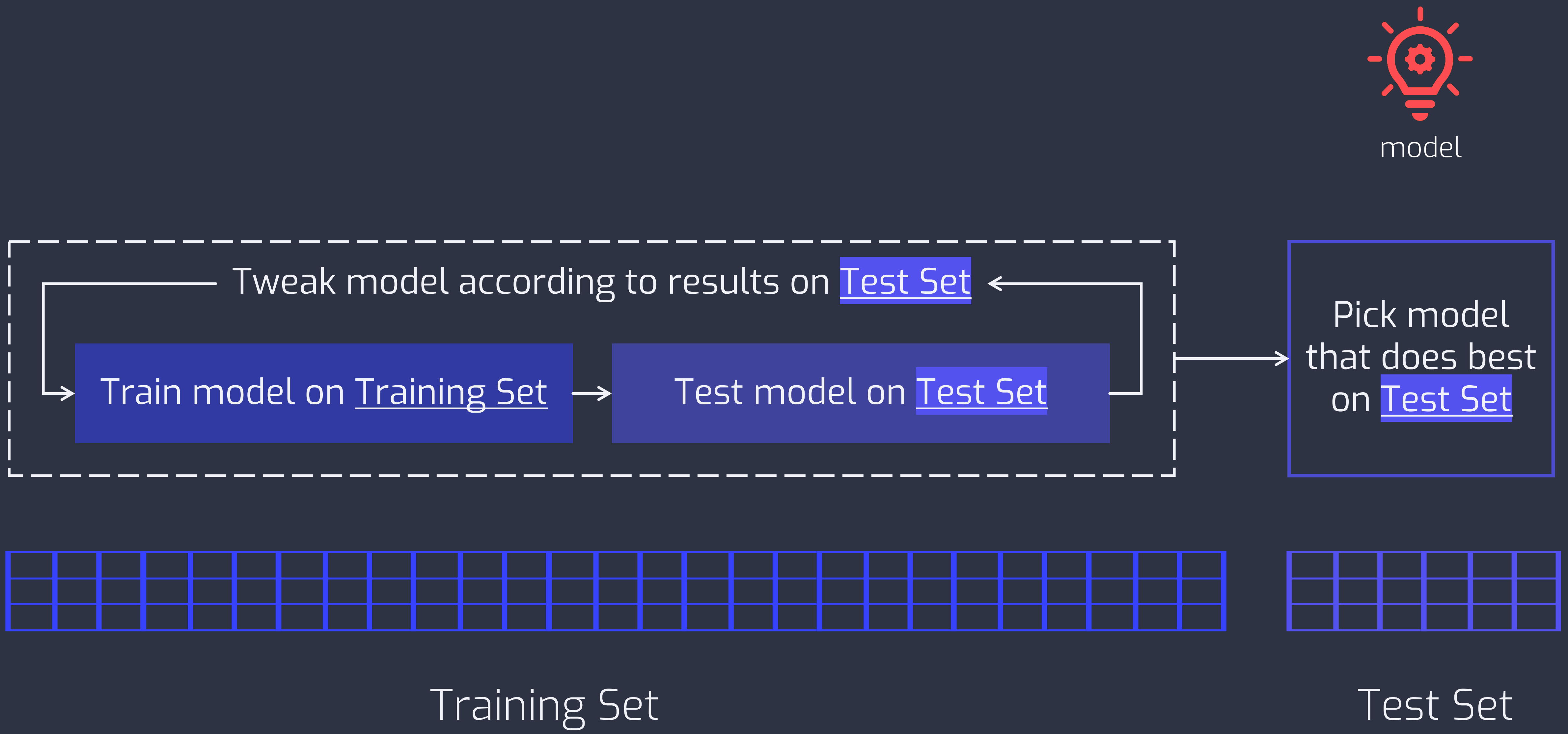
Learning Objectives

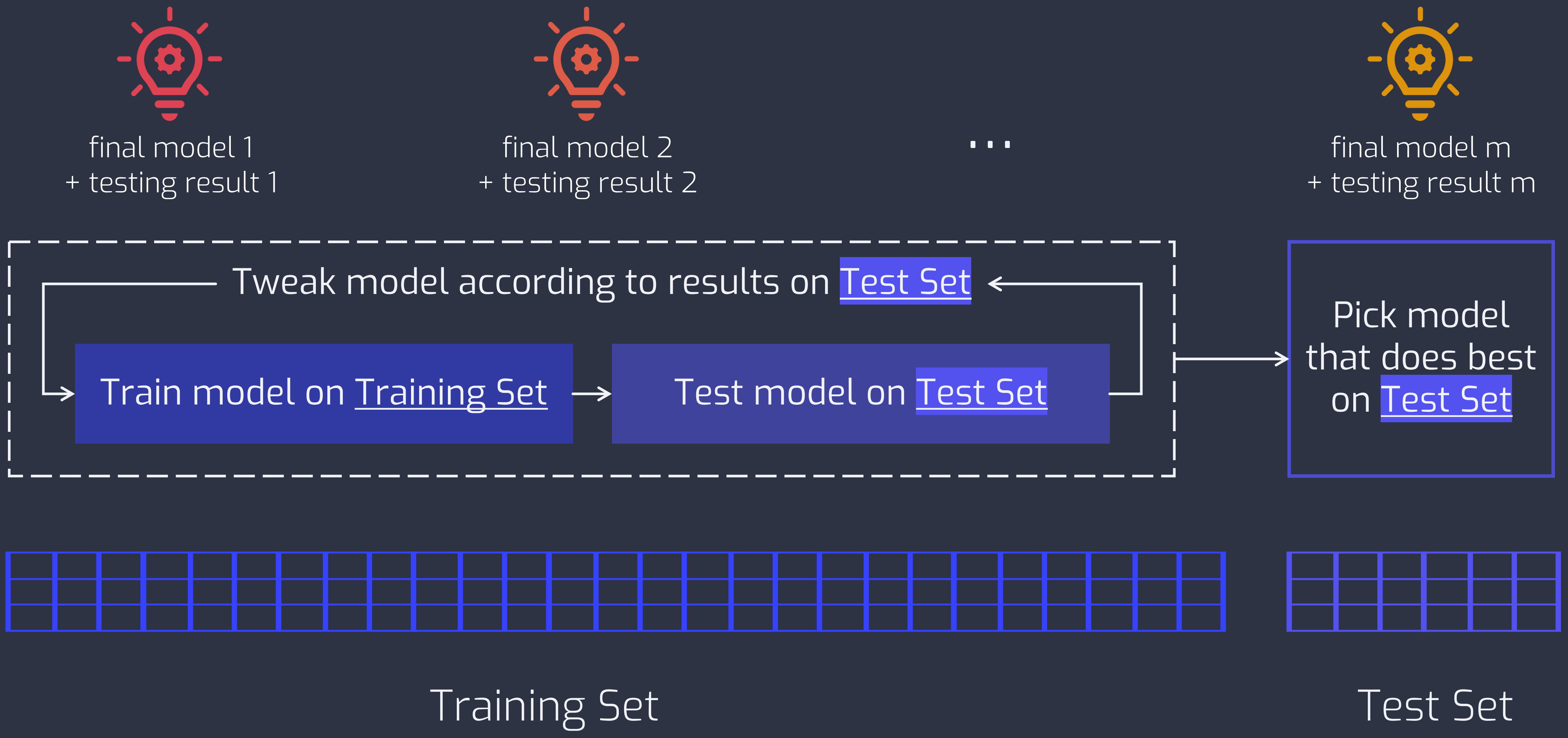
- Understand what is Cross-Validation
- Understand why we need Cross-Validation
- Understand how we use Cross-Validation

