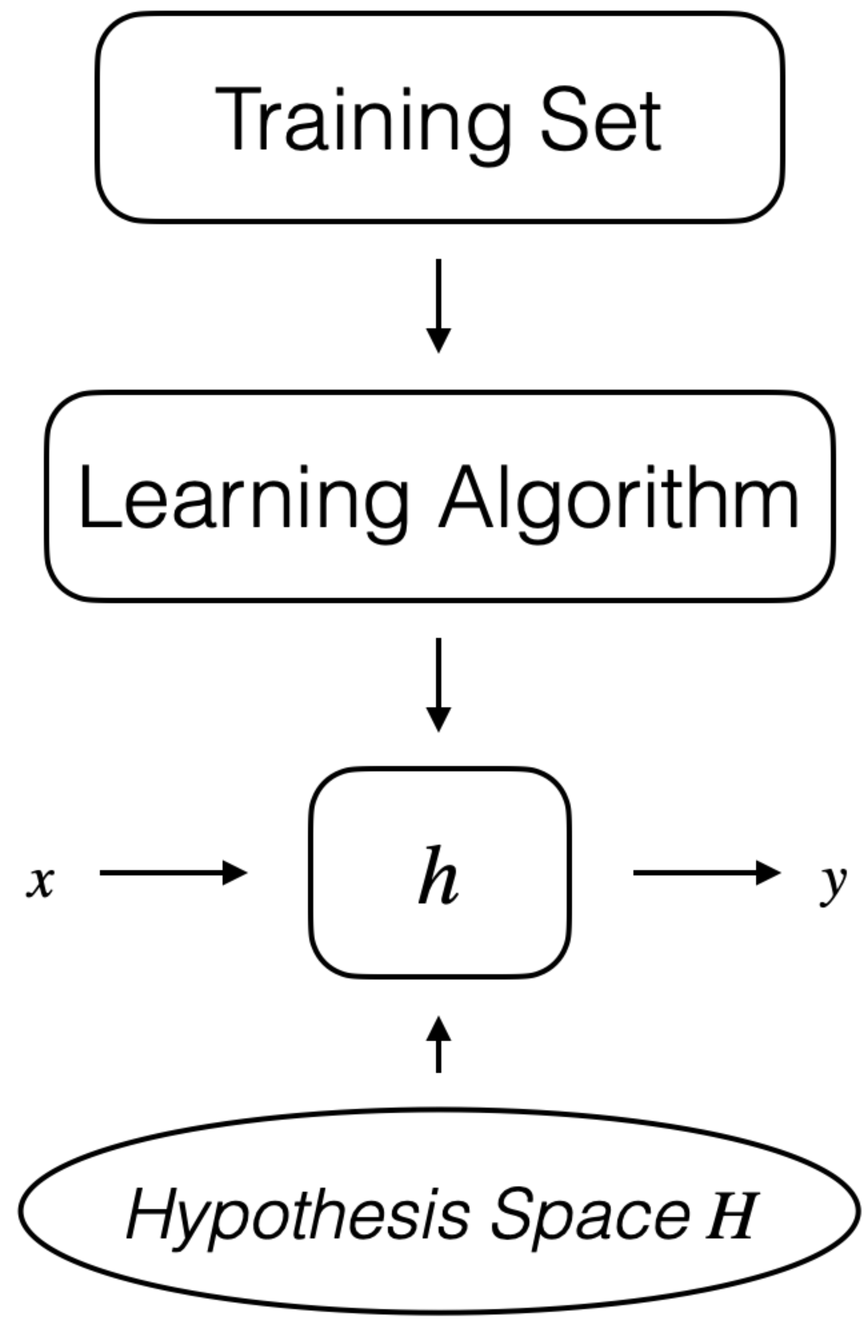
Model Selection

Recap: Supervised Learning



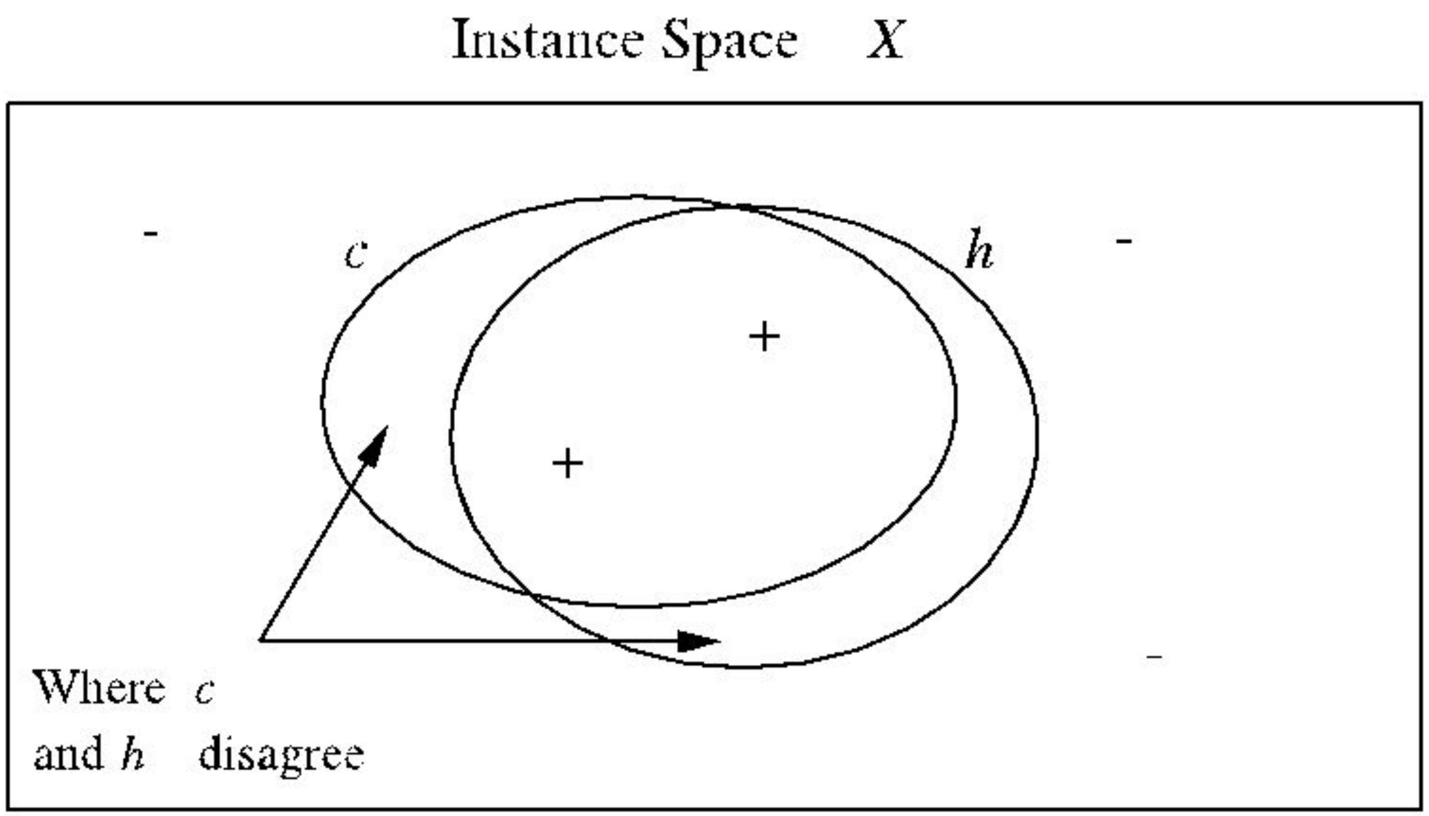
h approximates the unknown target f

$$h \sim f: X \rightarrow Y$$

The “Real” Scenario

- Let me know introduce a more formal definition for a machine learning problem
- Empirical Risk Minimization: it defines a family of learning algorithms and is used to give theoretical bounds on their performance
  - We don’t know how well a learning algorithm will work in practice (“true risk”) because we don’t know the true distribution of data
  - But we can measure its performance on our training dataset (“empirical risk”)

