

Model validation: Introduction to statistical considerations

Miguel Nakamura

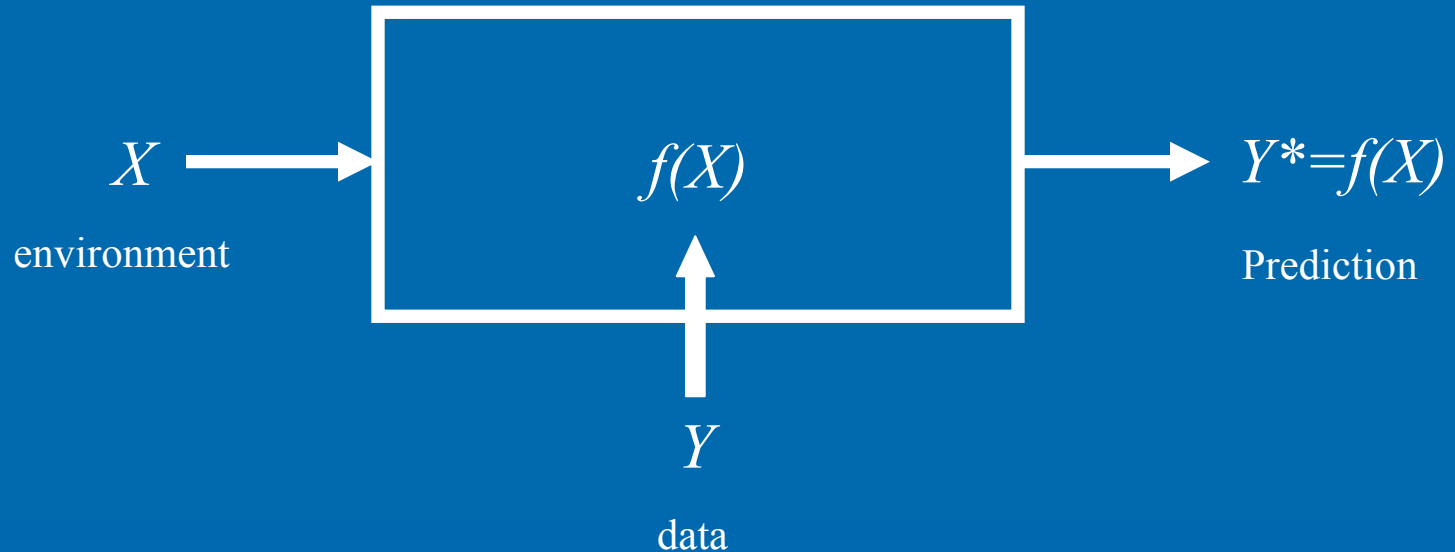
Centro de Investigación en Matemáticas (CIMAT),
Guanajuato, Mexico

nakamura@cimat.mx

Warsaw, November 2007

The slide features a solid blue background. In the lower right quadrant, there are several decorative elements consisting of concentric circles in various shades of blue, resembling ripples in water. These circles are arranged in a cluster, with some overlapping, and they vary in size and opacity.

Starting point: Algorithmic modeling culture



- Validation = examination of predictive accuracy.

Some key concepts

- Model complexity
- Loss
- Error



Model assessment: estimate *prediction error* on new data for a chosen model.

- Observed data $(X_1, Y_1), (X_2, Y_2), \dots, (X_N, Y_N)$ used to construct model, $f(X)$.
- Our intention is to use model at NEW values, X^*_1, X^*_2, \dots by using $f(X^*_1), f(X^*_2), \dots$
- Model is good if values of Y^*_1, Y^*_2, \dots are “close” to $f(X^*_1), f(X^*_2), \dots$
- Big problem: we do not know the values Y^*_1, Y^*_2, \dots (If we knew the values, we wouldn't be resorting to models!!)

Measuring error: Loss Function

$L(Y, f(X))$ is loss function that measures "closeness" between prediction $f(X)$ and observed Y .

Note: Either implicitly or explicitly, consciously or unconsciously, we specify a way of determining if a prediction is close to reality or not.

Measuring error: examples of loss functions

Squared loss: $L(Y, f(X)) = (Y - f(X))^2$

Absolute loss: $L(Y, f(X)) = |Y - f(X)|$

An asymmetric loss: $L(Y, f(X)) = \begin{cases} 2|Y - f(X)| & \text{if } Y \leq f(X) \\ |Y - f(X)| & \text{if } Y > f(X) \end{cases}$

Note: There are several ways of measuring how good a prediction is relative to reality. Which way is relevant to adopt is not a mathematical question, but rather a question for the user.