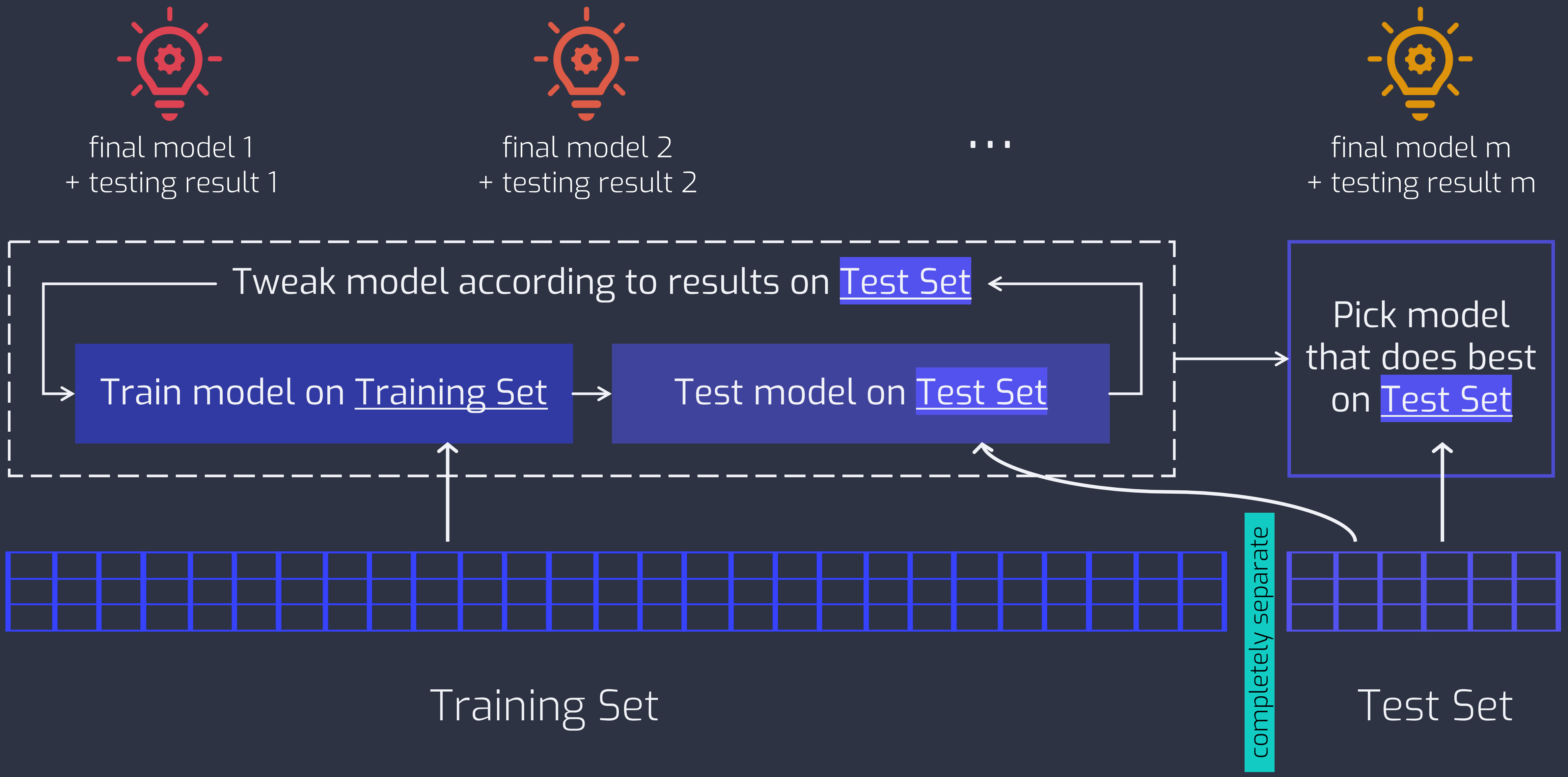
Data

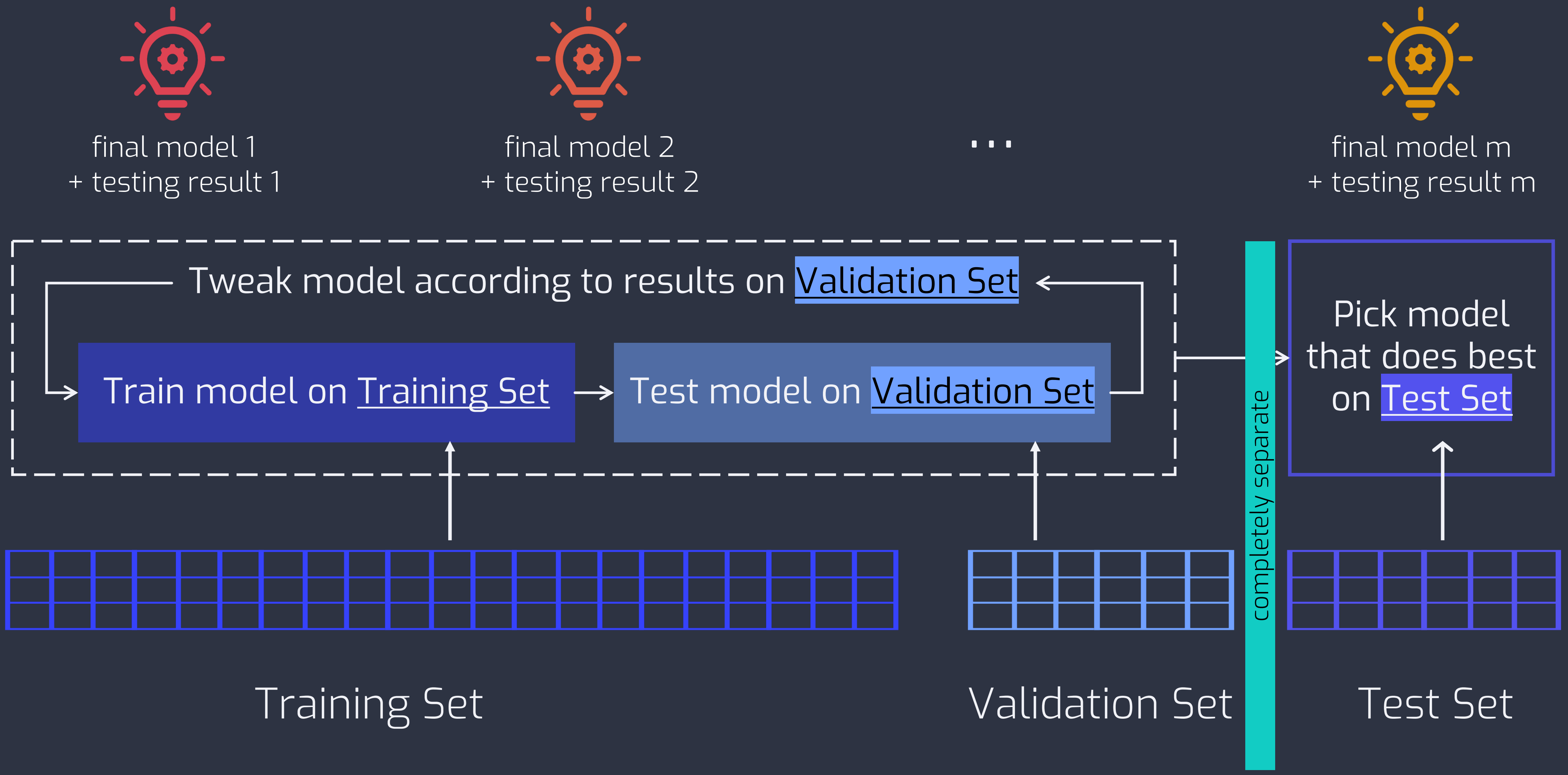
Evaluation metrics

Data


model
Tweak model according to results on Test Set
Train model on Training Set
Test model on Test Set
Pick model that does best on Test Set
Training Set
Test Set


final model 1
+ testing result 1
final model 2
+ testing result 2
...
final model m
+ testing result m
Tweak model according to results on Test Set
Train model on Training Set
Test model on Test Set
Pick model that does best on Test Set
Training Set
Test Set


final model 1
+ testing result 1
final model 2
+ testing result 2
...
final model m
+ testing result m
Tweak model according to results on Test Set
Train model on Training Set
Test model on Test Set
Pick model that does best on Test Set
completely separate
Training Set
Test Set



final model 1
+ testing result 1
final model 2
+ testing result 2
...
final model m
+ testing result m
Tweak model according to results on Validation Set
Train model on Training Set
Test model on Validation Set
completely separate
Pick model that does best on Test Set
Training Set
Validation Set
Test Set

However...

- The size of the Training Set is reduced, which the models are trained on.
- Not sure if the dataset is split in the best way.