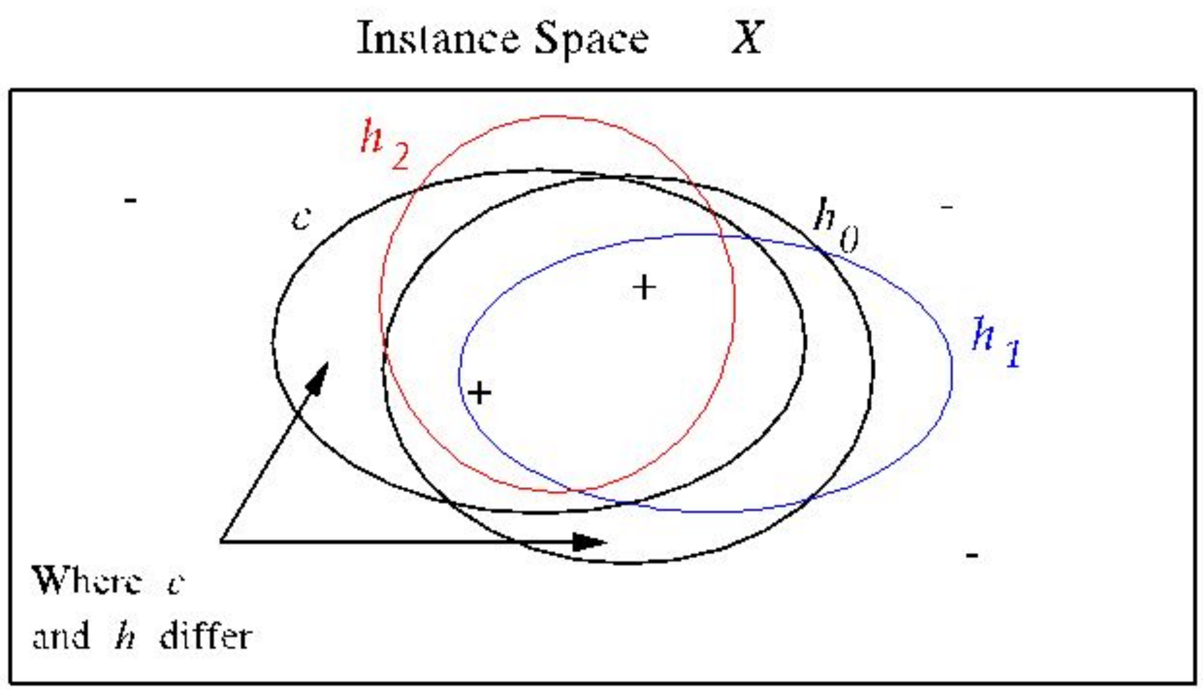
True Error



Def: The **True Error** ($error_{\mathcal{D}}(h)$) of hypothesis h with respect to target concept c and distribution $\mathcal{D}$ (to observe an input instance $x \in X$) is the probability that h will misclassify an instance drawn at random according to $\mathcal{D}$:

$$error_{\mathcal{D}}(h) \equiv \Pr_{x \in \mathcal{D}}[c(x) \neq h(x)]$$

Empirical Error



Def: The **Empirical Error** ($error_{Tr}(h)$) of hypothesis h with respect to Tr is the number of examples that h misclassifies:

$$error_{Tr}(h) = Pr_{(x,f(x))\in Tr}\,[f(x) \neq h(x)] = \frac{|\{(x,f(x)) \in Tr \mid f(x) \neq h(x)\}|}{|Tr|}$$

Def: $h \in \mathcal{H}$ overfits Tr if $\exists h' \in \mathcal{H}$ such that $error_{Tr}(h) < error_{Tr}(h')$, but $error_{\mathcal{D}}(h) > error_{\mathcal{D}}(h')$.

Estimating the True error

- Minimizing the error on the training set (Empirical Risk Minimization) may not be the best option (see overfitting later)

We want to minimize the true error!