Examples of loss functions for binary outcomes

$$\text{0-1 loss: } L(Y, f(X)) = \begin{cases} 0 & \text{if } Y = f(X) \\ 1 & \text{if } Y \neq f(X) \end{cases}$$

	$f(X)=0$	$f(X)=1$
$Y=0$	0	1
$Y=1$	1	0

$$\text{Asymmetric loss: } L(Y, f(X)) = \begin{cases} 0 & \text{if } Y = f(X) \\ a & \text{if } Y > f(X) \\ b & \text{if } Y < f(X) \end{cases}$$

	$f(X)=0$	$f(X)=1$
$Y=0$	0	b
$Y=1$	a	0

Expected loss

- $f(X)$ is model obtained using data.
- Test data X^* is assumed to be coming at random.
- $L(Y^*, f(X^*))$ is random: makes sense to consider “expected loss”, or $E\{L(Y^*, f(X^*))\}$. This represents “typical” difference between Y^* and $f(X^*)$.
- $E\{L(Y^*, f(X^*))\}$ becomes a key concept for model evaluation.
- Possible criticism or warning: if expected loss is computed/estimated under an assumed distribution for X^* but the model will be used under another distribution for X^* , then expected loss may be irrelevant or misleading.

Expected 0-1 loss

- Expected loss=probability of misclassification.

$$E\{L(f(X^*), Y^*)\} = 0 \times P\{L(f(X^*), Y^*) = 0\} + 1 \times P\{L(f(X^*), Y^*) = 1\} = P\{L(f(X^*), Y^*) = 1\}$$

Expected asymmetric loss

- Expected loss = $a \times P(\text{false absence}) + b \times P(\text{false presence}) = a \times (\text{omission rate}) + b \times (\text{commission rate})$

$$E\{L(f(X^*), Y^*)\} =$$

$$0 \times P\{L(f(X^*), Y^*) = 0\} + a \times P\{L(f(X^*), Y^*) = a\} + b \times P\{L(f(X^*), Y^*) = b\} = aP\{f(X^*) = 0, Y^* = 1\} + bP\{f(X^*) = 1, Y^* = 0\}$$

The validation challenge

To compute $E\{L(Y, f(X))\}$, given that we DO NOT KNOW the value of Y .

In data modeling culture, $E\{L(Y, f(X))\}$ can be computed mathematically. (Example to follow)

In algorithmic modeling culture, $E\{L(Y, f(X))\}$ must be estimated in some way.

Example (in statistical estimation): When expected loss can be *calculated*

Object to predict is μ , mean of population.

Data is set of n observations, X_1, \dots, X_n .

Prediction of μ is sample mean (\bar{X}_n).

Loss function is square error loss: $L(\mu, \bar{X}_n) = (\bar{X}_n - \mu)^2$.

Expected Loss is called Mean Square Error.

Mathematics says: Expected Loss = $\text{Var}(\bar{X}_n) = \sigma^2 / n$.

Note: This calculation for expected loss holds even if the parameter is unknown! But this can be calculated theoretically because assumptions are being made regarding the probability distribution of observed data.

Estimating expected loss

Training error:

$$\frac{1}{N} \sum_{i=1}^N L(Y_i, f(X_i)) \text{ (average error over training sample)}$$

Unfortunately, training error is a very bad estimator of expected loss,

$$E\{L(Y, f(X))\} \text{ (average error over all values where predictions are required)}$$

