Model Selection

Topics

- Bias variance trade-off
- Optimism of training error
- Estimates of in sample prediction error
- BIC
- VC dimension
- Cross-validation (chapter 3), bootstrap

Definitions

- Loss functions

$$\begin{aligned} L(Y, \hat{f}(X)) &= (Y - \hat{f}(X))^2 \quad \text{squared error,} \\ L(G, \hat{G}(X)) &= I(G \neq \hat{G}(X)) \quad \text{0-1 loss,} \\ L(G, \hat{p}(X)) &= -2 \sum_{k=1}^{K} I(G = k) \log \hat{p}_k(X) \\ &= -2 \log \hat{p}_G(X) \quad \text{log-likelihood.} \end{aligned}$$

- Training error (over training set $\mathcal{T}$):

$$\overline{\text{err}} = \frac{1}{N} \sum_{i=1}^{N} L(y_i, \hat{f}(x_i)).$$

- Test Error (Generalization Error):

$$\text{Err}_{\mathcal{T}} = \text{E}[L(Y, \hat{f}(X))|\mathcal{T}] \quad (\text{E}[L(Y, \hat{G}(X))|\mathcal{T}], \quad \text{E}[L(Y, \hat{p}(X))|\mathcal{T}])$$

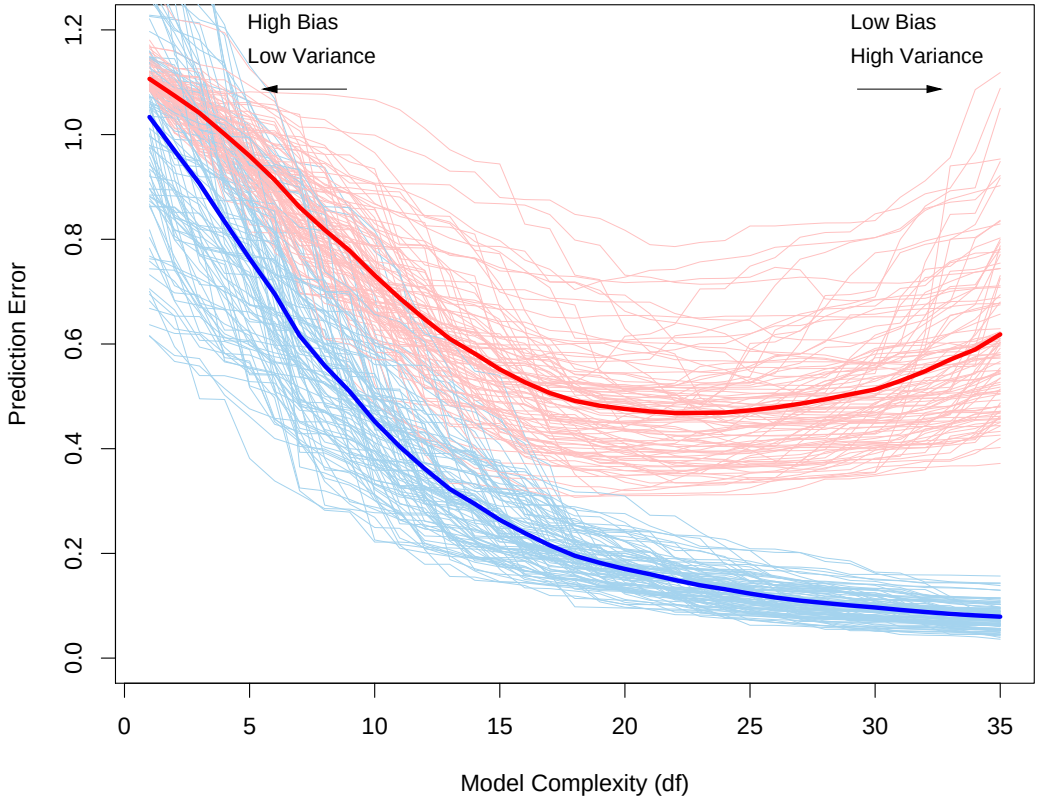
  This is the expected loss over random realizations of new data test data.

- Expected test error (expected prediction error, expected generalization error)

$$\text{Err} = \text{E}[L(Y, \hat{f}(X))] = \text{E}[\text{Err}_{\mathcal{T}}]$$

$\text{Err}_{\mathcal{T}}$ is the error we can expect if we use the function $\hat{f}(x)$ trained on our particular training set $\mathcal{T}$ to make our predictions.

Err averages in addition over all training sets. Since $\text{Err}_{\mathcal{T}}$ will depend on the particular nuances of our training set, we can make more general statements about Err than we can about $\text{Err}_{\mathcal{T}}$.



Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error $\overline{\text{err}}$, while the light red curves show the conditional test error $\text{Err}_{\mathcal{T}}$ for 100 training sets of size 50 each, as the model complexity is increased. The solid curves show the expected test error Err and the expected training error $\text{E}[\overline{\text{err}}]$.

Bias-variance decomposition

$$Y = f(X) + \varepsilon$$

$$\begin{aligned}
\text{Err}(x_0) &= E[(Y - \hat{f}(x_0))^2 | X = x_0] \\
&= \sigma_\varepsilon^2 + [E\hat{f}(x_0) - f(x_0)]^2 + E[\hat{f}(x_0) - E\hat{f}(x_0)]^2 \\
&= \sigma_\varepsilon^2 + \text{Bias}^2(\hat{f}(x_0)) + \text{Var}(\hat{f}(x_0)) \\
&= \text{Irreducible Error} + \text{Bias}^2 + \text{Variance}.
\end{aligned}$$

In the above, we need to decide whether x_i in the sample are random, or assumed fixed. We assume fixed for what follows (recall homework 1).