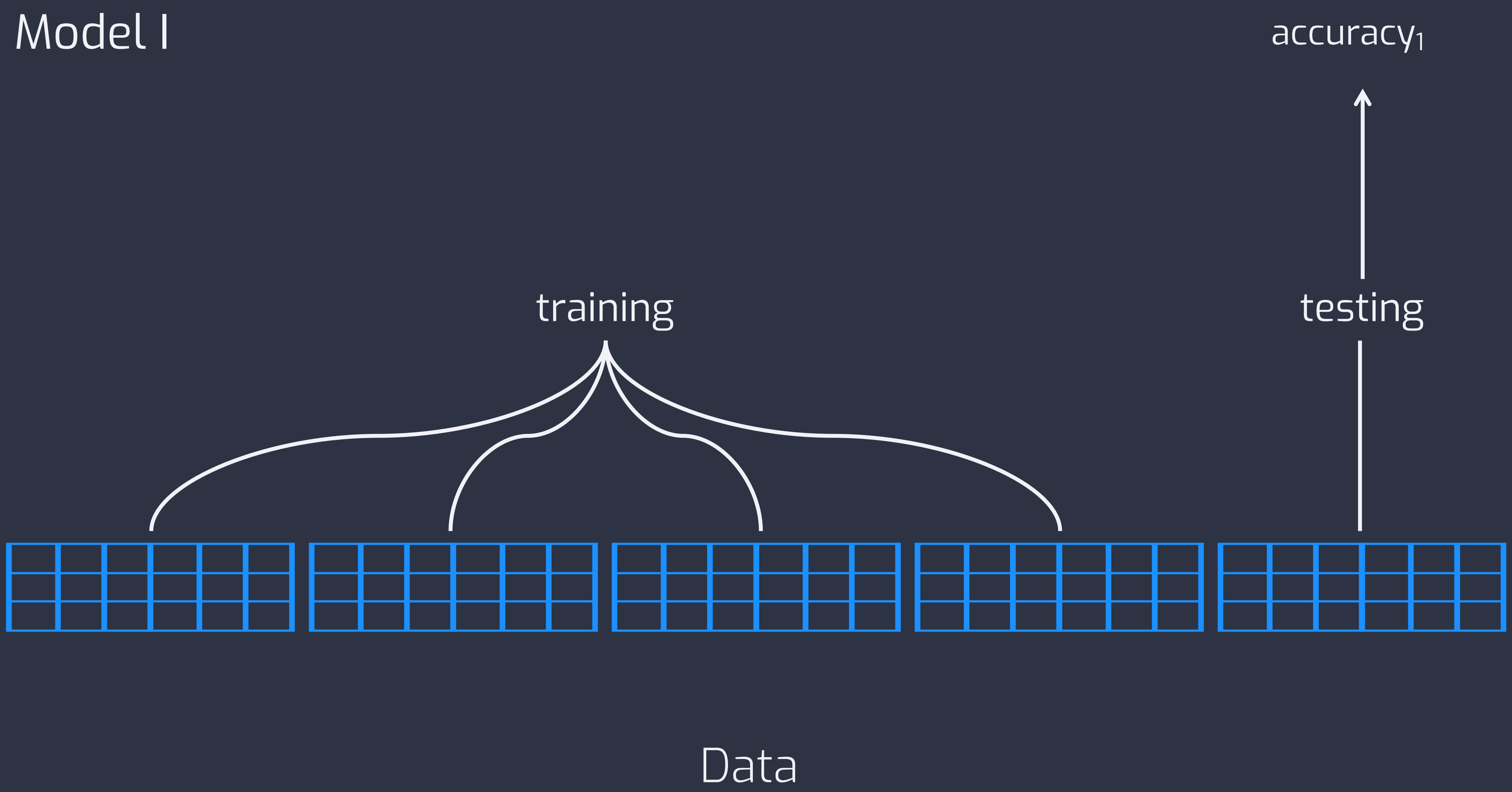
  (results may depend on a particular random choice of data split.)