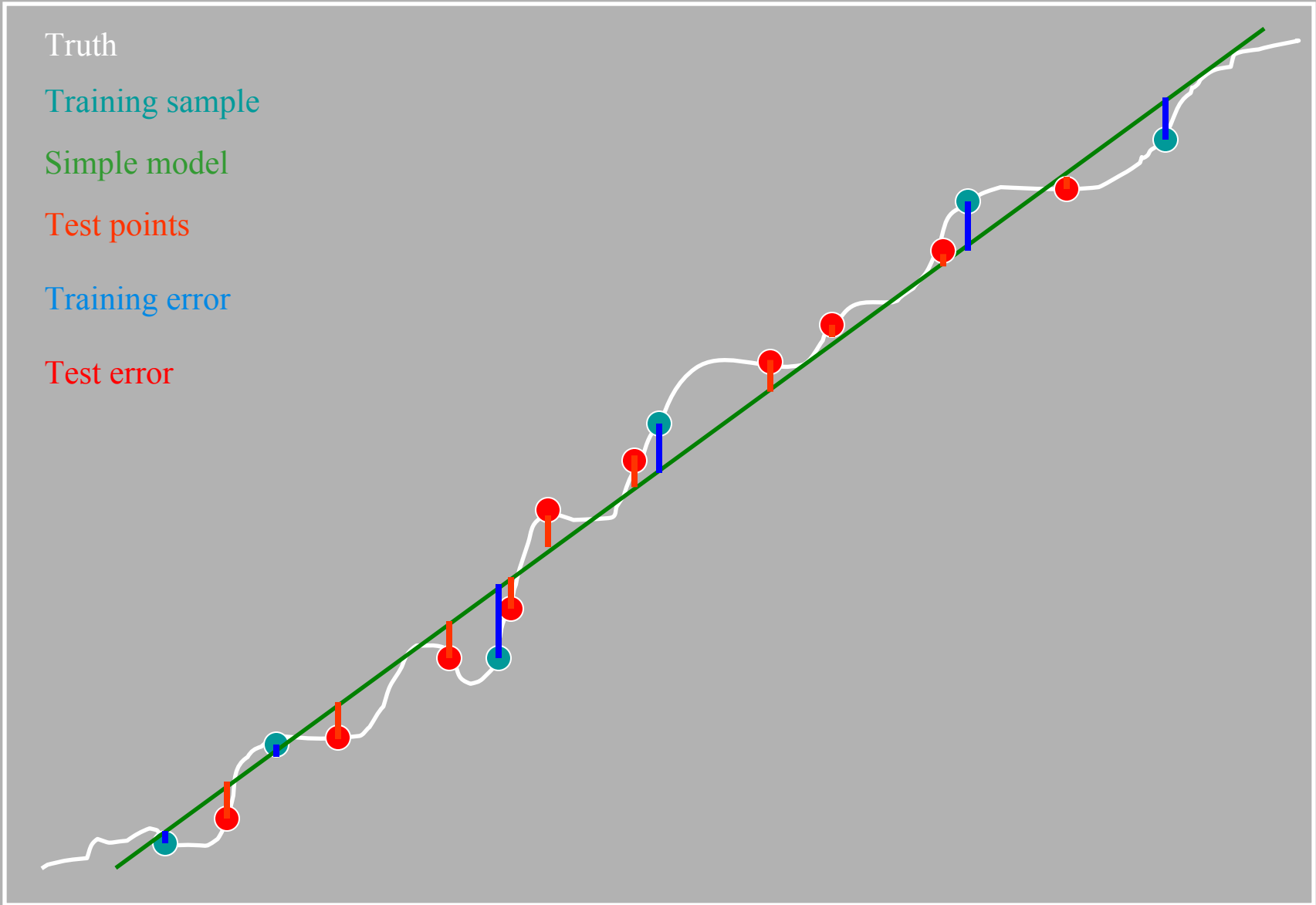
This is what we are interested in. Need independent test sample:
the notion of data-splitting is born.

$$\left\{ \begin{array}{l} (X_1, Y_1), \dots, (X_M, Y_M) \text{ used for model construction.} \\ (X_{M+1}, Y_{M+1}), \dots, (X_N, Y_N) \text{ used for estimating expected loss} \end{array} \right\}$$

Test error:

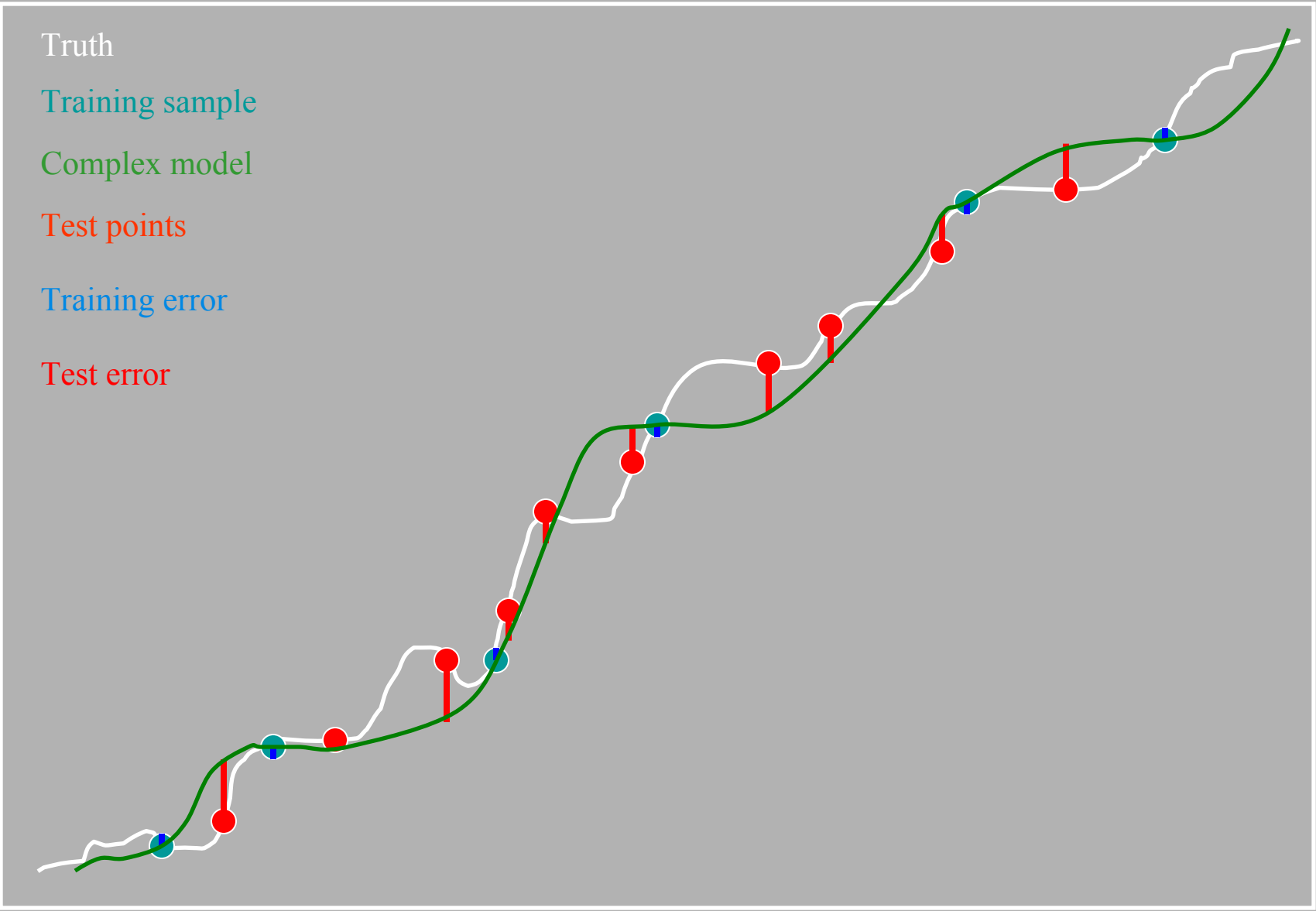
$$\frac{1}{N-M} \sum_{i=M+1}^{N-M} L(Y_i, f(X_i)) \text{ (average error over test sample)}$$

