Cross Validation for Evaluating Algorithms

Cynthia Rudin

Machine Learning Course, Duke

“Cross-validation” has multiple meanings

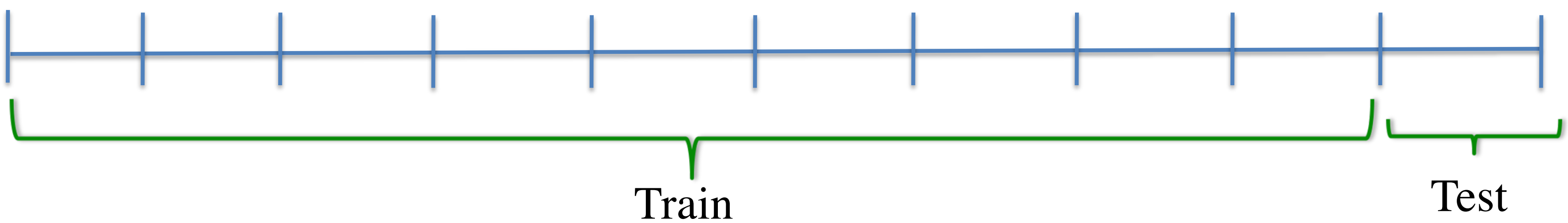
- “We evaluated the algorithm by 10 fold cross-validation”
- “The parameters of the algorithm were tuned by 10-fold cross-validation” (part of nested cross-validation)

Cross-Validation

- Cross Validation (CV) is the most popular way to evaluate a machine learning algorithm on a dataset.
- You will need a dataset, an algorithm, and an evaluation measure.
- The evaluation measure might be the squared error between the predictions and the truth. Or it might be misclassification error.

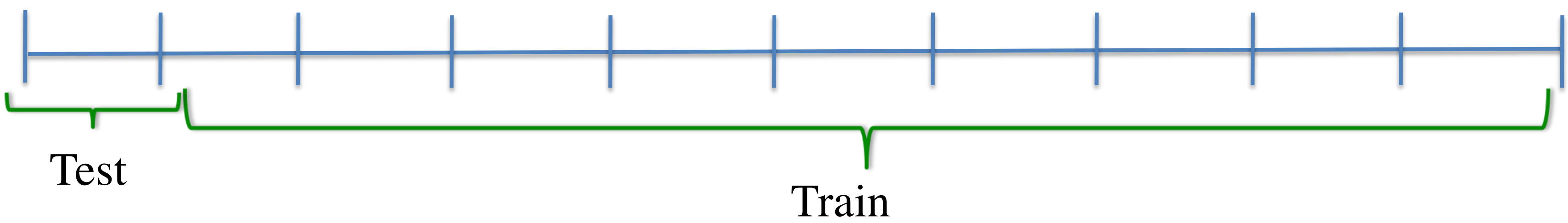
- Divide the data into approximately-equally sized 10 "folds"
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation



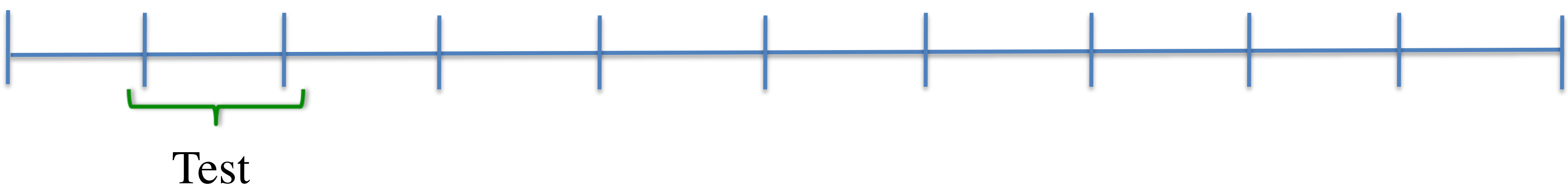
- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation



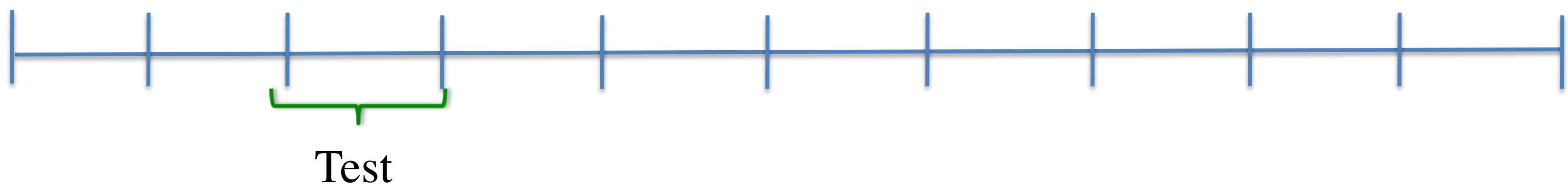
- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation



- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation



- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation

- The algorithm that performed the best was the one with the best average out-of-sample performance across the 10 test folds.
- If desired, compute significance tests on performance across folds.