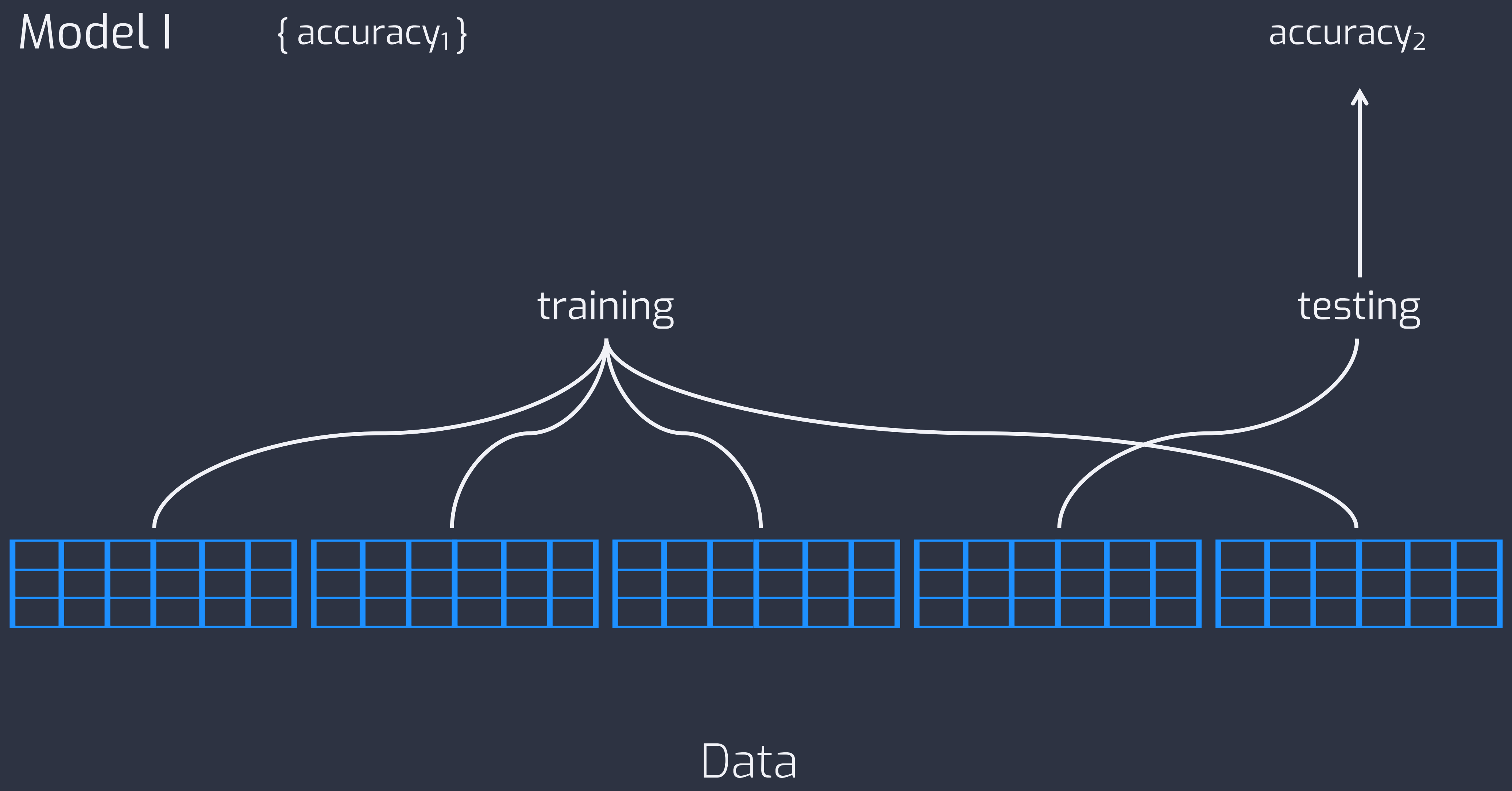
Cross Validation

Cross Validation (rotation estimation / out-of-sample testing)

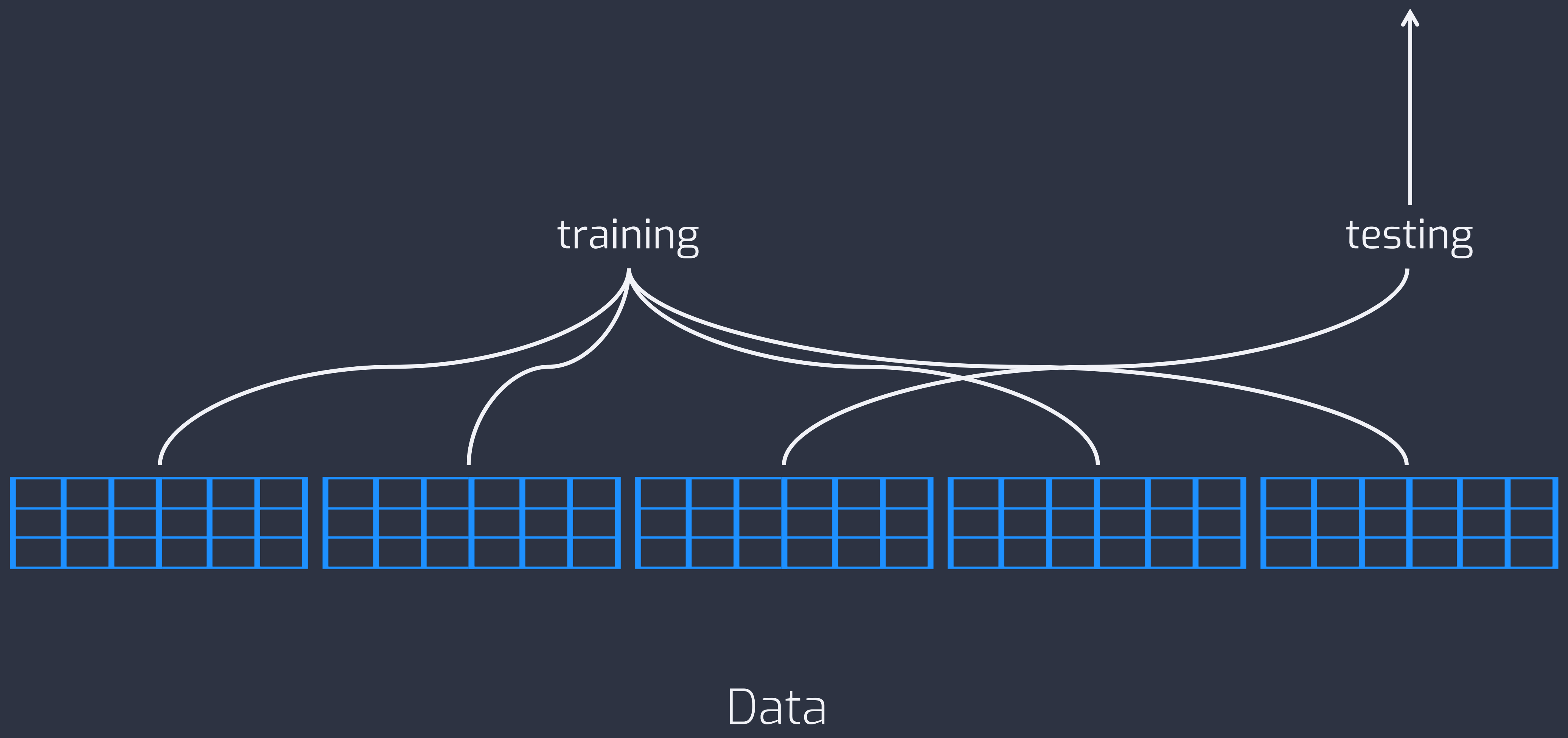
- Reduce the change of overfitting.
- Assess how well a model performs on previously unseen data.
- Is a resampling procedure to test models on a limited data sample.