Y



X

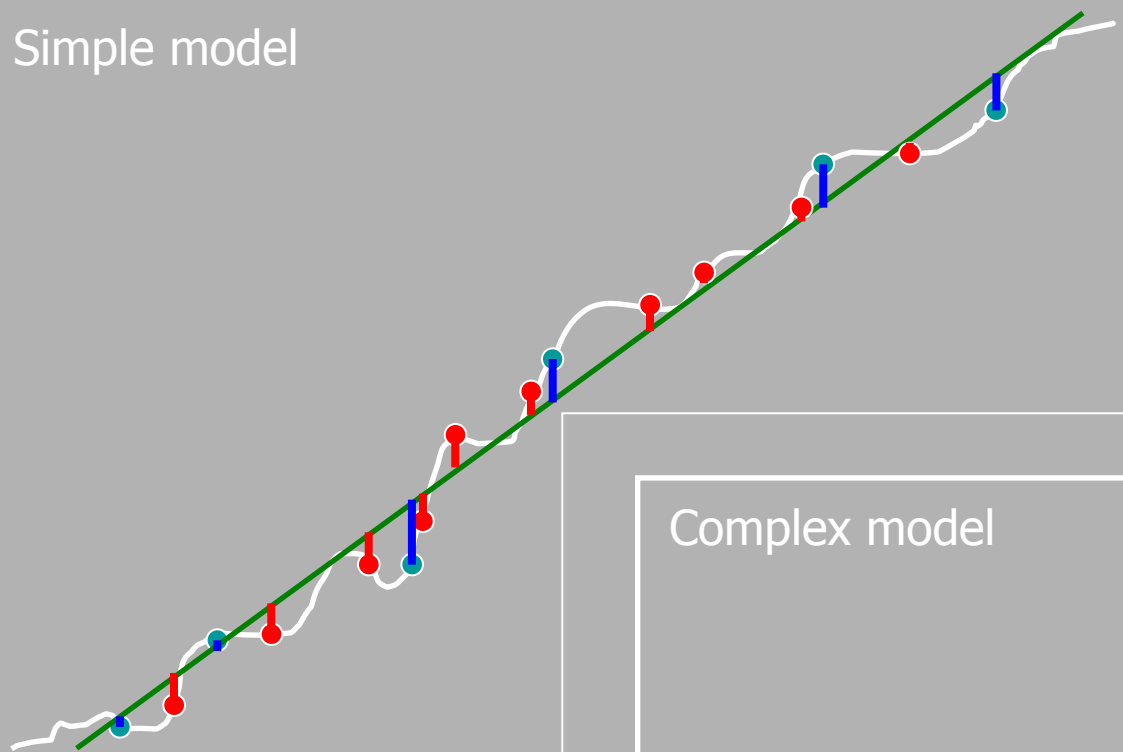
Y



X

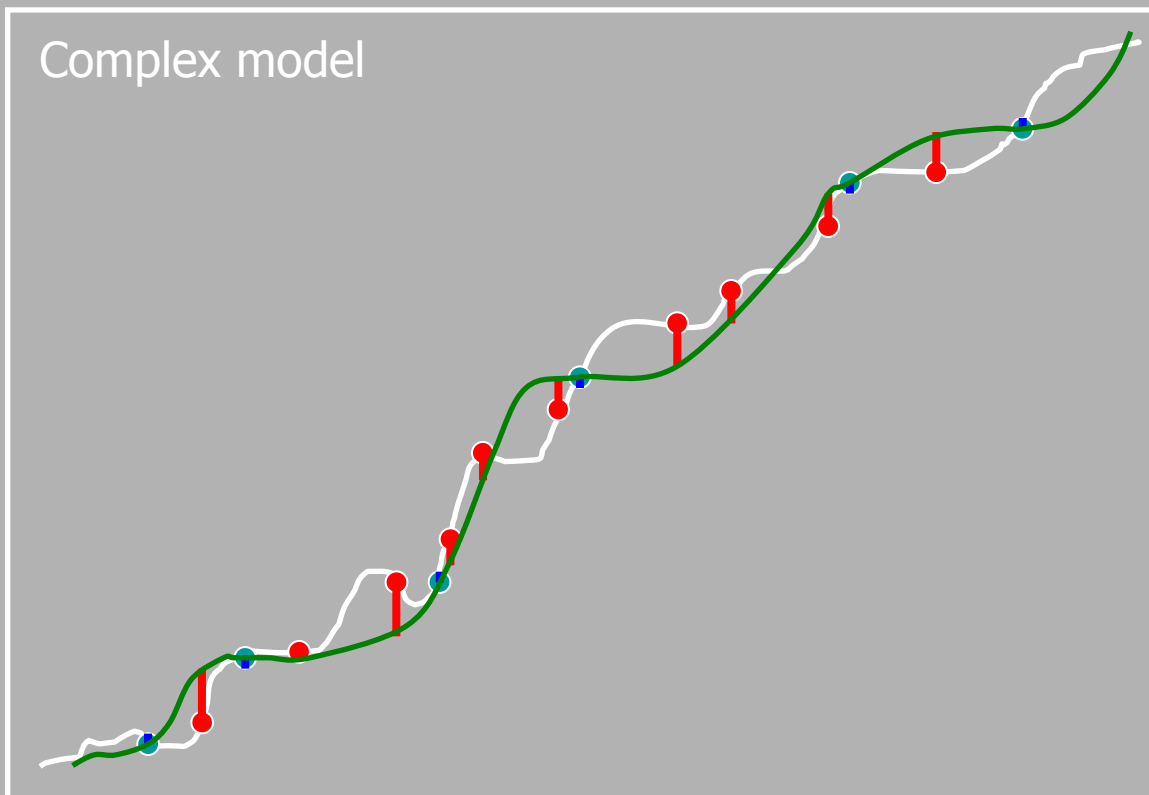
Simple model

Y



Complex model

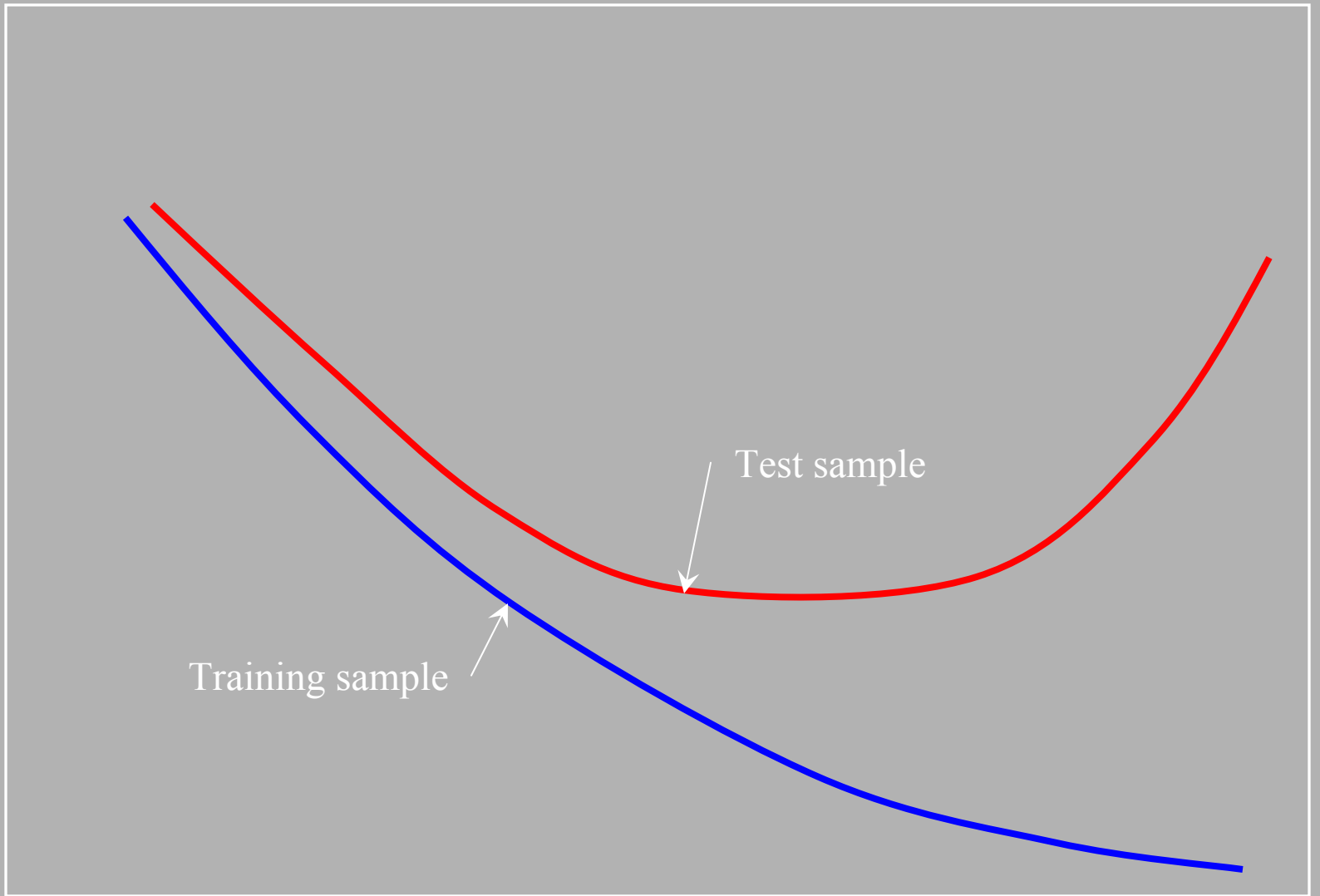
Y



Comparisons

X

Prediction Error



Training sample

Test sample

Low

Model Complexity

High

Model complexity

- In Garp: number of layers and/or convergence criteria
- In Maxent: it is the regularization parameter, β .



Things could get worse in preceding plots:

- Sampling bias may induce artificially smaller or larger errors.
- Randomness around “truth” may be present in observations, due to measurement error or other issues.
- X is multidimensional (as in niche models).
- Predictions may be needed where observed values of X are scarce.
- $f(X)$ itself may be random for some algorithms (same X yields different $f(X)$).

Notes regarding Loss Function

- Is any loss function universal, *i.e.* that it is reasonable for all problems and all type of applications?
- Once loss function is fixed, the notion of optimality follows!
- There is no such thing as an “optimal” method. It is only optimal relative to the given loss function. Thus it is optimal for the type of problems for which that particular loss function is appropriate.

General methods for estimating prediction error*

- Cross-Validation
- Bootstrap

*Others exist, but require structural knowledge of $f(X)$ and case-by-case analytical work. See Hastie, *et al.* (2001).

Cross-validation and bootstrap: Main Idea

- Assume prediction is required for situations directly similar to those applying to data X, Y . The idea is to use this data to represent future test sets as well.
- In data-rich situation, set aside validation sets (use one part of data set to fit model, second part for estimating prediction error for model selection, third part for assessing prediction error of final selected model).
- If data scarce, must resort to “artificially produced” validation sets.

Cross-Validation

Split data randomly into K roughly equal-sized parts. Take turns using each part as a test set and the other $K - 1$ parts for training the model.

Compute prediction error each time and save.