Alg 1	Alg 2	Alg 3
.87$\pm$.01	.85$\pm$.04	.81$\pm$.03

Coming Soon

- CV for tuning parameters
- Nested CV

Cross Validation for Tuning Parameters

Cynthia Rudin

Machine Learning Course, Duke

"Cross-validation" has multiple meanings

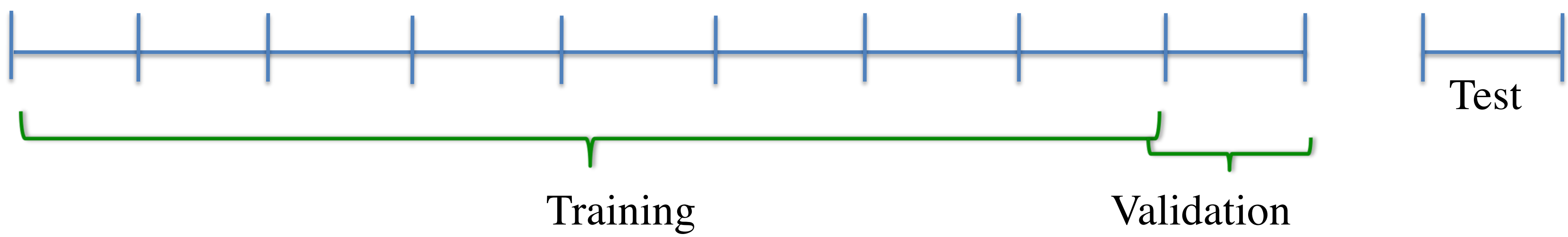
- "We evaluated the algorithm by 10 fold cross-validation"
- "The parameters of the algorithm were tuned by 10-fold cross-validation" (part of nested cross-validation)

We'll call the parameter K and it takes values 1, 10, 100, 1000, or 10000.

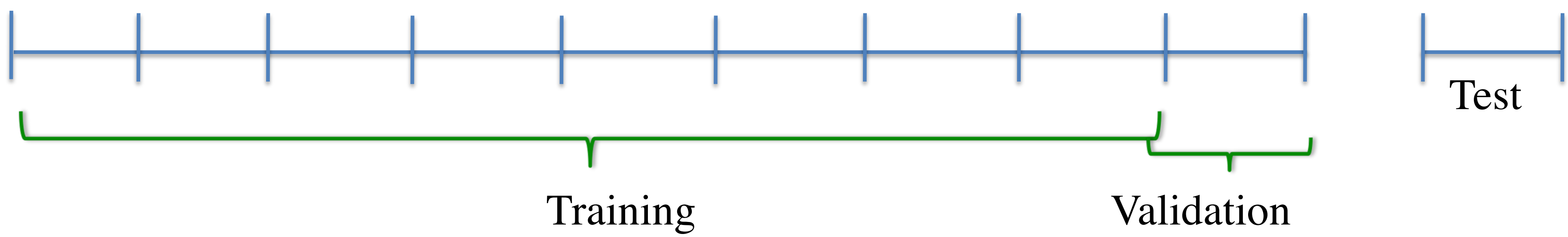
- **Set aside the test fold.**
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- Rotate the validation fold and repeat.
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.

Test

- Set aside the test fold.
- **Reserve a validation set from the training set.**
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- Rotate the validation fold and repeat.
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



- Set aside the test fold.
- Reserve a validation set from the training set.
- **Train the algorithm on the rest of training set for each K, evaluate on validation set.**
- Rotate the validation fold and repeat.
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



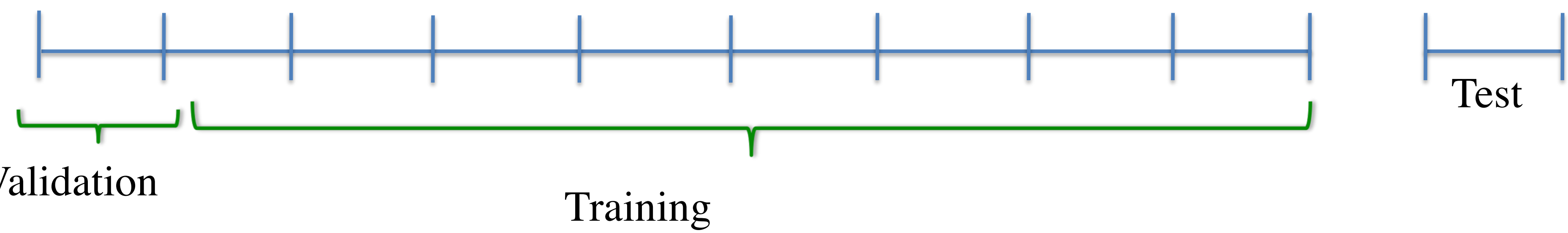
K=1: Accuracy is 86%

K=100: Accuracy is 91%

K=1000: Accuracy is 57%

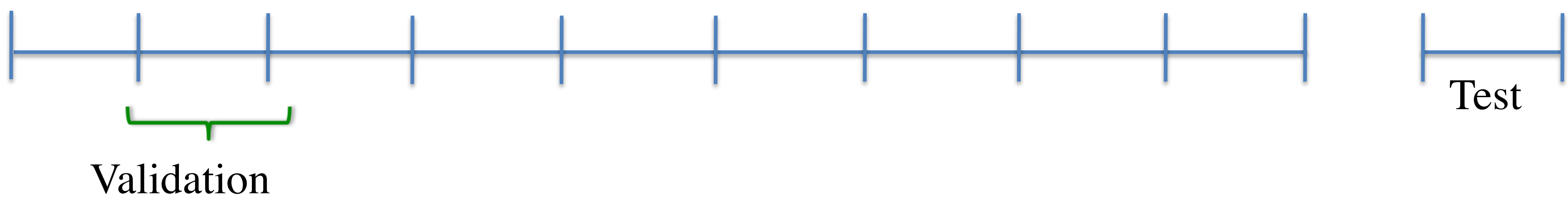
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