For K-nearest neighbors:

$$\begin{aligned}\mathrm{Err}(x_0) &= E[(Y-\hat{f}_k(x_0))^2|X=x_0] \\ &= \sigma_\varepsilon^2 + \Big[f(x_0) - \frac{1}{k}\sum_{\ell=1}^{k} f(x_{(\ell)})\Big]^2 + \sigma_\varepsilon^2/k.\end{aligned}$$

For linear regression:

$$\hat{f}_p(x_0) = x_0^T(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{h}(x_0)^T\mathbf{y}$$

$$\begin{aligned}\mathrm{Err}(x_0) &= E[(Y-\hat{f}_p(x_0))^2|X=x_0] \\ &= \sigma_\varepsilon^2 + [f(x_0) - E\hat{f}_p(x_0)]^2 + ||\mathbf{h}(x_0)||^2\sigma_\varepsilon^2.\end{aligned}$$

$$\frac{1}{N}\sum_{i=1}^{N}\mathrm{Err}(x_i) = \sigma_\varepsilon^2 + \frac{1}{N}\sum_{i=1}^{N}[f(x_i) - E\hat{f}(x_i)]^2 + \frac{p}{N}\sigma_\varepsilon^2,$$

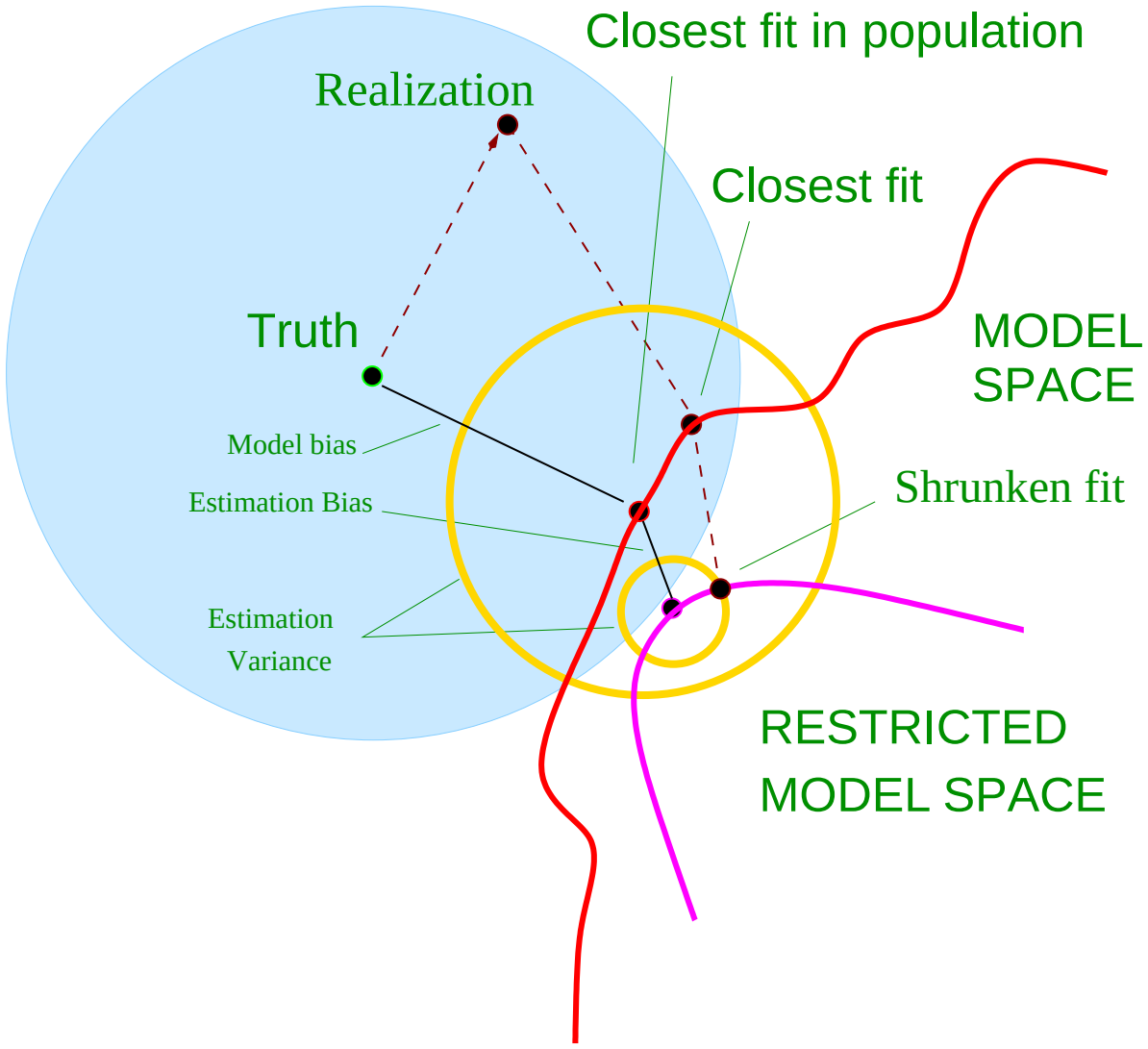
Classification and 0-1 loss

- Bias and variance and do not add as they do for squared error: variance tends to dominate, while bias is tolerable as long as you are on the correct side of the decision boundary. Hence biased methods often do well!
- Friedman (1996) "On Bias, Variance 0-1 loss..." shows

$$Pr(\hat{G}(x_0) \neq G(x_0)) \approx \Phi\left[\frac{\text{sign}(1/2 - f(x_0))\left(E\hat{f}(x_0) - 1/2\right)}{\sqrt{\text{Var}(\hat{f}(x_0))}}\right]$$

  where $G(x_0) = I(f(x_0) > \frac{1}{2})$ is the Bayes classifier (Exercise 7.2)
- Hence on the wrong side of the decision boundary, *increasing* the variance can help

Bias-variance schematic The model space is the set of all possible predictions from the model, with the "closest fit" labeled with a black dot. The model bias from the truth is shown, along with the variance, indicated by the large yellow circle centered at the black dot labelled "closest fit in population". A shrunken or regularized fit is also shown, having additional estimation bias, but smaller prediction error due to its decreased variance.