Data

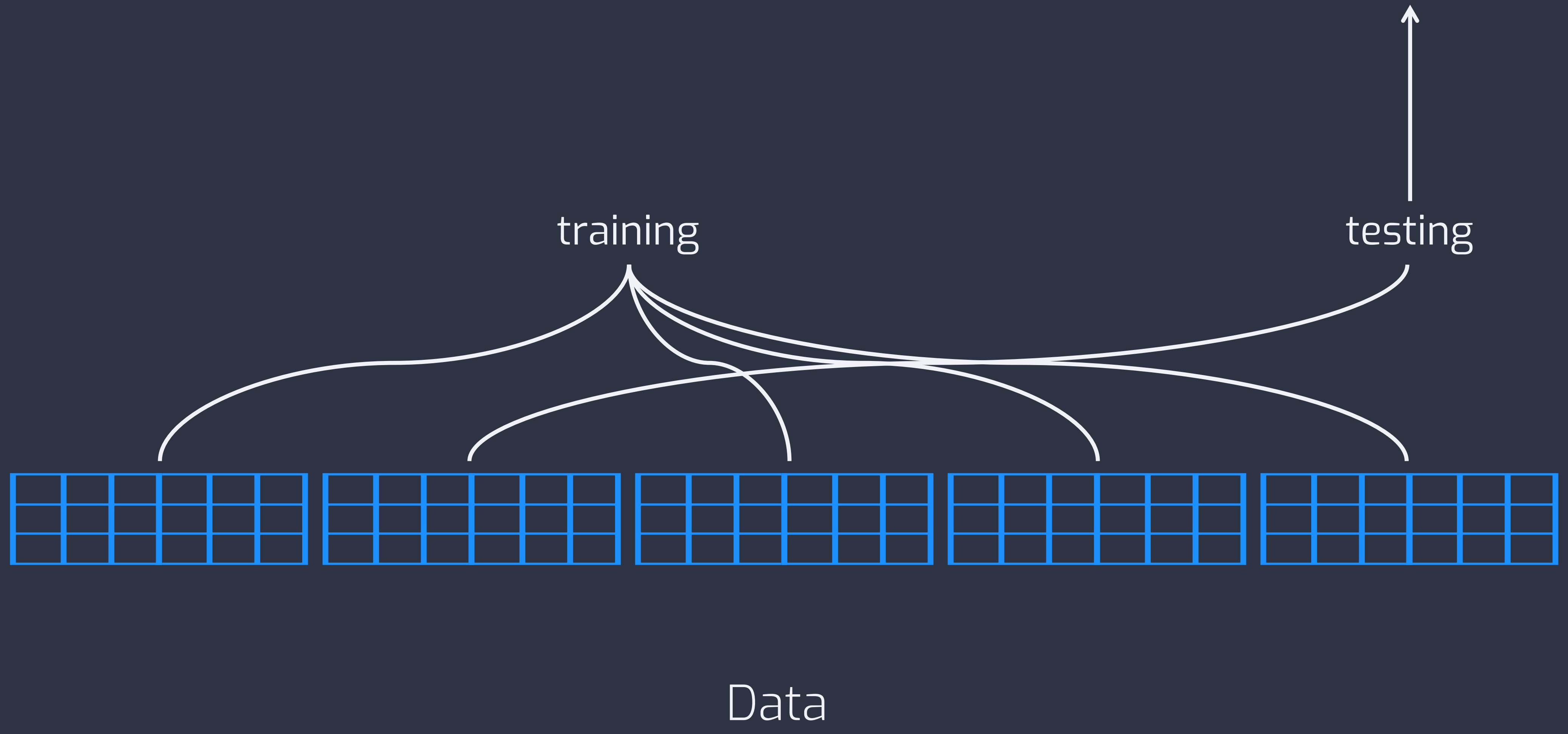

Model I
accuracy$_1$
training
testing
Data


Model I
{ $accuracy_1$ }
$accuracy_2$
training
testing
Data

Model I { $accuracy_1$, $accuracy_2$ } $accuracy_3$



Model I $\{ accuracy_1, accuracy_2, accuracy_3 \}$ $accuracy_4$



Model I $\{ accuracy_1 , accuracy_2 , accuracy_3 , accuracy_4 \}$ $accuracy_5$

Data

Summarise the result for the trained model

$$\text{Accuracy}_{\text{Model I}} = \frac{1}{5}(\text{accuracy}_1 + \text{accuracy}_2 + \text{accuracy}_3 + \text{accuracy}_4 + \text{accuracy}_5)$$

$$\text{Accuracy}_{\text{Model II}} = \frac{1}{5}(\text{accuracy}_1 + \text{accuracy}_2 + \text{accuracy}_3 + \text{accuracy}_4 + \text{accuracy}_5)$$

$$\text{Accuracy}_{\text{Model III}} = \frac{1}{5}(\text{accuracy}_1 + \text{accuracy}_2 + \text{accuracy}_3 + \text{accuracy}_4 + \text{accuracy}_5)$$

$$\text{Accuracy}_{\text{Model IV}} = \frac{1}{5}(\text{accuracy}_1 + \text{accuracy}_2 + \text{accuracy}_3 + \text{accuracy}_4 + \text{accuracy}_5)$$

$$\text{Accuracy}_{\text{Model V}} = \frac{1}{5}(\text{accuracy}_1 + \text{accuracy}_2 + \text{accuracy}_3 + \text{accuracy}_4 + \text{accuracy}_5)$$