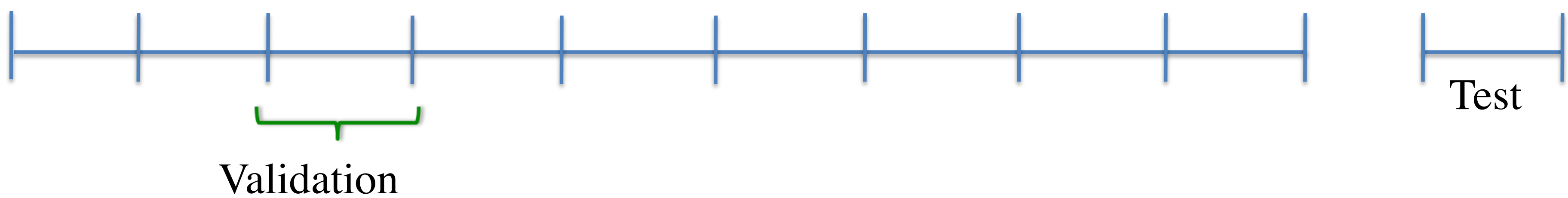
K=1: Accuracy is 83%
K=100: Accuracy is 94%
K=1000: Accuracy is 75%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



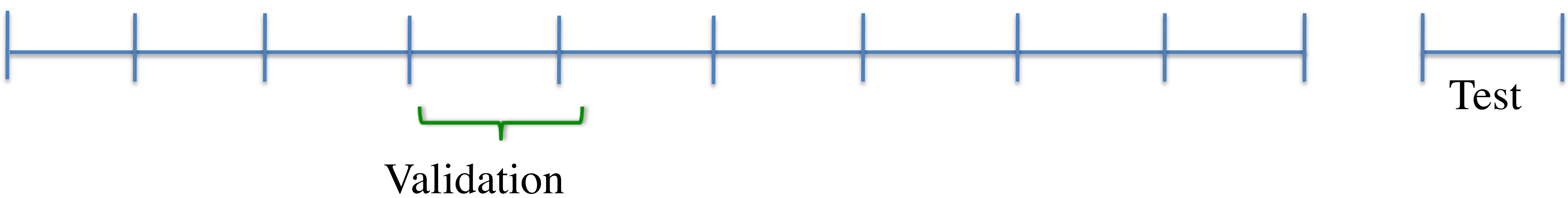
K=1: Accuracy is 82%
K=100: Accuracy is 79%
K=1000: Accuracy is 72%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



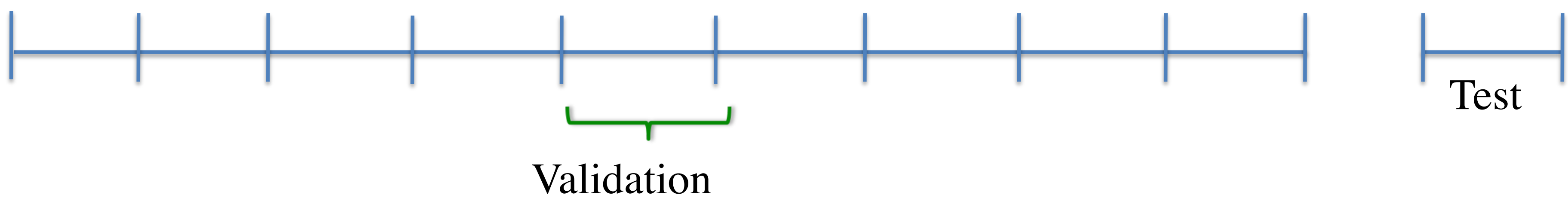
K=1: Accuracy is 87%
K=100: Accuracy is 92%
K=1000: Accuracy is 81%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



K=1: Accuracy is 83%
K=100: Accuracy is 94%
K=1000: Accuracy is 75%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.

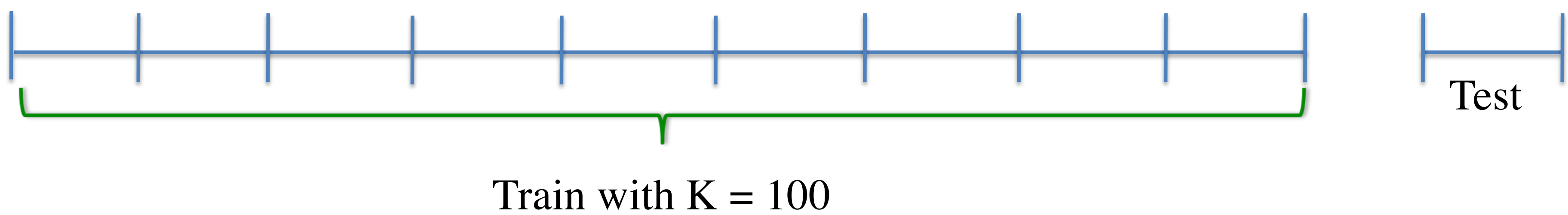


K=1: Accuracy is 81%
K=100: Accuracy is 90%
K=1000: Accuracy is 72%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- Rotate the validation fold and repeat.
- **Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.**
- Train on the full training set (training + validation) with best K, evaluate on test set.

K=1: Accuracy is 83%
K=100: Accuracy is 94% Best K
K=1000: Accuracy is 75%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- Rotate the validation fold and repeat.
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- **Train on the full training set (training + validation) with best K, evaluate on test set.**



Coming Soon

- Nested CV
  - Uses CV for evaluation as an outer loop, and CV for tuning parameters as an inner loop.