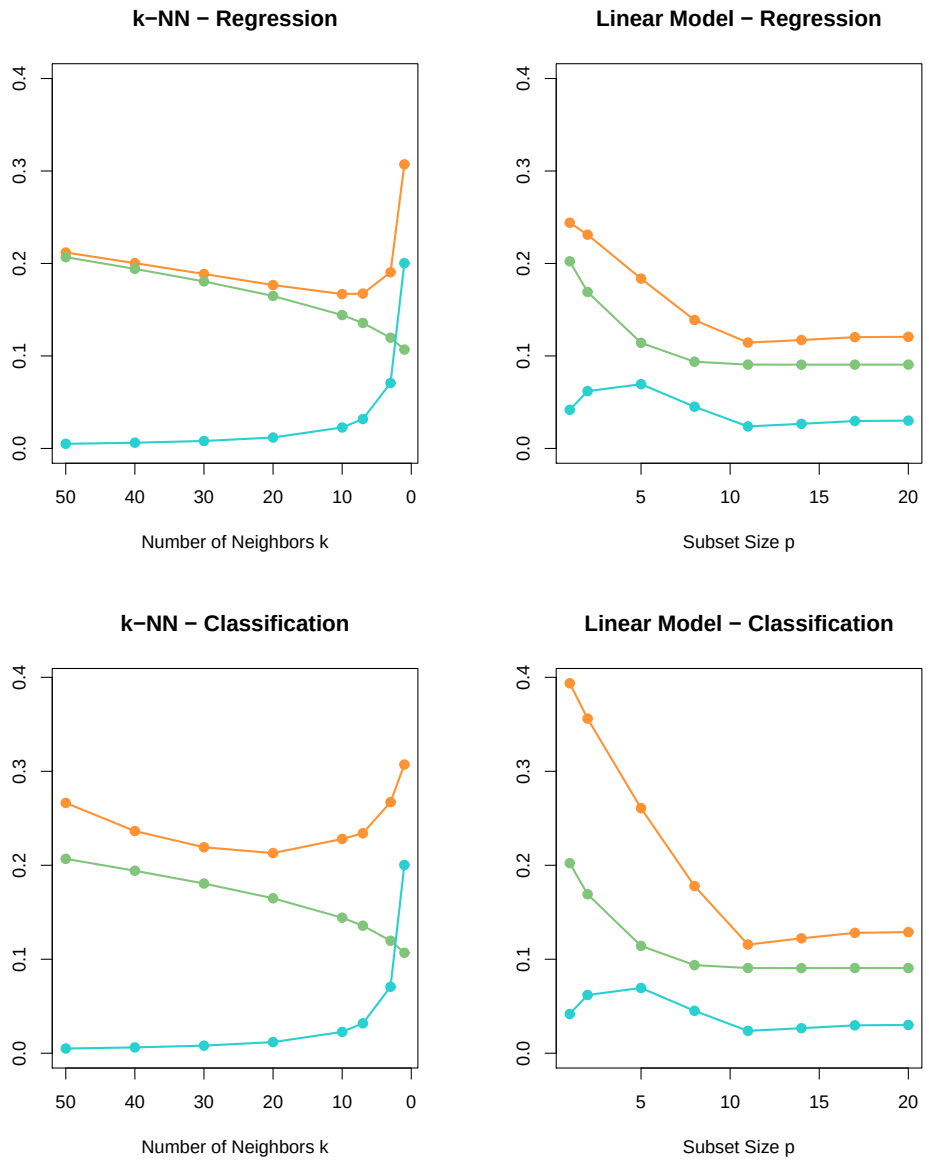
Simulation Study (next slide)

There are 50 observations and 20 predictors, uniformly distributed in the hypercube $[0,1]^{20}$. The situations are as follows:

Left panels: Y is 1 if $X_1 \geq 1/2$ and 0 otherwise, and we use k-nearest neighbors.

Right panels: Y is 1 if $\frac{1}{10}\sum_{j=1}^{10} X_j \geq 1/2$ and 0 otherwise, and we use best subset linear regression indexed by subset size p.

There are 100 realizations of this simulation, and the same fixed test set of size 10,000 is used for each.



Prediction error, squared bias and variance for a simulated example. The top row is regression with squared error loss; the bottom row is classification with 0–1 loss. The models are k-nearest neighbors (left) and best subset regression of size p (right). The variance and bias curves are the same in regression and classification, but the prediction error curve is different.

Optimism of the Training Error

- training error

$$\overline{\text{err}} = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}(x_i))$$

- In-sample error

$$\text{Err}_{\text{in}} = \frac{1}{N}\sum_{i=1}^{N} \text{E}_{Y^0}[L(Y_i^0, \hat{f}(x_i))|\mathcal{T}]$$

  where at each $x_i \in \mathcal{T}$, we observe a new Y_i^0.

- Optimism

$$\text{op} \equiv \text{Err}_{\text{in}} - \overline{\text{err}}.$$

  If we can estimate op, then we can estimate

$$\text{Err}_{\text{in}} = \overline{\text{err}} + \widehat{\text{op}}.$$

- For squared error, 0–1, and other loss functions, one can show quite generally that

$$\omega = E_{\mathbf{y}}(\text{op}) = \frac{2}{N}\sum_{i=1}^{N}\text{Cov}(\hat{y}_i, y_i),$$

  where $E_{\mathbf{y}}$ takes expectation wrt the $y_i \in \mathcal{T}$, but the $x_i \in \mathcal{T}$ are held fixed.

- For linear model fitting (with d coefficients):

$$\sum_{i=1}^{N}\text{Cov}(\hat{y}_i, y_i) = d\sigma_\varepsilon^2$$

  for the additive error model $Y = f(X) + \varepsilon$, and so

$$\widehat{\text{Err}}_{\text{in}} = \overline{\text{err}} + 2 \cdot \frac{d}{N}\sigma_\varepsilon^2$$

has the property that

$$\mathrm{E}_{\mathbf{y}}\widehat{\mathrm{Err}}_{\mathrm{in}} = \mathrm{E}_{\mathbf{y}}\mathrm{Err}_{\mathrm{in}}$$

- Cp and AIC statistics:

$$C_p = \overline{\mathrm{err}} + 2 \cdot \frac{d}{N}\hat{\sigma_\varepsilon}^2.$$

$$\mathrm{AIC} = -\frac{2}{N} \cdot \mathrm{loglik} + 2 \cdot \frac{d}{N}$$

where $\mathrm{loglik} = \sum_{i=1}^{N} \log \Pr_{\hat{\theta}}(y_i)$ is the observed log-likelihood and $\hat{\theta}$ is the MLE of the parameter vector θ.

Example: Cp for linear operators

- Assume $y_i = f(x_i) + \epsilon_i, \quad \epsilon_i \sim (0, \sigma_\epsilon^2)$.
- Often the fitted vector $\hat{\mathbf{f}}$ is linear in $\mathbf{y}$:

$$\hat{\mathbf{f}} = \mathbf{S}\mathbf{y}$$

  e.g. linear regression, ridge regression, cubic smoothing splines.