How large should K be? Somewhere between 5 and 10.



$N=200, K=5$

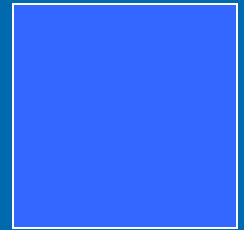
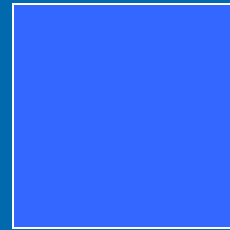
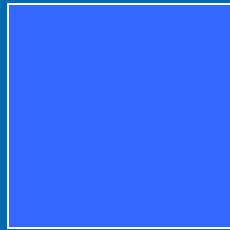
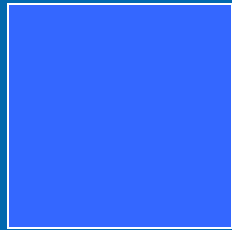
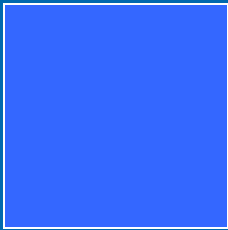
1

2

3

4

5



1

2

3

4

5



$$\begin{matrix} L(Y_1, f_1(X_1)) \\ \text{A} \\ L(Y_{40}, f_1(X_{40})) \end{matrix}$$

Note “non-overlapping” training and test sets.

1

2

3

4

5

Train

Test

Train

Train

Train



$L(Y_1, f_1(X_1))$
A
 $L(Y_{40}, f_1(X_{40}))$

$L(Y_{41}, f_2(X_{41}))$
A
 $L(Y_{80}, f_2(X_{80}))$

1

2

3

4

5

Train

Train

Test

Train

Train



$$\begin{matrix} L(Y_1, f_1(X_1)) \\ \text{A} \\ L(Y_{40}, f_1(X_{40})) \end{matrix}$$

$$\begin{matrix} L(Y_{41}, f_2(X_{41})) \\ \text{A} \\ L(Y_{80}, f_2(X_{80})) \end{matrix}$$

$$\begin{matrix} L(Y_{81}, f_3(X_{81})) \\ \text{A} \\ L(Y_{120}, f_3(X_{120})) \end{matrix}$$



1

2

3

4

5

Train

Train

Train

Test

Train



$$\begin{matrix} L(Y_1, f_1(X_1)) \\ \text{A} \\ L(Y_{40}, f_1(X_{40})) \end{matrix}$$

$$\begin{matrix} L(Y_{41}, f_2(X_{41})) \\ \text{A} \\ L(Y_{80}, f_2(X_{80})) \end{matrix}$$

$$\begin{matrix} L(Y_{81}, f_3(X_{81})) \\ \text{A} \\ L(Y_{120}, f_3(X_{120})) \end{matrix}$$

$$\begin{matrix} L(Y_{121}, f_4(X_{121})) \\ \text{A} \\ L(Y_{160}, f_4(X_{160})) \end{matrix}$$