Nested Cross Validation

Cynthia Rudin

Machine Learning Course, Duke

“Cross-validation” has multiple meanings

- “We evaluated the algorithm by 10 fold cross-validation”
- “The parameters of the algorithm were tuned by 10-fold cross-validation” (part of nested cross-validation)

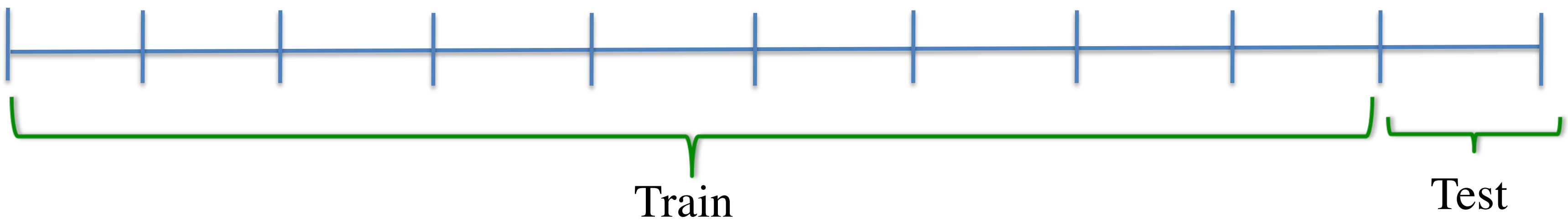
Nested Cross-validation combines both.

Nested CV evaluates an algorithm including parameter tuning

- Outer loop: CV for evaluation
- Inner loop: CV for parameter tuning

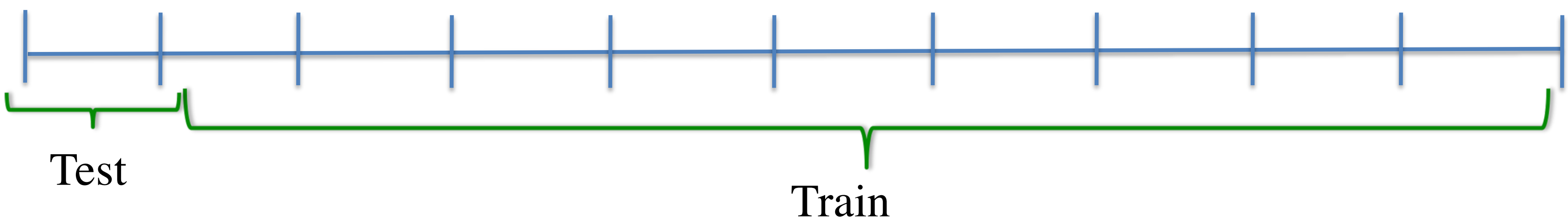
Nested CV evaluates an algorithm including parameter tuning

Cross-Validation for Evaluation



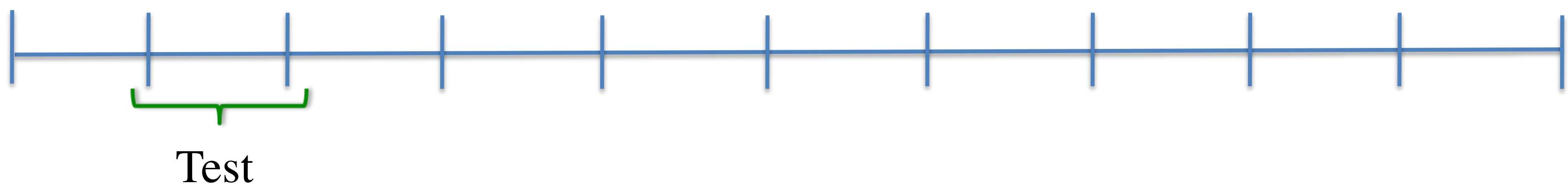
- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation for Evaluation



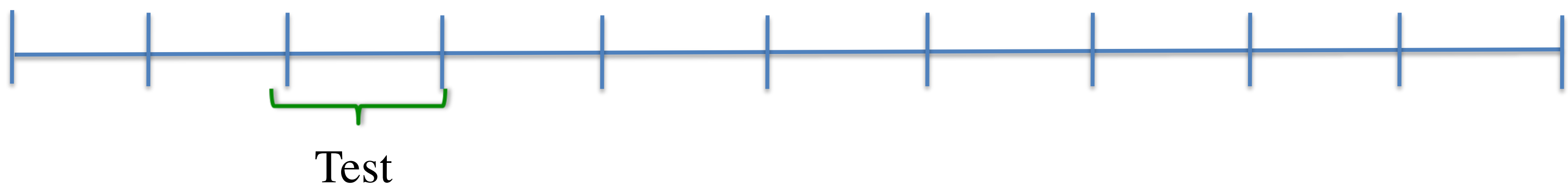
- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation for Evaluation



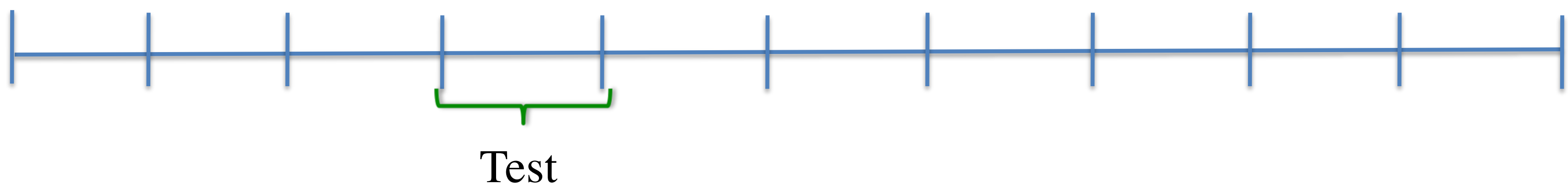
- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation for Evaluation