- $$N \cdot \mathrm{E}_{\mathbf{y}}\mathrm{Err}_{in} = N \cdot \sigma_\epsilon^2 + \sum_{i=1}^{N}[f(x_i) - \{\mathbf{S}\mathbf{f}\}_i]^2 + \sigma_\epsilon^2 \mathrm{tr}(\mathbf{S}^T\mathbf{S})$$

- $$\begin{aligned} N \cdot \mathrm{E}_{\mathbf{y}}(\overline{\mathrm{err}}) &= \mathrm{E}||(\mathbf{I} - \mathbf{S})\mathbf{y}||^2 \\ &= \mathbf{f}^T(\mathbf{I} - \mathbf{S})^T(\mathbf{I} - \mathbf{S})\mathbf{f} + \mathrm{E}_{\boldsymbol{\epsilon}}(\boldsymbol{\epsilon}^T(\mathbf{I} - \mathbf{S})^T(\mathbf{I} - \mathbf{S})\boldsymbol{\epsilon}) \\ &= \mathrm{Bias}^2 + \sigma_\epsilon^2 \mathrm{tr}(\mathbf{I} - 2\mathbf{S} + \mathbf{S}^T\mathbf{S}) \end{aligned}$$

- Hence

$$\mathrm{E}_{\mathbf{y}}\mathrm{Err}_{in} - \mathrm{E}_{\mathbf{y}}(\overline{\mathrm{err}}) = \frac{2}{N}\sigma_\epsilon^2 \mathrm{tr}(\mathbf{S})$$

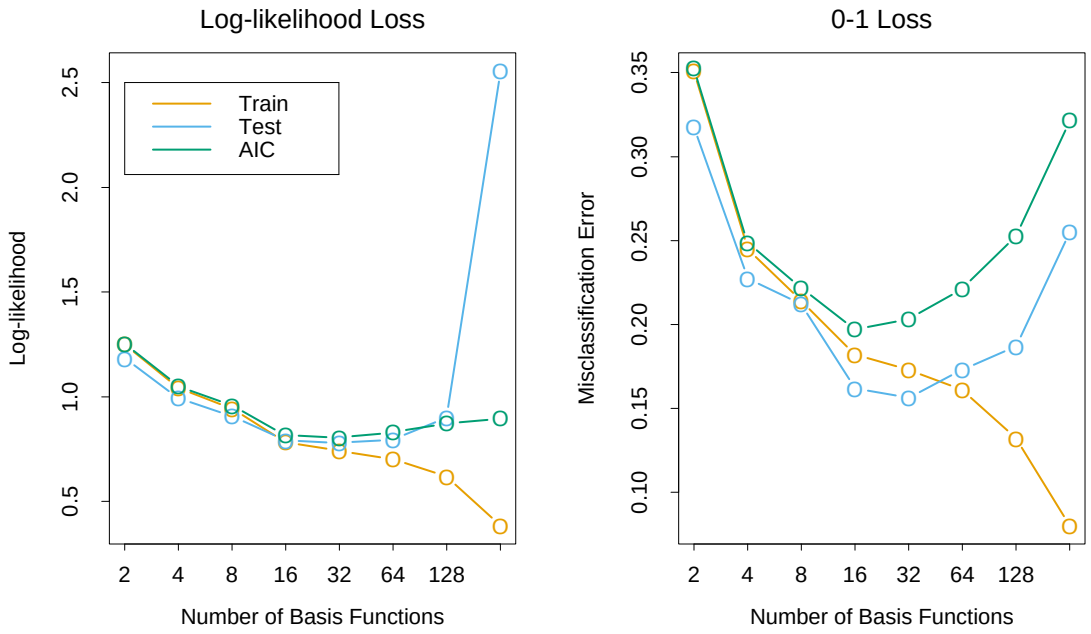
- But this is $\omega = \mathrm{E}_{\mathbf{y}}\mathrm{op}$, and

$$\mathrm{Cov}(\mathbf{y}, \hat{\mathbf{f}}) = \mathrm{Cov}(\mathbf{y}, \mathbf{S}\mathbf{y}) = \sigma_\epsilon^2 \mathbf{S}$$

- Based on the above, we define the *effective degrees of freedom*

$$\mathrm{df} = \frac{\sum_{i=1}^N \mathrm{Cov}(\hat{y}_i, y_i)}{\sigma_\epsilon^2}$$

  For fits from linear operators like above $\mathrm{df} = \mathrm{tr}(\mathbf{S})$.



Phoneme recognition example. The logistic regression coefficient function $\beta(f) = \sum_{m=1}^{M} h_m(f)\theta_m$ is modeled in M spline basis functions.

Left panel: AIC statistic used to estimate $\mathrm{Err}_{\mathrm{in}}$ using log-likelihood loss. Included is an estimate of Err based on a test sample. It does well except for the overparametrized case ($M = 256$ parameters for $N = 1000$ observations).

Right panel: Same is done for 0–1 loss. Although the AIC formula does not strictly apply here, it does a reasonable job in this case.

BIC— Bayesian information criterion

$$\text{BIC} = -2 \cdot \text{loglik} + (\log N) \cdot d$$

- under Guassian model σ_ε^2 is known, $-2 \cdot \text{loglik}$ equals (up to a constant) $\sum_i (y_i - \hat{f}(x_i))^2/\sigma_\varepsilon^2$, which is $N \cdot \overline{\text{err}}/\sigma_\varepsilon^2$ for squared error loss. Hence we can write

$$\text{BIC} = \frac{N}{\sigma_\varepsilon^2}\left[\overline{\text{err}} + (\log N) \cdot \frac{d}{N}\sigma_\varepsilon^2\right].$$

  Hence BIC is proportional to AIC, except 2 is replaced by $\log N$.

- candidate models $\mathcal{M}_m, m = 1, \ldots, M$, with prior distribution $\Pr(\theta_m|\mathcal{M}_m)$

- Given data $\mathbf{Z}$, the posterior probability of a given model is

$$\begin{aligned}\Pr(\mathcal{M}_m|\mathbf{Z}) &\propto \Pr(\mathcal{M}_m) \cdot \Pr(\mathbf{Z}|\mathcal{M}_m) \\ &\propto \Pr(\mathcal{M}_m) \cdot \int \Pr(\mathbf{Z}|\theta_m, \mathcal{M}_m)\Pr(\theta_m|\mathcal{M}_m)d\theta_m,\end{aligned}$$

where $\mathbf{Z}$ represents the training data $\{x_i, y_i\}_1^N$.

- posterior odds

$$\frac{\Pr(\mathcal{M}_m|\mathbf{Z})}{\Pr(\mathcal{M}_\ell|\mathbf{Z})} = \frac{\Pr(\mathcal{M}_m)}{\Pr(\mathcal{M}_\ell)} \cdot \frac{\Pr(\mathbf{Z}|\mathcal{M}_m)}{\Pr(\mathbf{Z}|\mathcal{M}_\ell)}.$$

- The rightmost quantity

$$\text{BF}(\mathbf{Z}) = \frac{\Pr(\mathbf{Z}|\mathcal{M}_m)}{\Pr(\mathbf{Z}|\mathcal{M}_\ell)}$$

is called the *Bayes factor*

- Typically we assume that the prior over models is uniform, so that $\Pr(\mathcal{M}_m)$ is constant.
- A Laplace approximation to the integral gives

$$\log \Pr(\mathbf{Z}|\mathcal{M}_m) = \log \Pr(\mathbf{Z}|\hat{\theta}_m, \mathcal{M}_m) - \frac{d_m}{2} \cdot \log N + O(1).$$