$$error_{\mathcal{D}}(h) = error_{Tr}(h) + generalization(h)$$

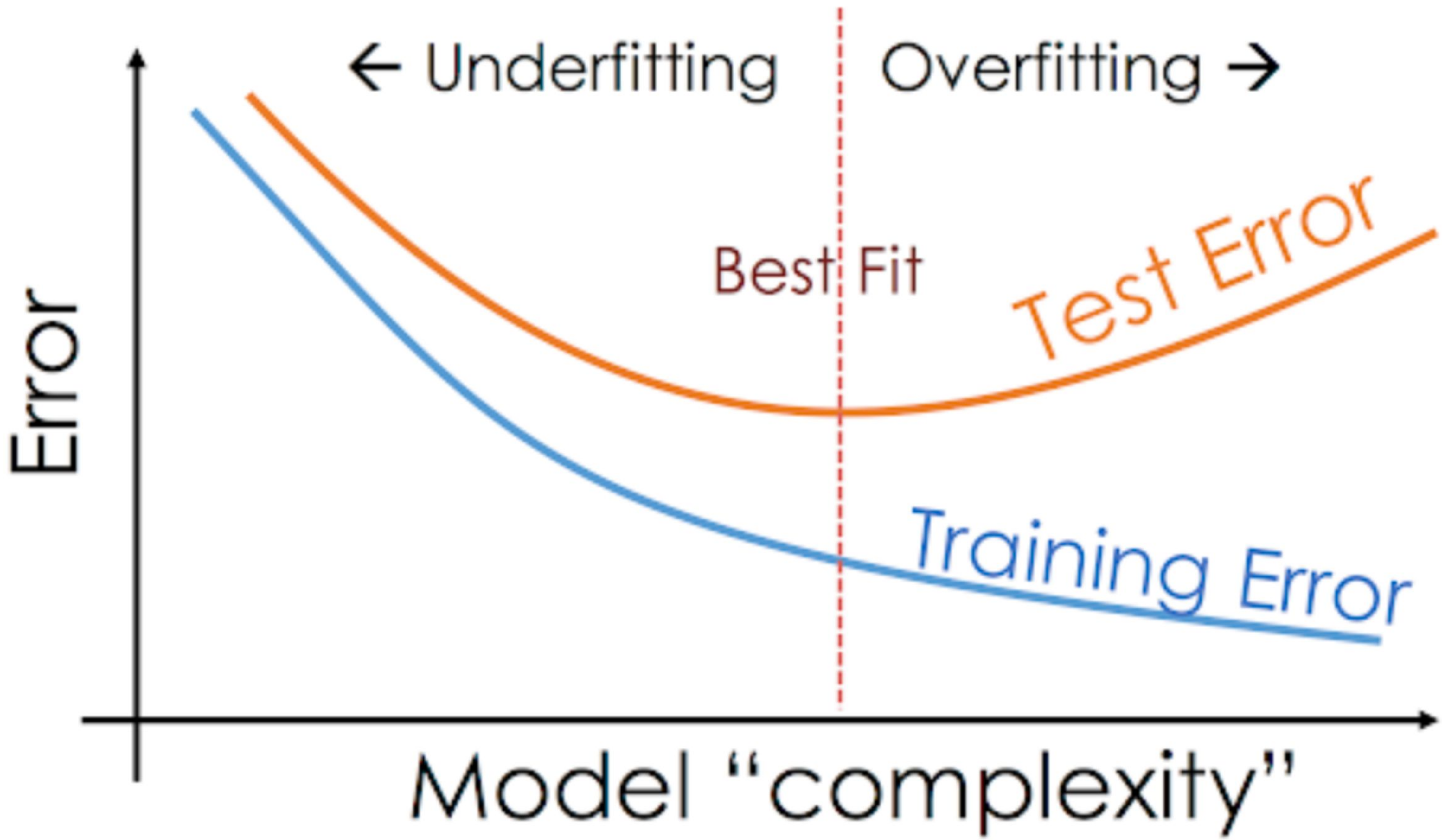
2 ways: bounds and estimation

1. relate the Empirical error and the true error with generalization bounds:

$$error_{\mathcal{D}}(h) \leq error_{Tr}(h) + complexityMeasure(\mathcal{H})$$

   with $\mathrm{h} \in \mathcal{H}$, exploiting some complexity measure of the hypothesis space

2. compute the error on unseen data (TEST set)


← Underfitting
Overfitting →
Best Fit
Test Error
Training Error
Error
Model "complexity"

Model Selection and Hold-out

We can hold out some of our original training data

Hold-out procedure

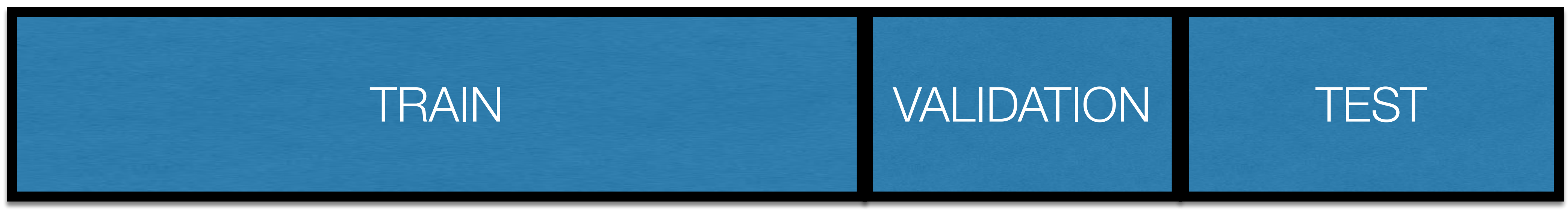
1. A small subset of Tr, called the $\boxed{\text{validation set}}$ (or hold-out set), denoted Va, is identified
2. A classifier/regressor is learnt using examples in $Tr - Va$
3. Step 2 is performed with different values of the parameter(s) (in our example, p), and tested against the hold-out sample