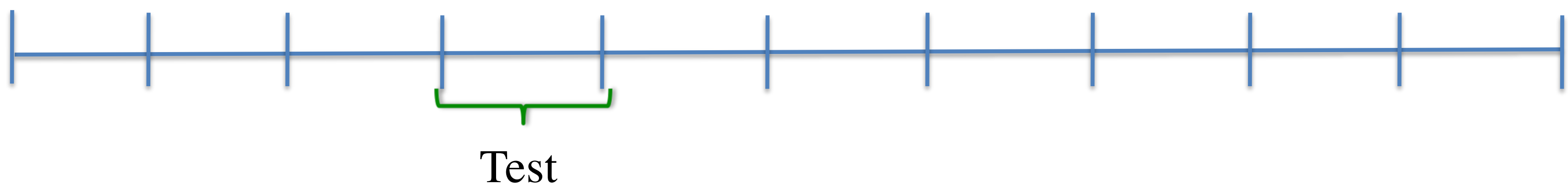
- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

Cross-Validation for Evaluation



- Divide the data into approximately-equally sized 10 “folds”
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

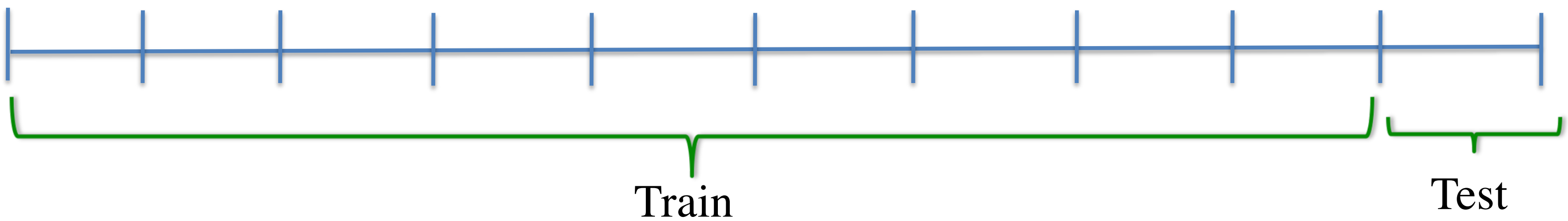
Cross-Validation for Evaluation



- Divide the data into approximately-equally sized 10 "folds"
- Train the algorithm on 9 folds, compute the evaluation measure on the last fold.
- Repeat this 10 times, using each fold in turn as the test fold.
- Report the mean and standard deviation of the evaluation measure over the 10 folds.

- Outer loop: CV for evaluation
- Inner loop: CV for parameter tuning

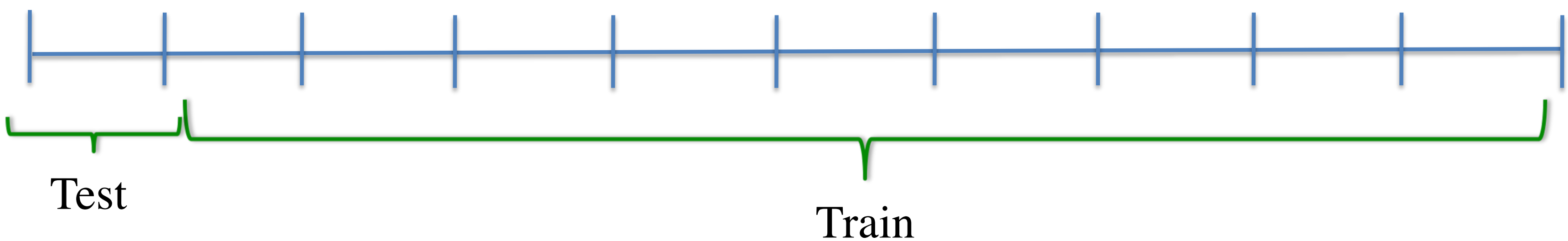
Nested CV evaluates an algorithm including parameter tuning



Best K=100

Test Accuracy = 87%

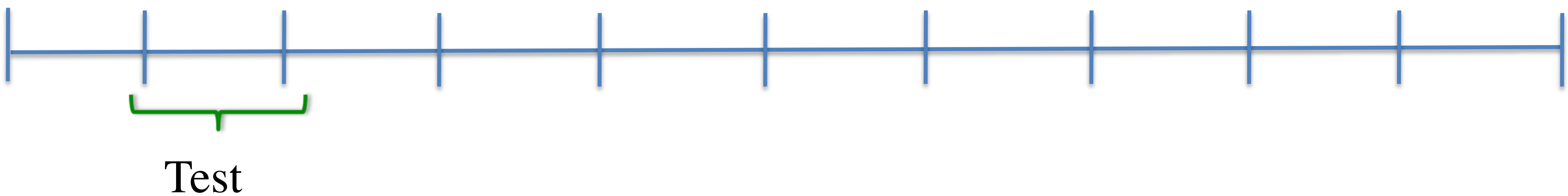
(I got this from CV inside the training set)