  $\hat{\theta}_m$ is a maximum likelihood estimate and d_m is the number of free parameters
- If we define our loss function to be

$$-2 \log \Pr(\mathbf{Z}|\hat{\theta}_m, \mathcal{M}_m),$$

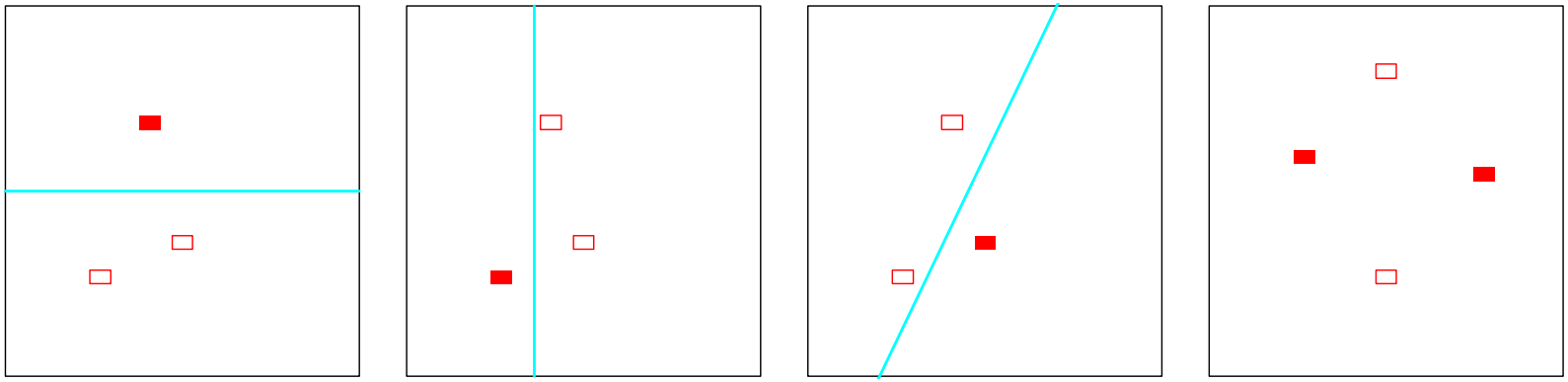
  this is equivalent to the BIC criterion.
- $\Pr(\mathcal{M}_m|\mathbf{Z}) \propto \Pr(\mathcal{M}_m) e^{-\frac{1}{2}\text{BIC}_m}$.

VC dimension (Vapnik-Chervonenkis)

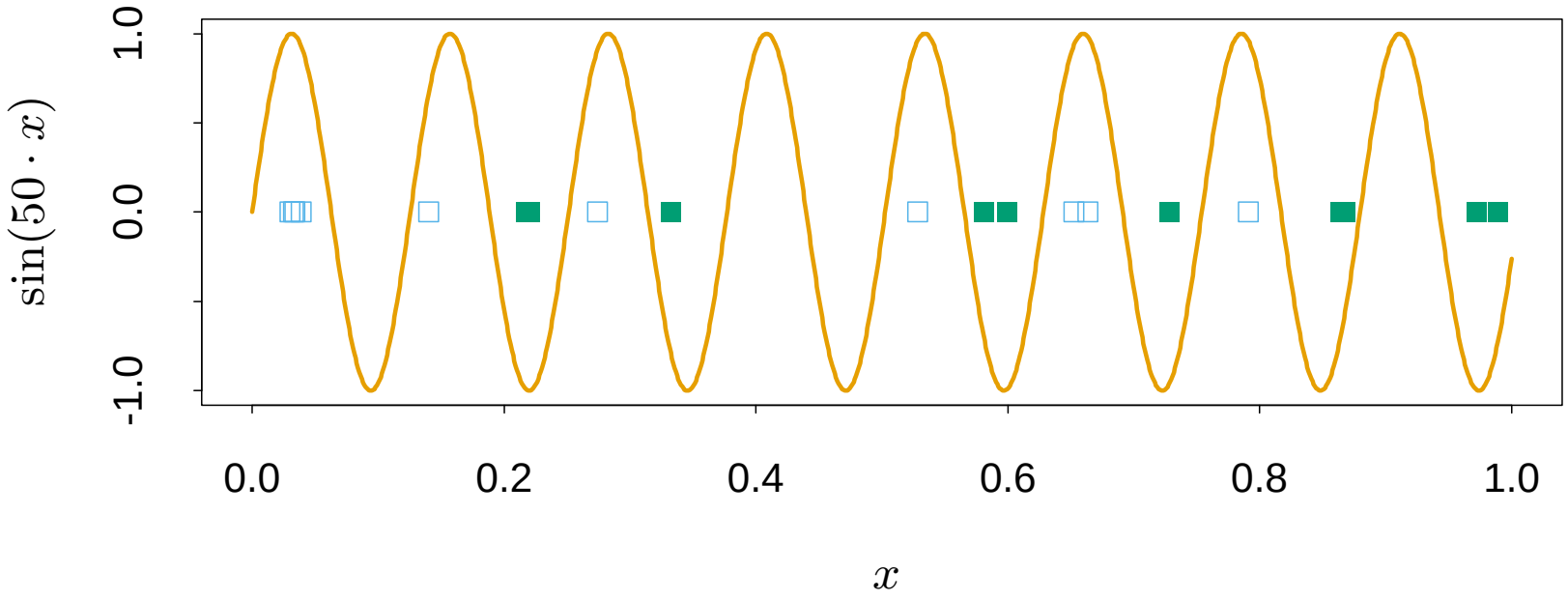
- Consider class of indicator functions $\mathcal{F} = \{f(x, \alpha)\}_\alpha$, indexed by a parameter α. eg $\mathcal{F}_1 = \{I(\alpha_0 + \alpha_1 x > 0)\}_\alpha$, or $\mathcal{F}_2 = \{I(\sin(\alpha x) > 0)\}_\alpha$.
- VC dimension of a class $\mathcal{F} = \{f(x, \alpha)\}_\alpha$ is defined to be the largest number of points (in some configuration) that can be *shattered* by members of $\mathcal{F}$.

A set of points is said to be shattered by a class of functions if, no matter how we we assign a binary label to each point, a member of the class can perfectly separate them.

- VC dim($\mathcal{F}_1$)=2, VC dim ($\mathcal{F}_2$)=∞



The first three panels show that the class of lines in the plane can shatter three points. The last panel shows that this class cannot shatter four points, as no line will put the hollow points on one side and the solid points on the other. Hence the VC dimension of the class of straight lines in the plane is three. Note that a class of nonlinear curves could shatter four points, and hence has VC dimension greater than three.