In an operational setting, after parameter optimization, one typically re-trains the classifier on the entire training corpus, in order to boost effectiveness (debatable step!)

It is possible to show that the evaluation performed in Step 2 gives an unbiased estimate of the error performed by a classifier learnt with the same parameter(s) and with training set of cardinality $|Tr| - |Va| < |Tr|$

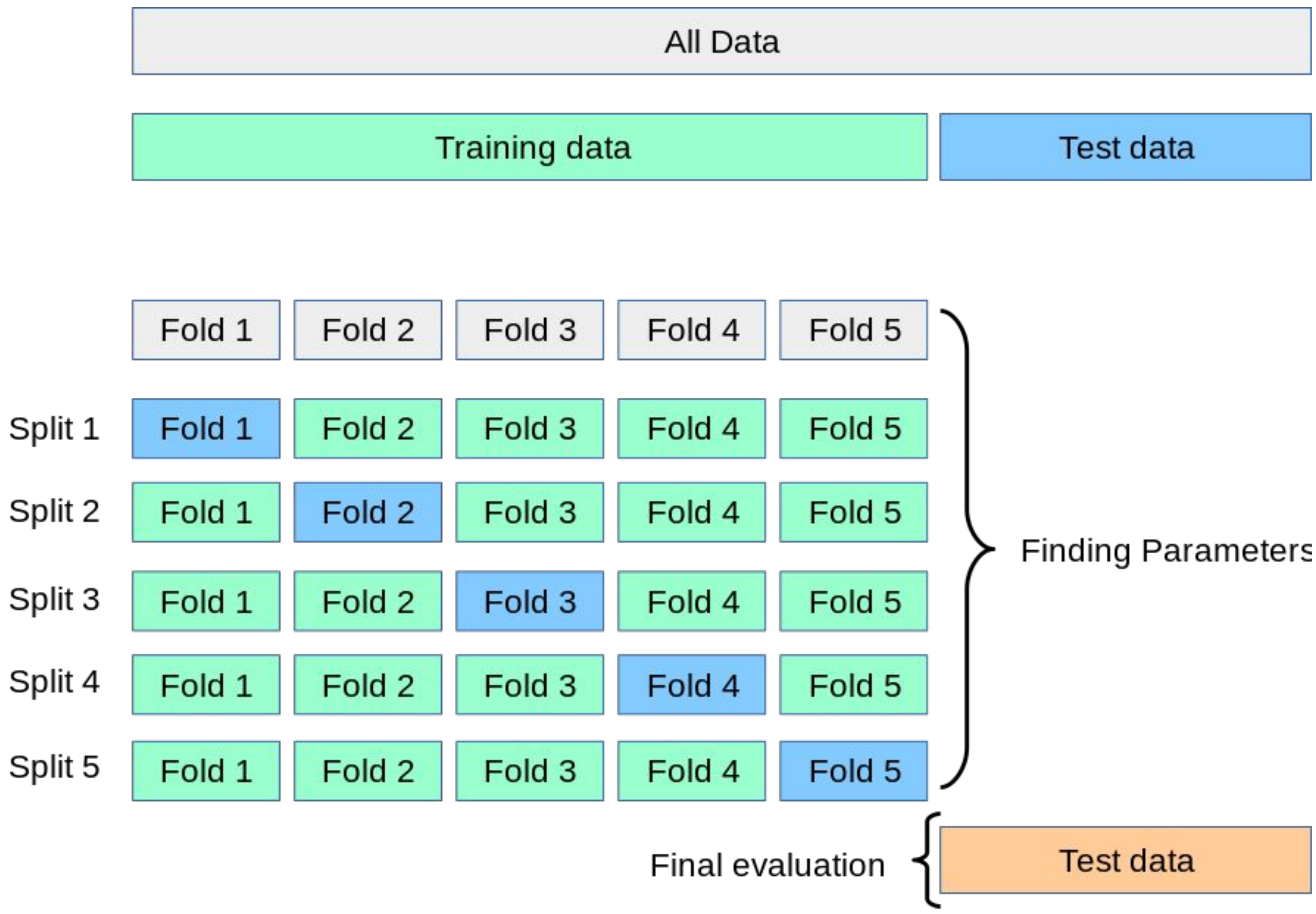