Test Accuracy = 86%

Best K=10000

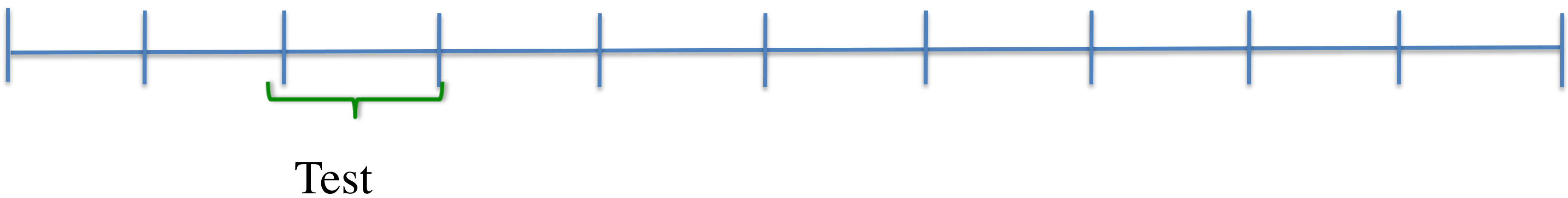
(I got this from CV inside the training set)



Test Accuracy = 89%

Best K=1

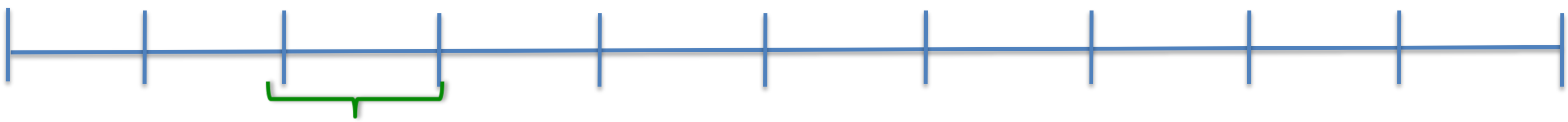
(I got this from CV inside the training set)



Test Accuracy = 86%

Best K=100

(I got this from CV inside the training set)



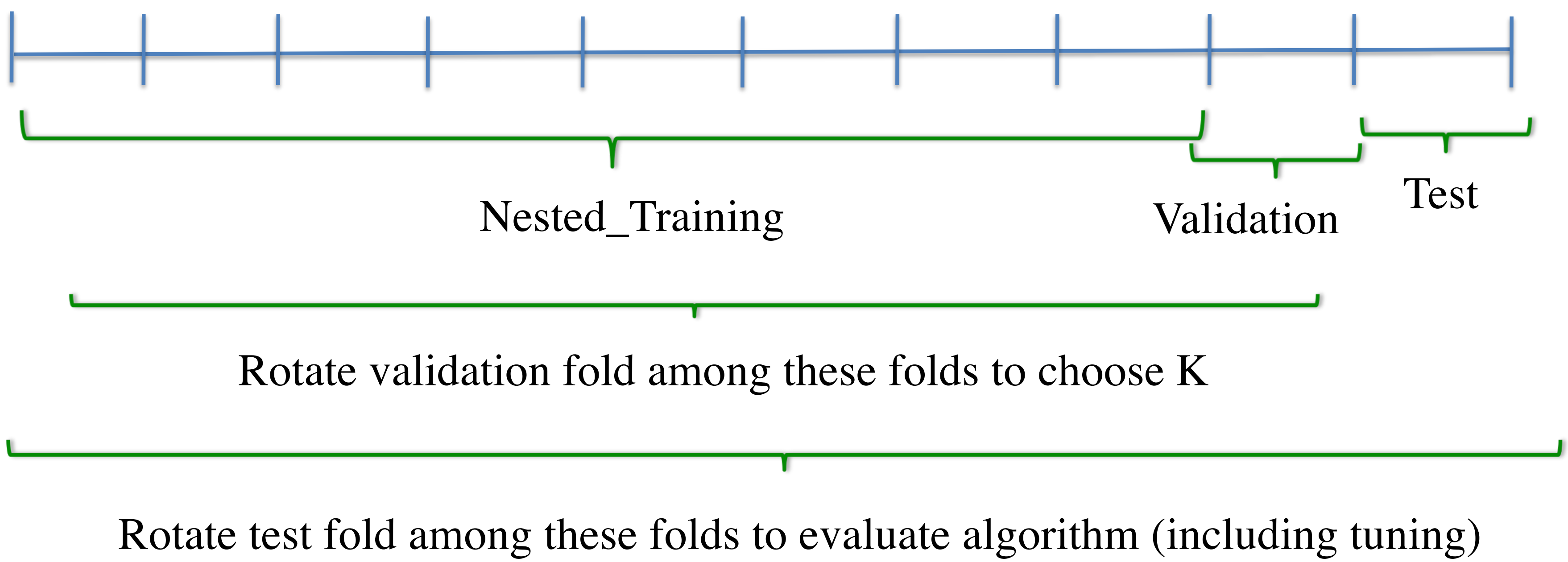
Test Accuracy = 86%

Best K=100

(I got this from CV inside the training set)

Nested Cross-Validation

- …is a lot of work



Nested Cross-Validation

- Outer loop: CV for evaluation
- Inner loop: CV for parameter tuning

Nested CV evaluates an algorithm including parameter tuning

A common question

- What is the “final model”?

Hint: Remember, Nested CV is for evaluating an algorithm. To produce a final model, you must ask about parameter tuning.