The solid curve is the function $\sin(50x)$ for $x \in [0, 1]$. The blue (hollow) and green (solid) points illustrate how the associated indicator function $I(\sin(\alpha x) > 0)$ can shatter (separate) an arbitrarily large number of points by choosing an appropriately high frequency α.

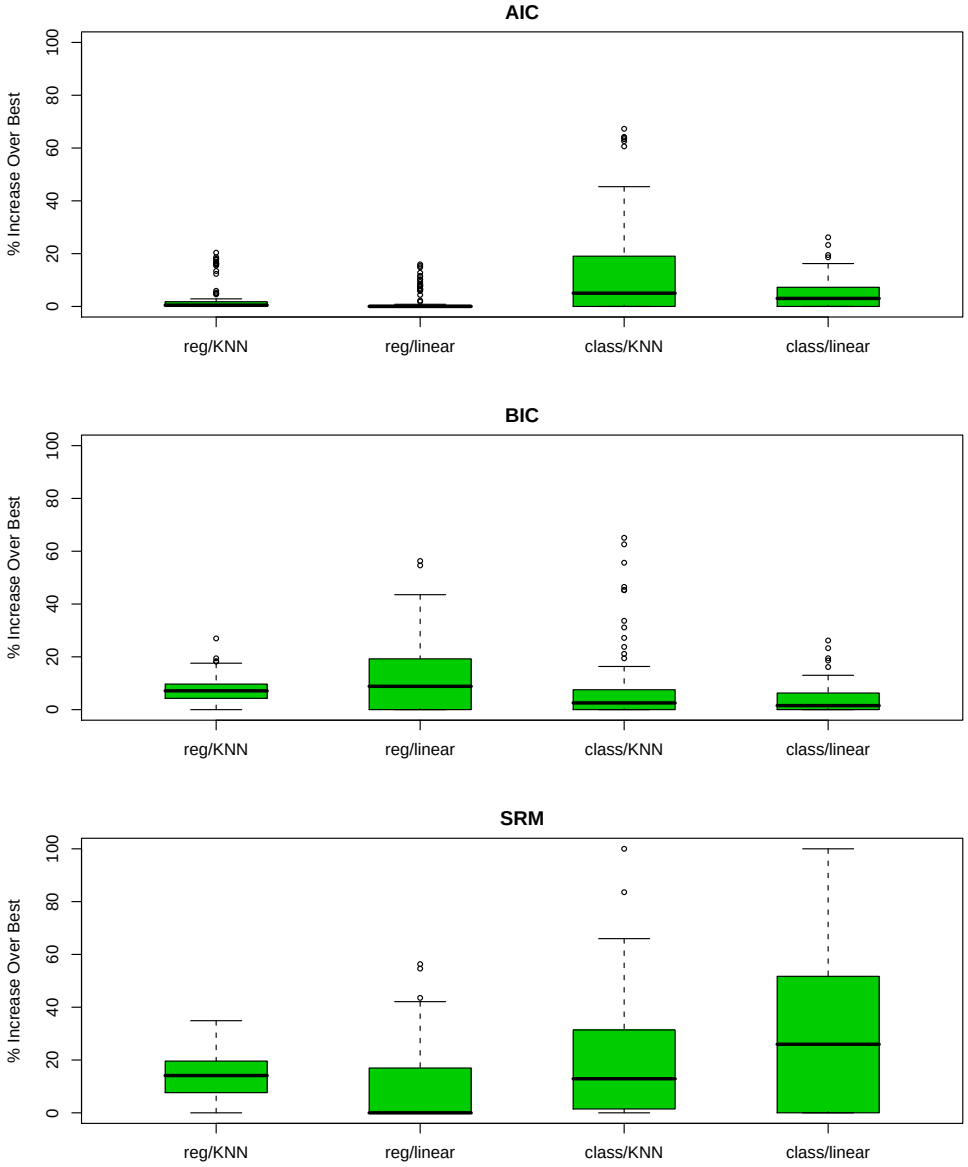
VC bounds

Example — binary classification, with class $\mathcal{F} = \{f(x, \alpha)\}_\alpha$ with VC dimension h. With Pr $> 1 - \eta$ over training samples

$$\begin{aligned} \text{Err}_{\mathcal{T}} \quad &\leq \quad \overline{\text{err}} + \frac{\epsilon}{2}\left(1 + \sqrt{1 + \frac{4 \cdot \overline{\text{err}}}{\epsilon}}\right) \\ &\text{where} \quad \epsilon = a_1 \frac{h[\log(a_2 N/h) + 1] - \log(\eta/4)}{N}, \end{aligned}$$

where $0 < a_1 \leq 4$ and $0 < a_2 \leq 2$.

These bounds are typically far too loose for reliable error estimation, but can nevertheless guide model selection (*Structural Risk Minimization* to Vapnik.)



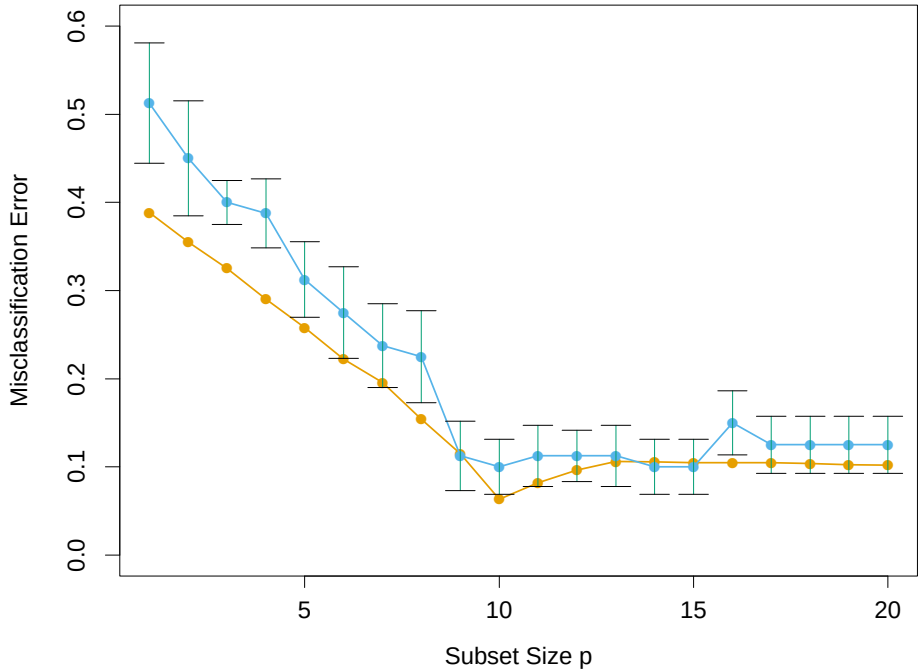
Boxplots show the distribution of the relative error

$$100\frac{\mathrm{Err}_{\mathcal{T}}(\hat{\alpha}) - \min_\alpha \mathrm{Err}_{\mathcal{T}}(\alpha)}{\max_\alpha \mathrm{Err}_{\mathcal{T}}(\alpha) - \min_\alpha \mathrm{Err}_{\mathcal{T}}(\alpha)}$$

over the four scenarios. This is the error in using the chosen model relative to the best model. There are 100 training sets each of size 80 represented in each boxplot, with the errors computed on test sets of size 10,000.