Model Selection – an example

Corpus

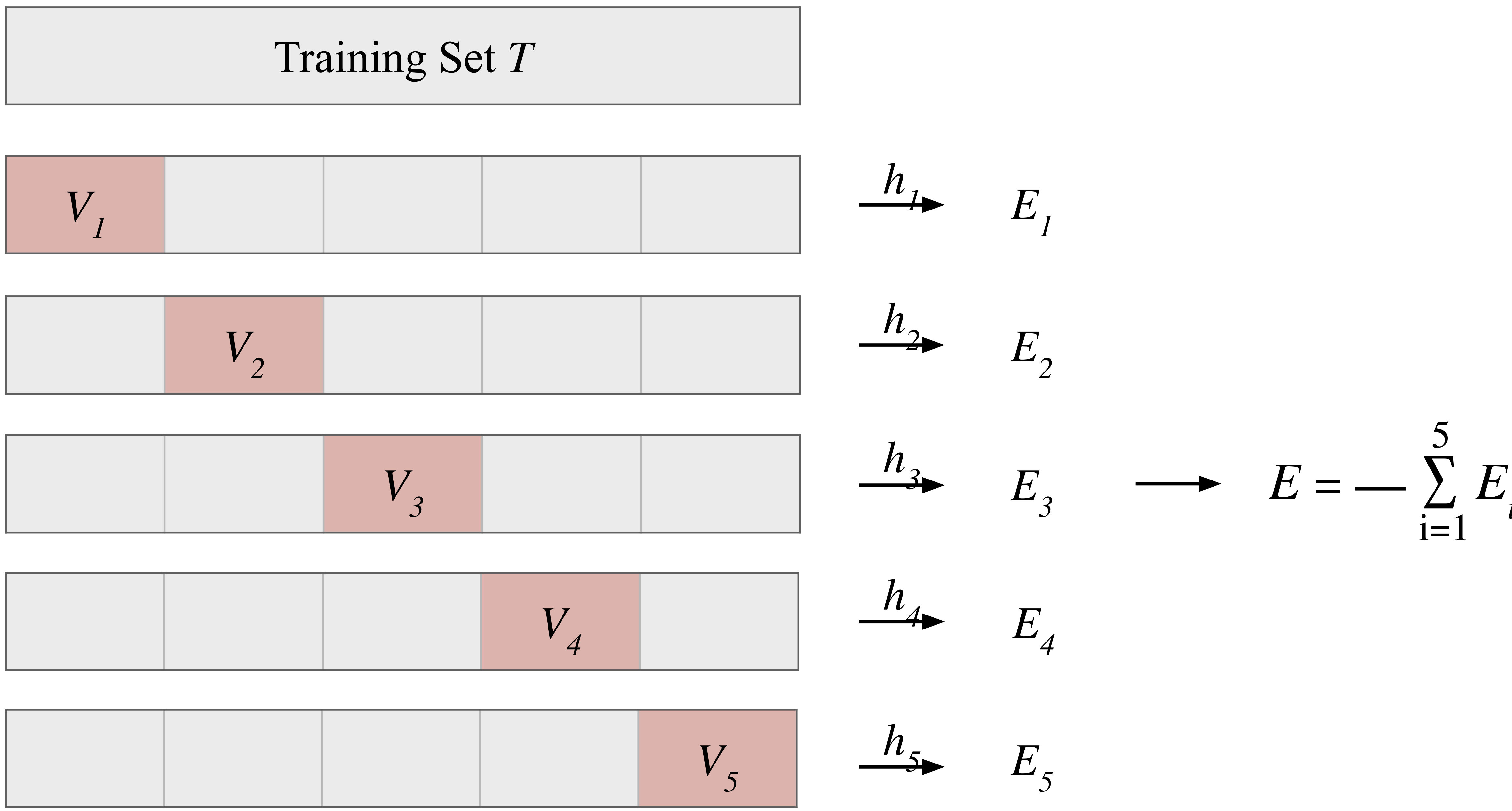


- use a KNN to solve a classification task
  - we need to find the best hyperparameter K
    - e.g., we might test K = {1, 3, 5, 9}
- We use the **training** as ground truth of known neighbours
- We select the best K with the model that perform better in the **validation**
- We estimate the true error with the **test** set