Cross Validation for Tuning Parameters

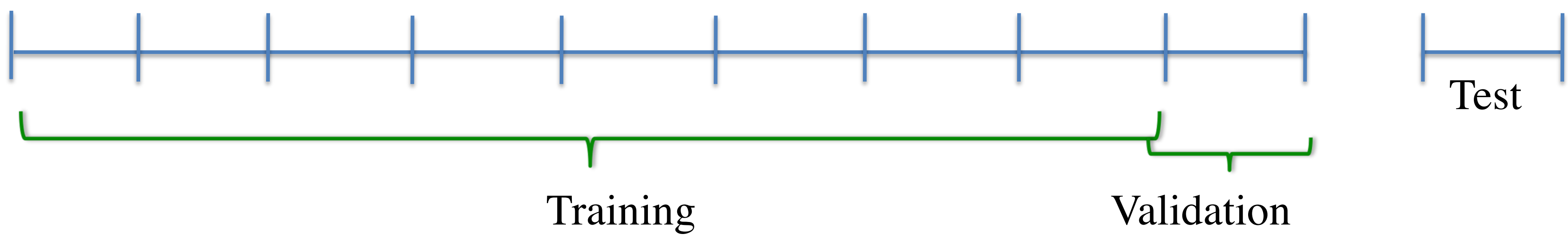
Cynthia Rudin

Machine Learning Course, Duke

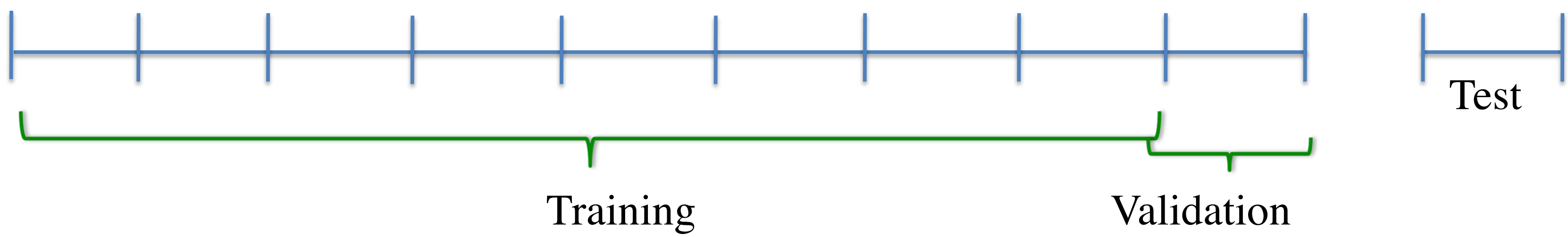
- **Set aside the test fold.**
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- Rotate the validation fold and repeat.
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.

Test

- Set aside the test fold.
- **Reserve a validation set from the training set.**
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- Rotate the validation fold and repeat.
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.

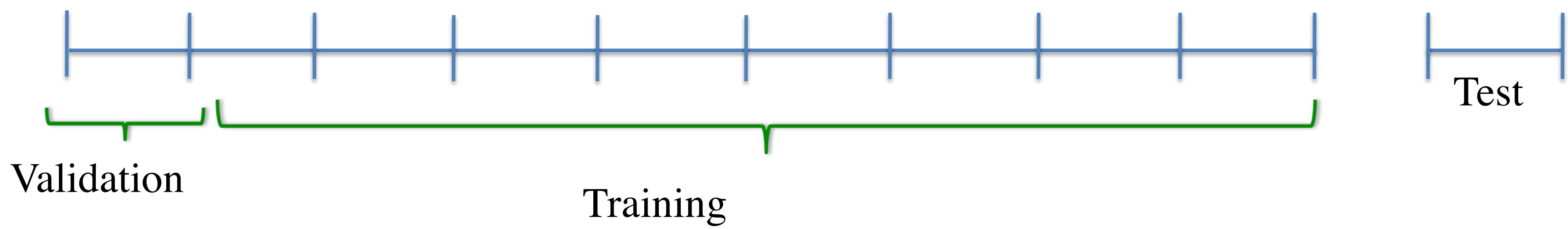


- Set aside the test fold.
- Reserve a validation set from the training set.
- **Train the algorithm on the rest of training set for each K, evaluate on validation set.**
- Rotate the validation fold and repeat.
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



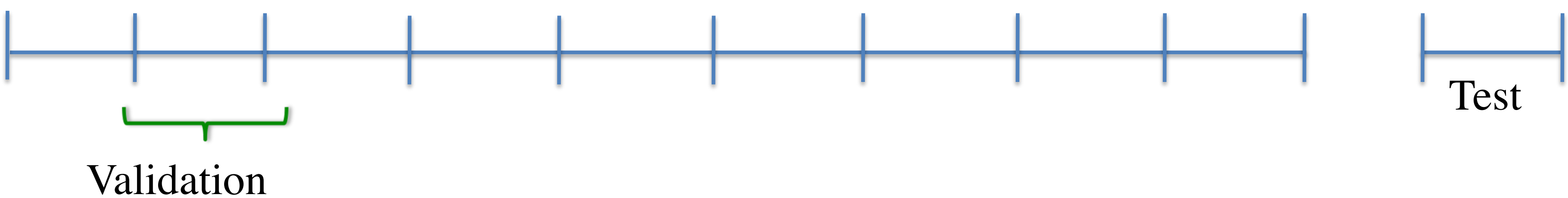
K=1: Accuracy is 86%
K=100: Accuracy is 91%
K=1000: Accuracy is 57%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