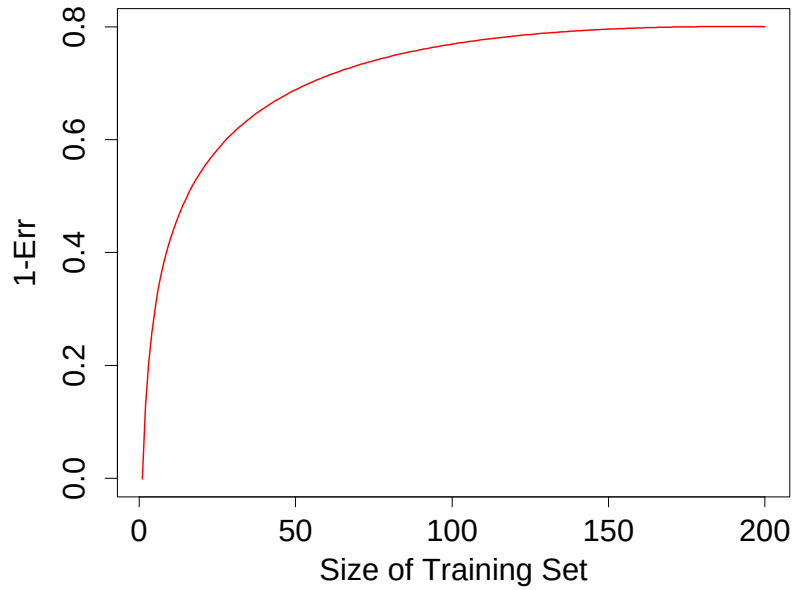
Cross-validation

Simple, and best overall—see chapter 3



Prediction error and ten-fold cross-validation curve estimated from a single training set, from the *Linear Model - Classification* scenario defined earlier.

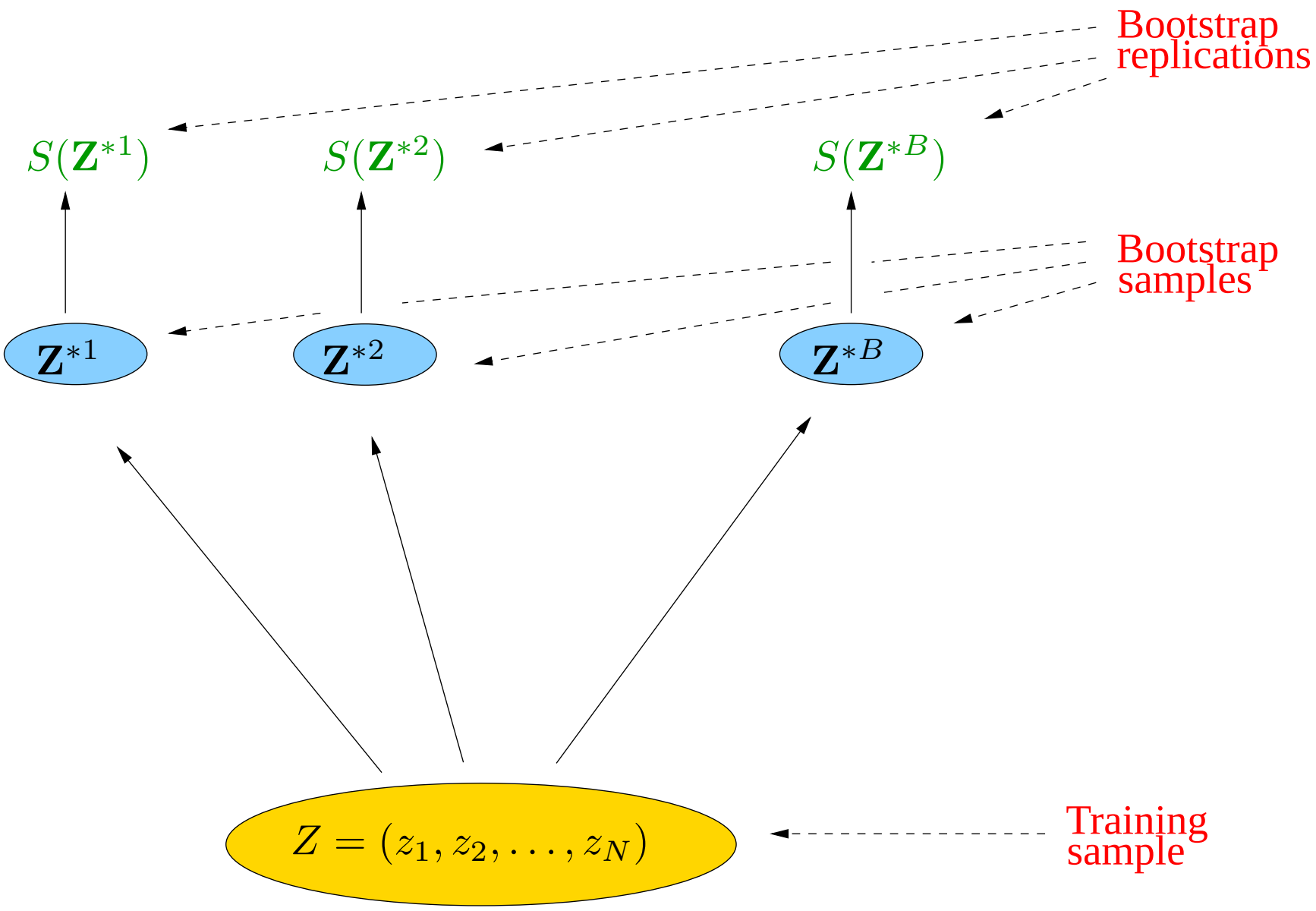
Cross-validation: Bias due to reduced training set size



Hypothetical *learning curve* for a classifier on a given task; a plot of $1 - \text{Err}$ versus the size of the training set N. With a dataset of 200 observations, fivefold cross-validation would use training sets of size 160, which would behave much like the full set. However, with a dataset of 50 observations fivefold cross-validation would use training sets of size 40, and this would result in a considerable overestimate of prediction error.