K-fold Cross Validation



K-fold Cross Validation



K-fold Cross Validation

An alternative approach to model selection (and evaluation) is the K-fold cross-validation method

K-fold CV procedure

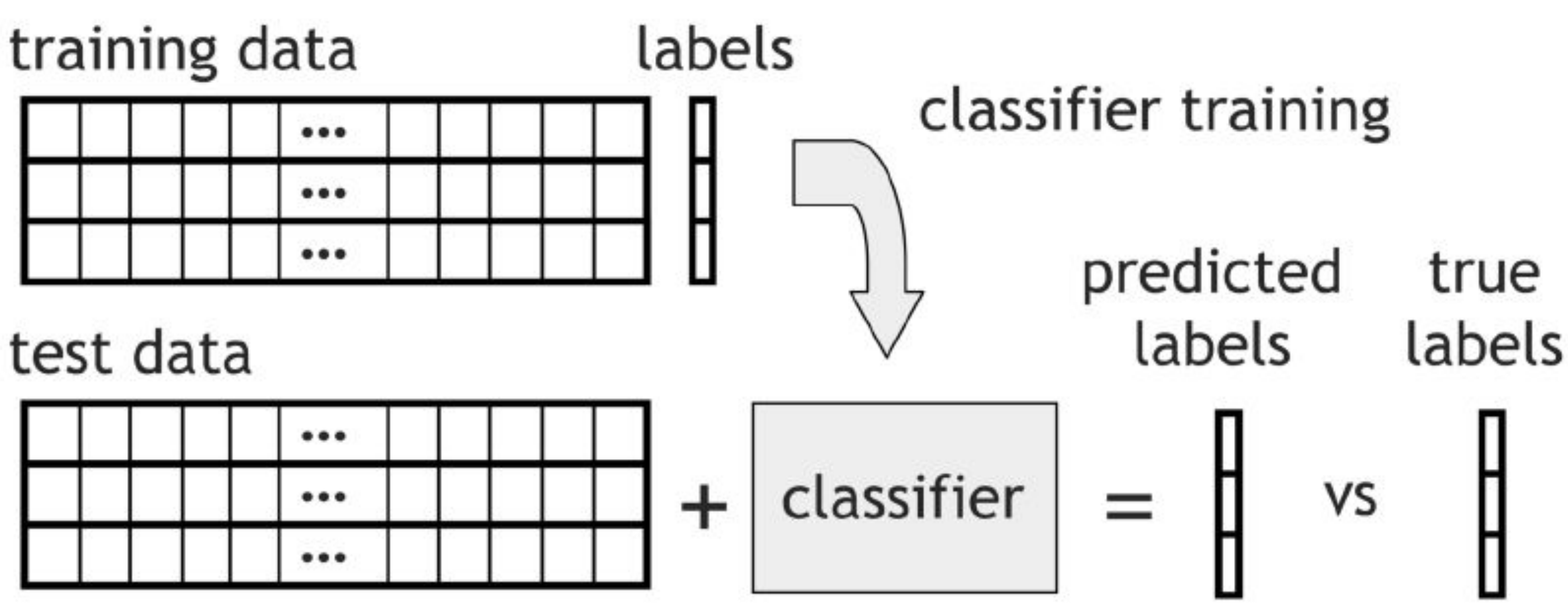
1. K different classifiers/regressors $h_1, h_2, \ldots, h_k$ are built by partitioning the initial corpus Tr into k disjoint sets $Va_1, \ldots, Va_k$ and then iteratively applying the Hold-out approach on the k-pairs $(Tr_i = Tr - Va_i, Va_i)$
2. Final error is obtained by individually computing the errors of $h_1, \ldots, h_k$, and then averaging the individual results

The above procedure is repeated for different values of the parameter(s) and the setting (model) with smaller final error is selected

The special case $k = |Tr|$ of k-fold cross-validation is called **leave-one-out** cross-validation