Compare the result for each trained model

$\text{Accuracy}_{\text{Model II}}$

$\text{Accuracy}_{\text{Model I}}$

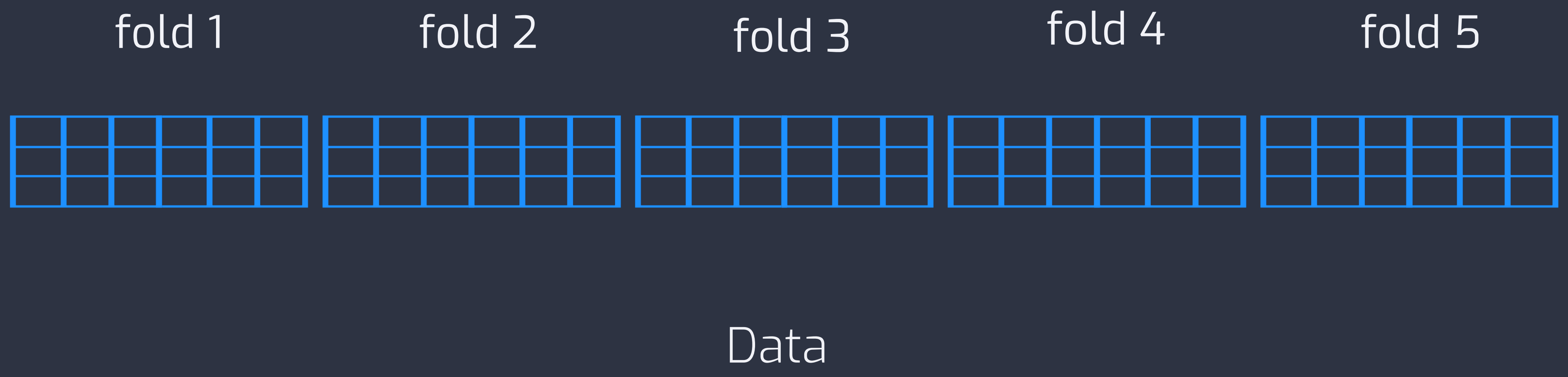
$\text{Accuracy}_{\text{Model III}}$

Which is the best?

$\text{Accuracy}_{\text{Model IV}}$

$\text{Accuracy}_{\text{Model V}}$

5-Fold Cross-Validation

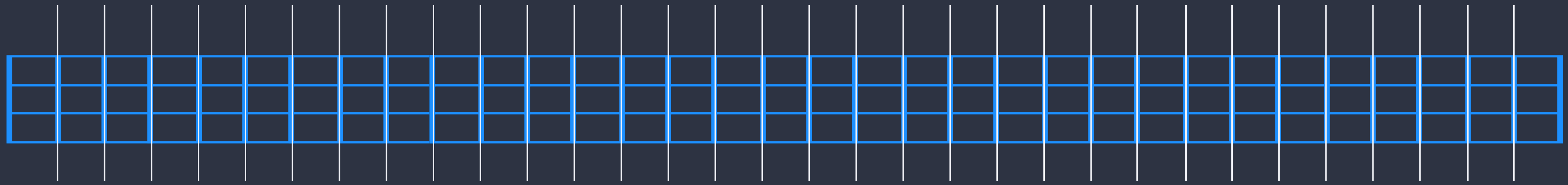


K-Fold Cross-Validation

Data

- k is arbitrary & may depend on size of dataset and how many models to train and compare.
- Larger k means less (pessimistic) bias.

Leave-One-Out Cross-Validation



- The largest possible k is equal to the number of instances in the original dataset.

- One instance each fold – and each instance is tested individually.

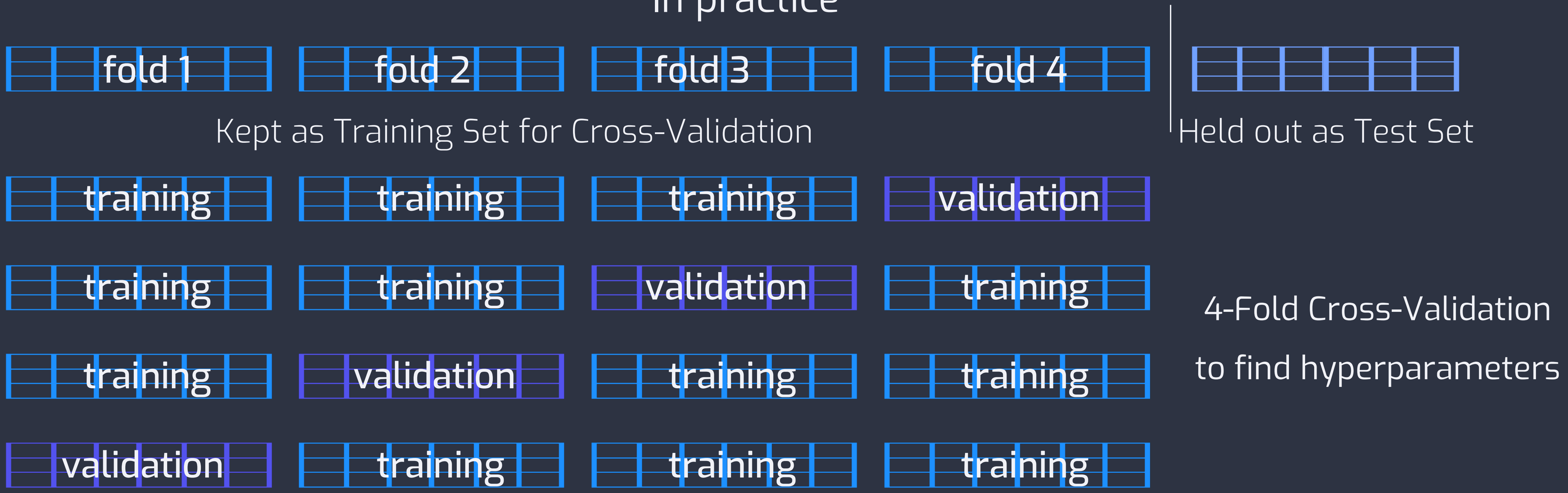
In practice

Kept as Training Set for Cross-Validation

Held out as Test Set

- Before cross validation for training, dataset randomly shuffled.
- Then, a subset is held out as Test Set, which will not be used in training at all.
- Test Set will be used only for the final evaluation of trained models.
- The rest dataset is kept as Training Set and to be used during cross validation.