Bootstrap methods

- We wish to assess the statistical accuracy of a quantity $S(\mathbf{Z})$ computed from our dataset.
- B training sets $\mathbf{Z}^{*b},\ b = 1, \ldots, B$ each of size N are drawn with replacement from the original dataset.
- The quantity of interest $S(\mathbf{Z})$ is computed from each bootstrap training set, and the values $S(\mathbf{Z}^{*1}), \ldots, S(\mathbf{Z}^{*B})$ are used to assess the statistical accuracy of $S(\mathbf{Z})$.


Bootstrap replications
$S(\mathbf{Z}^{*1})$
$S(\mathbf{Z}^{*2})$
$S(\mathbf{Z}^{*B})$
Bootstrap samples
$\mathbf{Z}^{*1}$
$\mathbf{Z}^{*2}$
$\mathbf{Z}^{*B}$
$Z = (z_1, z_2, \ldots, z_N)$
Training sample

- bootstrap is useful for estimating the standard error of a statistic $s(\mathbf{Z})$: we use the standard error of the bootstrap values $s(\mathbf{Z}^{*1}), s(\mathbf{Z}^{*2}), \ldots s(\mathbf{Z}^{*B})$
- E.g. $s(\mathbf{Z})$ could be the prediction from a cubic spline curve at some fixed predictor value x.
- There is often more than one way to draw bootstrap samples—e.g. for a smoother, could draw samples from the data or draw samples from the residuals
- bootstrap is "non-parametric"— i.e doesn't assume a parametric distribution for the data. If we carry the bootstrap out parametrically (i.e. draw from a normal distribution), then we get the usual textbook (Fisher information-based) formulas for standard errors as $N \to \infty$.
- can get confidence intervals for an underlying population parameter from the percentiles for the bootstrap values $s(\mathbf{Z}^{*1}), \ldots s(\mathbf{Z}^{*B})$. Other more sophisticated confidence intervals via the bootstrap

Bootstrap estimation of prediction error

- $$\widehat{\mathrm{Err}}_{\mathrm{boot}} = \frac{1}{B}\frac{1}{N}\sum_{b=1}^{B}\sum_{i=1}^{N} L(y_i, \hat{f}^{*b}(x_i)).$$

- $$\begin{aligned} \Pr\{\text{observation } i \in \text{bootstrap sample } b\} &= 1 - \left(1 - \frac{1}{N}\right)^N \\ &\approx 1 - e^{-1} \\ &= 0.632. \end{aligned}$$

- Can be a poor estimate: Consider: 1-NN, 2 equal classes, class labels independent of features Then $\widehat{\mathrm{Err}}_{\mathrm{boot}} = 0.5(1 - 0.632) = 0.184$, while $\mathrm{Err} = 0.5$!

- Leave-one out bootstrap:

$$\widehat{\mathrm{Err}}^{(1)} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|C^{-i}|}\sum_{b\in C^{-i}} L(y_i, \hat{f}^{*b}(x_i)).$$

  where

$$C^{-i} = \{b : i \notin \text{bootstrap sample } b\}.$$

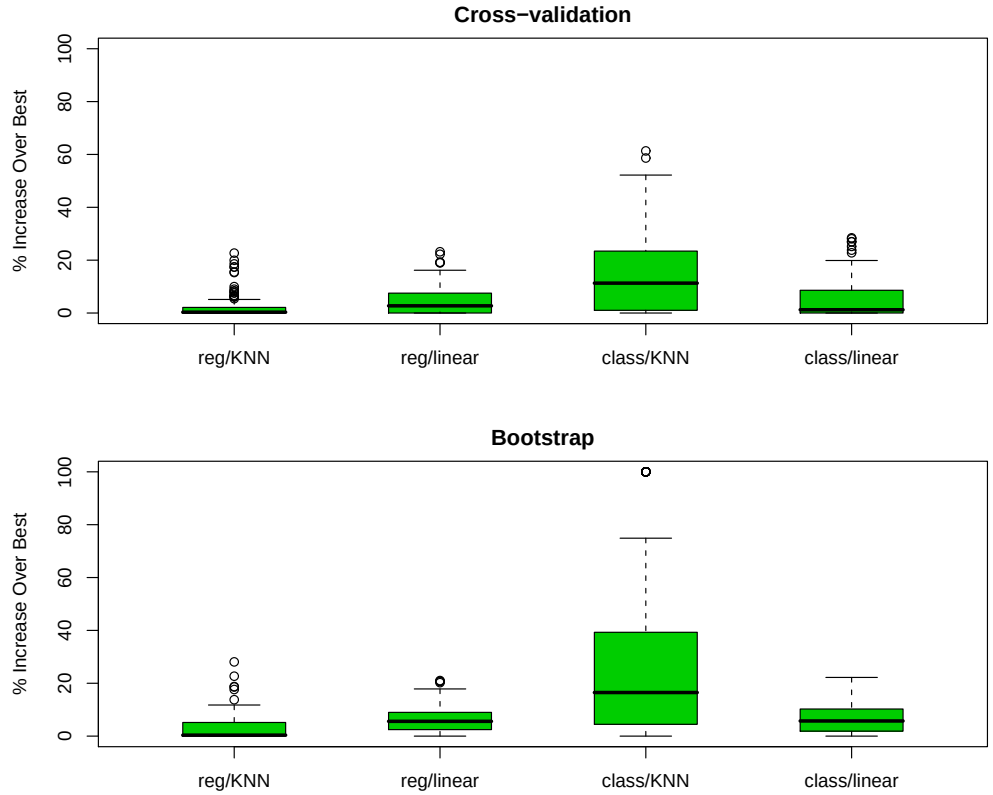
- .632 bootstrap estimator:

$$\widehat{\mathrm{Err}}^{(.632)} = .368 \cdot \overline{\mathrm{err}} + .632 \cdot \widehat{\mathrm{Err}}^{(1)}$$

  corrects for *learning curve* bias — bootstrap samples represent typically 0.632 of training samples.

- .632+ bootstrap estimator:

$$\vdots$$



Boxplots show the distribution of the relative error
$100 \cdot [\mathrm{Err}_{\mathcal{T}}(\hat{\alpha}) - \min_\alpha \mathrm{Err}_{\mathcal{T}}(\alpha)]/[\max_\alpha \mathrm{Err}_{\mathcal{T}}(\alpha) - \min_\alpha \mathrm{Err}_{\mathcal{T}}(\alpha)]$
over the four scenarios This is the error in using the chosen model relative to the best model. There are 20 training sets represented in each boxplot.