1

2

3

4

5

Train

Train

Train

Train

Test

$$L(Y_1, f_1(X_1))$$

A

$$L(Y_{40}, f_1(X_{40}))$$

$$L(Y_{41}, f_2(X_{41}))$$

A

$$L(Y_{80}, f_2(X_{80}))$$

$$L(Y_{81}, f_3(X_{81}))$$

A

$$L(Y_{120}, f_3(X_{120}))$$

$$L(Y_{121}, f_4(X_{121}))$$

A

$$L(Y_{160}, f_4(X_{160}))$$

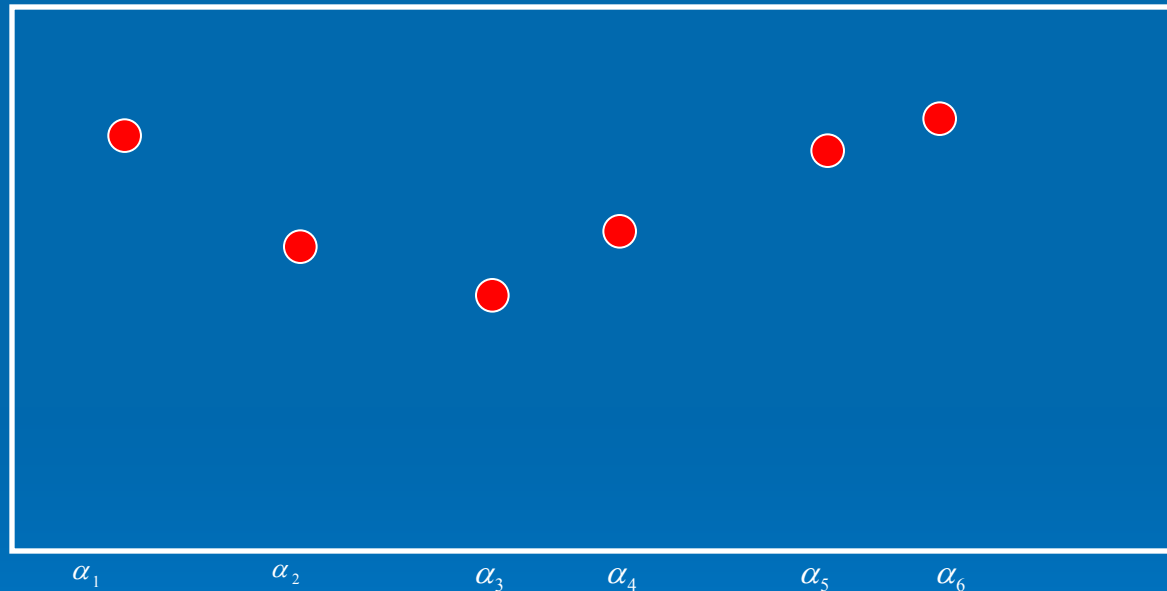
$$L(Y_{181}, f_5(X_{181}))$$

A

$$L(Y_{200}, f_5(X_{200}))$$

Cross-validation estimate of prediction error is average of these.

If a method has a "tuning parameter" related to complexity, α , then this can be repeated for different values of α , in order to estimate the error curve:



Chose value of α that minimizes this,
and then use for final model.

Bootstrap

Let $\{(X_1, Y_1), \dots, (X_N, Y_N)\}$ be the training set.

A bootstrap sample is a sample of size N created from random drawings with replacement from the original training set.

Example: $N = 5$

N random drawings with replacement may produce the bootstrap sample

$(X_4, Y_4), (X_1, Y_1), (X_1, Y_1), (X_3, Y_3), (X_4, Y_4)$

Notice there can be repeats in a sample, and samples that don't contain an original point.

Generate a large number (B of them) of bootstrap samples:

$$S_1, S_2, S_3, \dots, S_B$$

For each bootstrap sample, compute loss $L(Y_i, f^b(X_i))$, where $f^b(X_i)$ is the predicted value of Y_i using the b -th bootstrap sample.

original observation #	bootstrap samples that do NOT contain observation	average loss
1	$S_{303}, S_{272}, \dots, S_{23}$	A_1
2	$S_{109}, S_3, \dots, S_{526}$	A_2
A	A	A
N	$S_{204}, S_{97}, \dots, S_{45}$	A_N
		$\frac{1}{N} \sum_{i=1}^N A_i$



This is bootstrap estimate of prediction error.

“No free lunch theorem”

- If a method performs well (higher than average generalization accuracy) over a set of problems, then it **MUST** perform worse than average elsewhere.
- No method can perform well throughout all of nature’s black boxes.

Validation in “data modeling culture” (just to get flavor of things, for contrast)

- Requirements for a statistical model:
 - Establish link with background knowledge.
 - Set up connection with previous work.
 - Give some pointer toward generating process.
 - Have parameters with clear subject-matter interpretations.
 - Specify haphazard aspects well enough to lead to meaningful assessment of precision*.
 - Check that fit is adequate.

* To be exemplified shortly.

Cox and Wermuth (1996)

A note on thresholding

“Continuous output” model, $f(X)$.

For each value of the threshold, t , a functionally different “classification model” is obtained. Let’s call it $f_t(X)$.

A loss function for binary outcomes is decided upon,

$L(Y, f_t(X))$. We thus have an expected loss for each t . Let us call this Err_t .

Averaging values of Err_t across values of t is an “overall” notion of expected loss for the original model, $f(X)$.

Note: AUC is akin to an “overall average loss” for the algorithm.

Some references

- Hastie, T., Tibshirani, R., and Friedman, J. (2001), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, New York.
- Duda, R.O., Hart, P.E., and Stork, D.G. (2001), *Pattern Classification*, Wiley, New York.
- Wahba, G. (1990), *Spline Models for Observational Data*, SIAM, Philadelphia.
- Stone, M. (1974), “Cross-validatory choice and assessment of statistical predictions”, *Journal of the Royal Statistical Society*, 36, 111–147.
- Breiman, L., and Spector, P. (1992), “Submodel selection and evaluation in regression: the X-random case”, *The International Statistics Review*, 60, 291–319.
- Efron, B. (1986), “How biased is the apparent error rate of a prediction rule?”, *Journal of the American Statistical Association*, 81, 461–470.