Model selection and error estimation

- Training phase:
  - Select the appropriate set of hyperparameters (with corresponding hypothesis space and regularization)
  - fit the models
  - Model selection: select the best model estimating its true error (without looking at test data)
- Testing phase
  - performance assessment (error estimation)

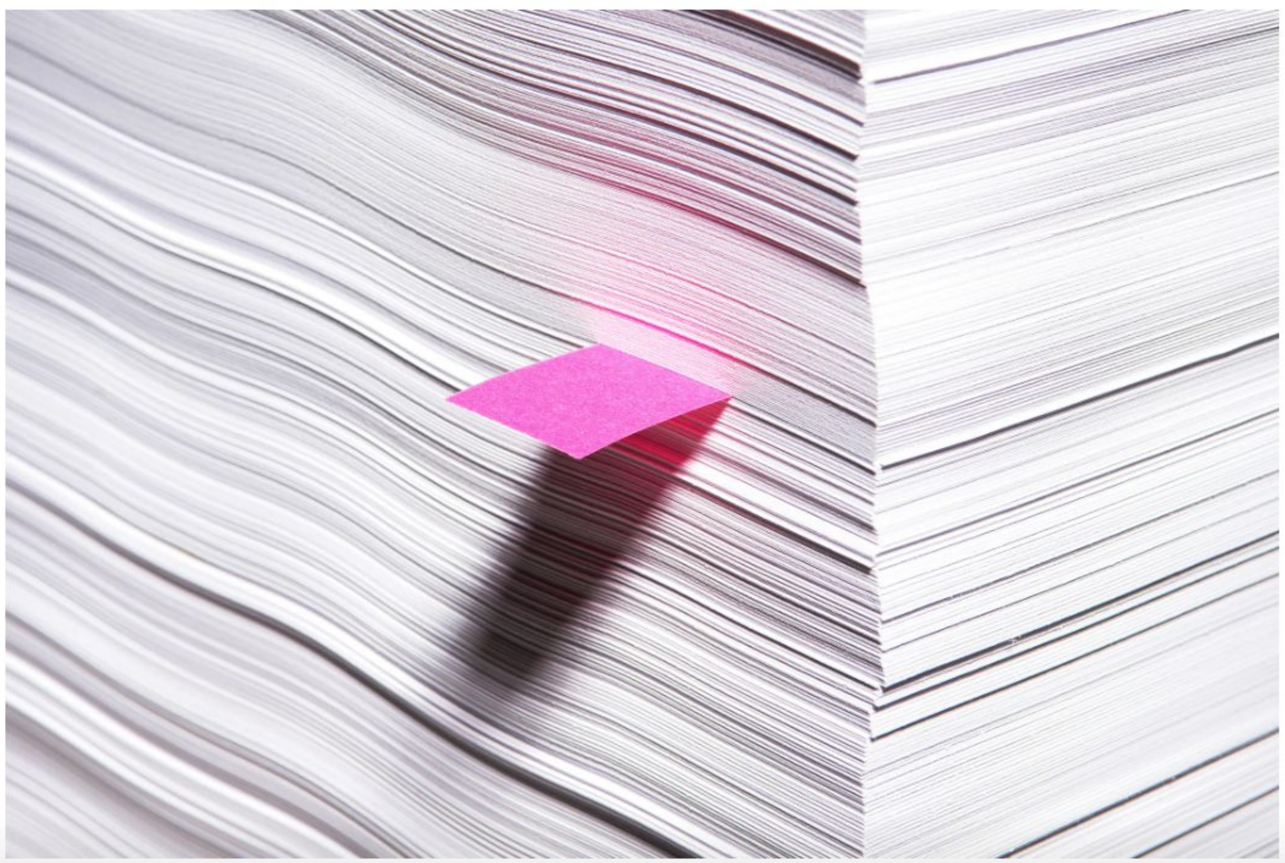


Importance of Model selection

WIRED BACKCHANNEL BUSINESS CULTURE GEAR IDEAS SCIENCE SECURITY SIGN IN SUBSCRIBE

Sloppy Use of Machine Learning Is Causing a 'Reproducibility Crisis' in Science

AI hype has researchers in fields from medicine to sociology rushing to use techniques that they don't always understand—causing a wave of spurious results.



PHOTOGRAPH: PM IMAGES/GETTY IMAGES