In practice



1. Split Training Set into k smaller folds, of approximately equal size.
2. Now, run training k times, each time:
   - Train a model using k-1 of the folds
   - Validate the resulting model using the remaining 1 fold
3. Calculate the average of the performance measured in the loop, as the K-Fold CV result.

In practice

fold 1	fold 2	fold 3	fold 4	
Kept as Training Set for Cross-Validation				Held out as Test Set
training	training	training	validation	
training	training	validation	training	
training	validation	training	training	
validation	training	training	training	

Computationally expensive.

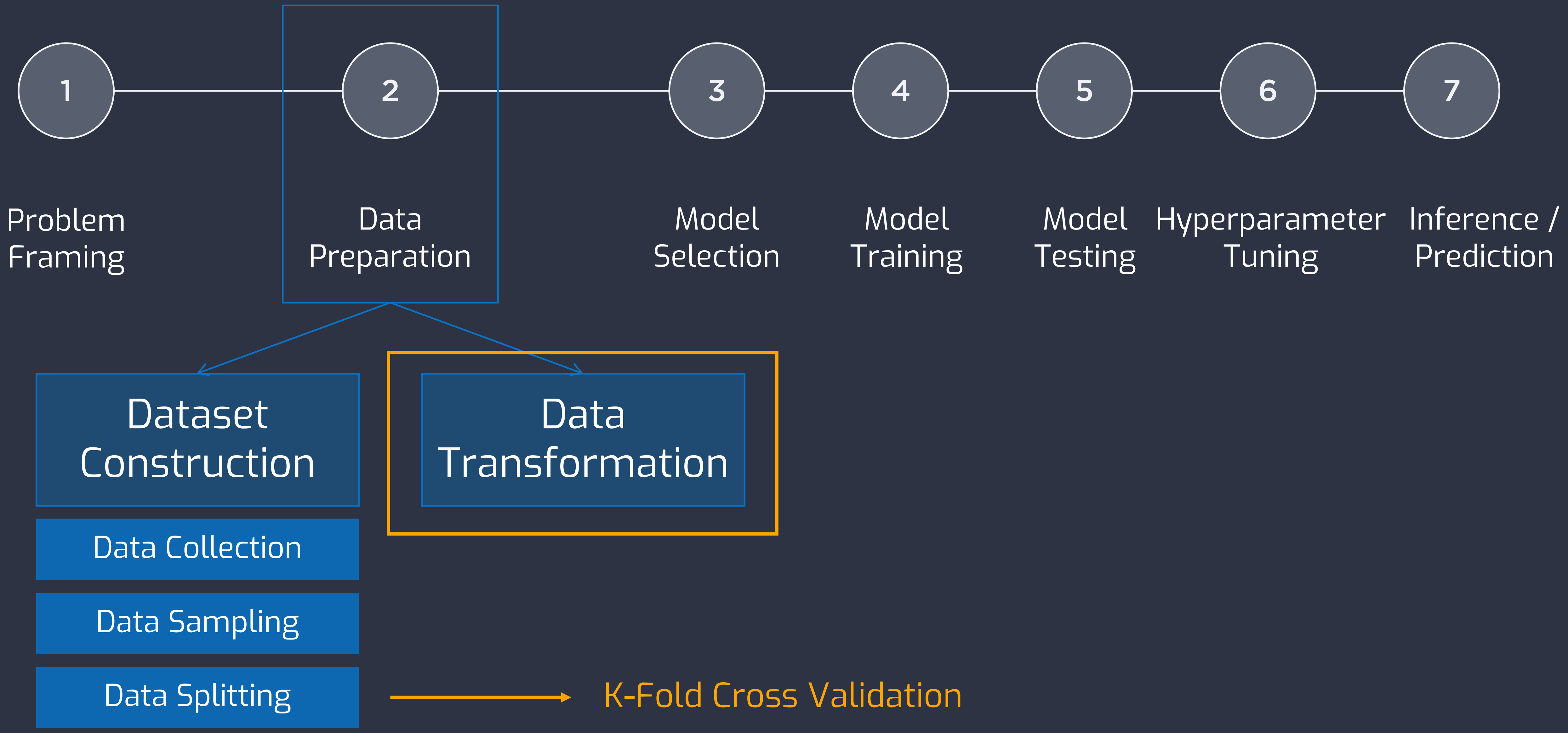
Re-uses data, which is very efficient, especially in case of small size of datasets.

Note, each instance is assigned into one fold and it stays in that fold for the whole k-Fold Cross-Validation procedure. This means that each instance has only one chance to be held out in the training Set and has k-1 times being used to train the model.

1. Split Training Set into k smaller folds, of approximately equal size.
2. Now, run training k times, each time:
   - Train a model using k-1 of the folds
   - Validate the resulting model using the remaining 1 fold
3. Calculate the average of the performance measured in the loop, as the K-Fold CV result.

Takeaway Points

- Must ensure Test Set is completely separate from Training
- Cross-validation is preventative measure against overfitting.
- Cross-validation can use data efficiently, especially small dataset.
- Cross-validation can help with determining hyperparameters.


1
2
3
4
5
6
7
Problem Framing
Data Preparation
Model Selection
Model Training
Model Testing
Hyperparameter Tuning
Inference / Prediction
Dataset Construction
Data Transformation
Data Collection
Data Sampling
Data Splitting
K-Fold Cross Validation