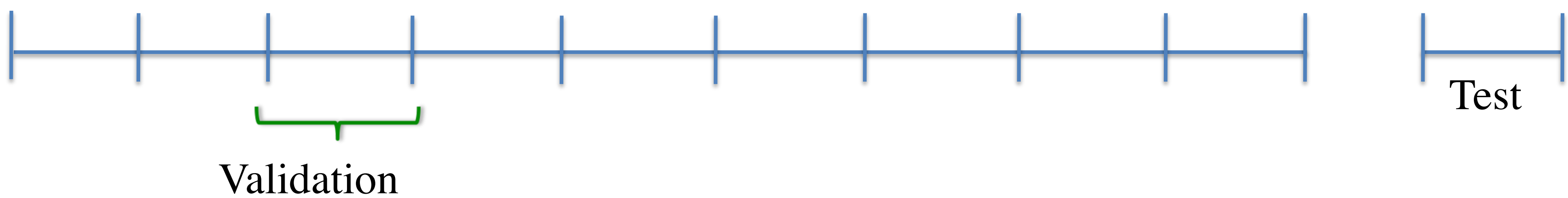
K=1: Accuracy is 83%
K=100: Accuracy is 94%
K=1000: Accuracy is 75%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



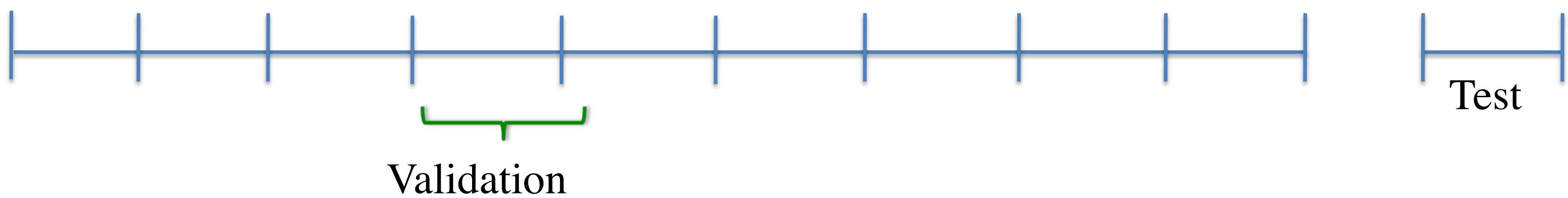
K=1: Accuracy is 82%
K=100: Accuracy is 79%
K=1000: Accuracy is 72%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



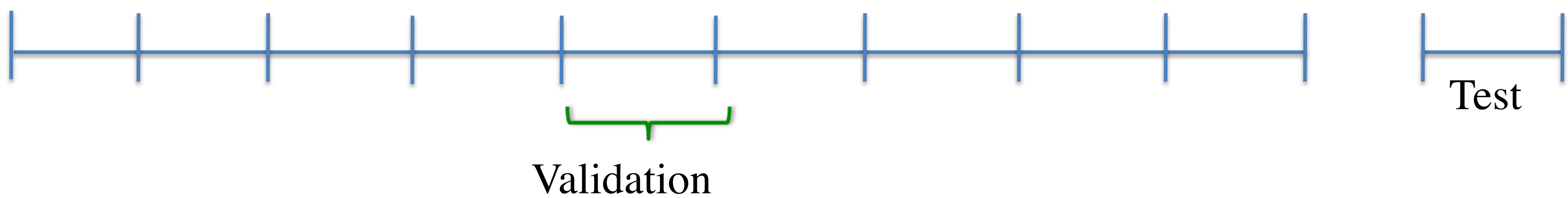
K=1: Accuracy is 87%
K=100: Accuracy is 92%
K=1000: Accuracy is 81%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.



K=1: Accuracy is 83%
K=100: Accuracy is 94%
K=1000: Accuracy is 75%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- **Rotate the validation fold and repeat.**
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- Train on the full training set (training + validation) with best K, evaluate on test set.

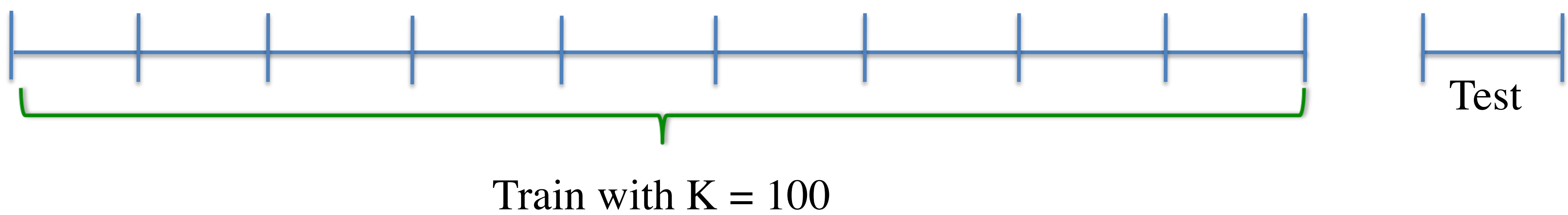


K=1: Accuracy is 81%
K=100: Accuracy is 90%
K=1000: Accuracy is 72%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- Rotate the validation fold and repeat.
- **Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.**
- Train on the full training set (training + validation) with best K, evaluate on test set.

K=1: Accuracy is 83%
K=100: Accuracy is 94% Best K
K=1000: Accuracy is 75%
:

- Set aside the test fold.
- Reserve a validation set from the training set.
- Train the algorithm on the rest of the training set for each K, evaluate on validation set.
- Rotate the validation fold and repeat.
- Report the mean of the evaluation measure for each K over the validation folds. Choose the best K.
- **Train on the full training set (training + validation) with best K, evaluate on test set.**



A common question

- What is the “final model”?