Outline

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

Model selection

- Loss functions

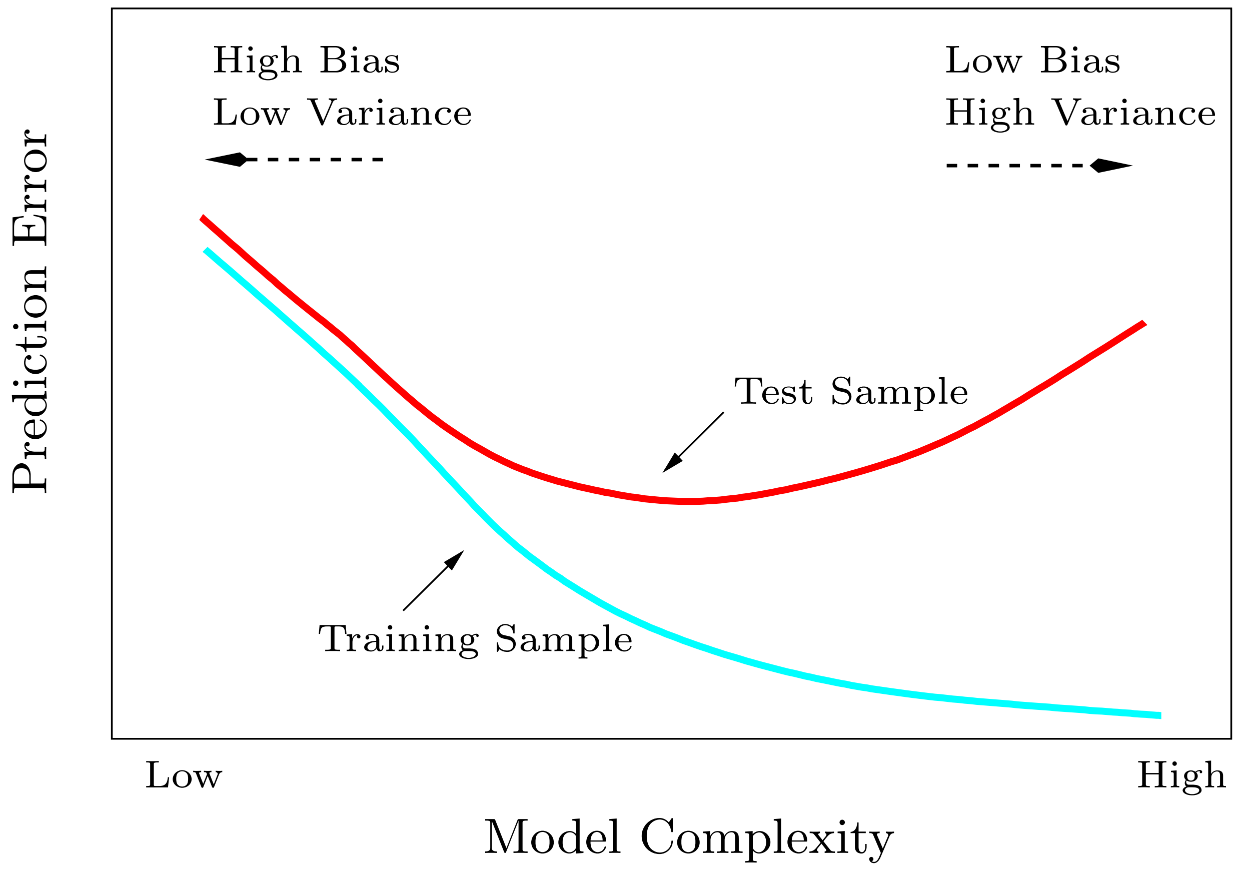
$$\begin{aligned}
L(Y, \hat{Y}(X)) &= (Y - \hat{Y}(X))^2 \quad \text{squared error,} \\
L(G, \hat{G}(X)) &= I(G \neq \hat{G}(X)) \quad \text{0–1 loss,} \\
L(G, \hat{p}(X)) &= -2 \sum_{k=1}^{K} I(G = k) \log \hat{p}_k(X) \\
&= -2 \log \hat{p}_G(X) \quad \text{log-likelihood.}
\end{aligned}$$

- Training error:

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

- Generalization error:

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



Behavior of test sample and training sample error as the model complexity is varied.

Bias-variance decomposition

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

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:

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

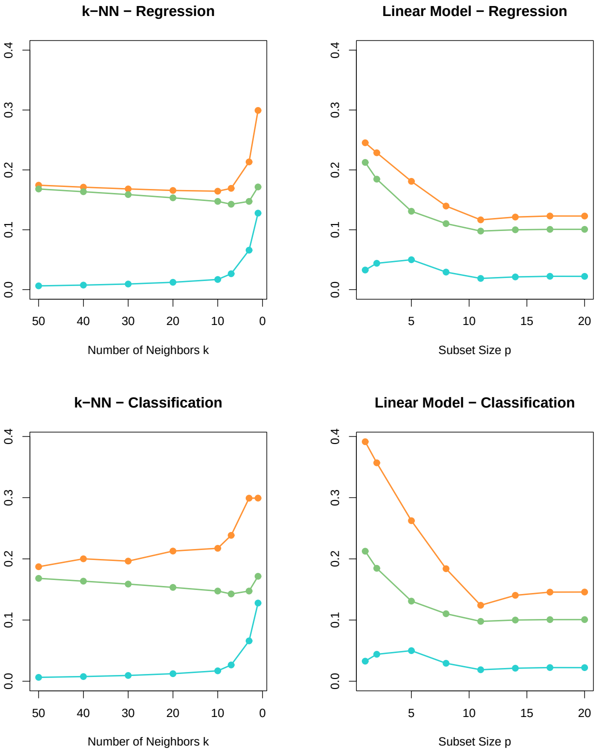
$$\frac{1}{N}\sum_{i=1}^{N}\mathrm{Err}(x_i) = \sigma_\varepsilon^2 + \frac{1}{N}\sum_{i=1}^{N}[f(x_i) - E\hat{f}(x_i)]^2 + \frac{p}{N}\sigma_\varepsilon^2, \qquad (1)$$

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

  $$Pr(\hat{y} \neq \hat{y}_B) = \Phi\Big[(\text{sign}(1/2 - f)\frac{E\hat{f} - 1/2}{\sqrt{\text{var}\hat{f}}}\Big]$$

  where $\hat{y}_B$ is the Bayes classifier (Exercise 7.63)
- Hence on the wrong side of the decision boundary, *increasing* the variance can help



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

Optimism of the Training Error Rate

- training error

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

- In-sample error rate

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

- Optimism

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

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

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

- For linear fitting:

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

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

$$\mathrm{Err_{in}} = \mathrm{E}_{\mathbf{y}}\overline{\mathrm{err}} + 2\cdot\frac{d}{N}\sigma_\varepsilon^2.$$

- Cp and AIC statistics:

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

$$-2\cdot\mathrm{E}[\log\mathrm{Pr}_{\hat{\theta}}(Y)] \approx -\frac{2}{N}\cdot\mathrm{E}[\mathrm{loglik}] + 2\cdot\frac{d}{N}. \qquad (2)$$

CP, AIC and linear operators

- $\hat{\mathbf{f}} = S\mathbf{y}$ (eg linear regression, ridge regression, cubic smoothing splines); $y_i = f_i + \epsilon_i$, $\mathrm{E}(\epsilon_i) = 0$.
- $\mathrm{Err}_{in} = \sigma_\epsilon^2 + (1/N)\sum[f(x_i) - \hat{f}_i]^2 + \sigma_\epsilon^2 \mathrm{tr}(S^T S)/N$
-

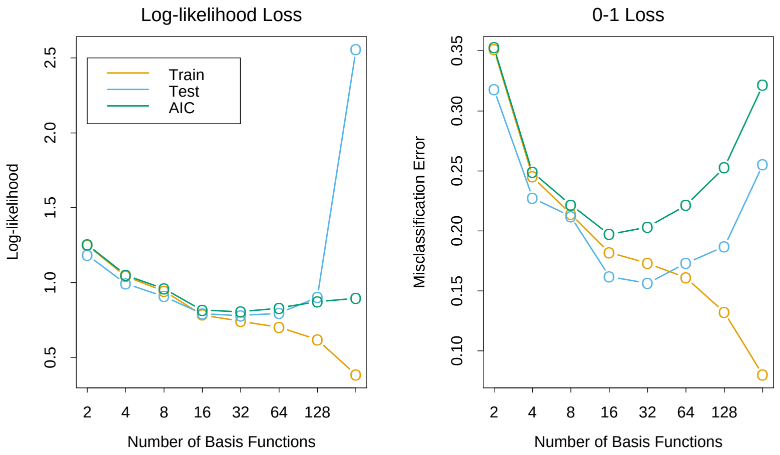
$$\begin{aligned}
\mathrm{E}(\overline{\mathrm{err}}) &= (1/N)\mathrm{E}||(I - S)\mathbf{y}||^2 \\
&= (1/N)\mathrm{E}(\mathbf{f}^T(I - S)^T(I - S)\mathbf{f} + (1/N)\mathrm{E}(\epsilon^T(I - S)^T(I - S)\epsilon) \\
&= \mathrm{Bias}^2 + (1/N)\sigma_\epsilon^2 \mathrm{tr}(I - 2S - S^2)
\end{aligned}$$

- Hence

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

and

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



AIC used for model selection for the 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. In the left panel we see the AIC statistic used to estimate* $\mathrm{Err_{in}}$ *using log-likelihood loss. Included is an estimate of* Err *based on a test sample. It does well except for the over-parametrized case* ($M = 256$ *parameters for* $N = 1000$ *observations). In the right panel the 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 Gaussian 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

  $$\text{BIC} = \frac{N}{\sigma_\varepsilon^2}\Big[\overline{\text{err}} + (\log N) \cdot \frac{d}{N}\sigma_\varepsilon^2\Big].$$
- candidate models $\mathcal{M}_m, m = 1, \ldots, M$ prior distribution $\Pr(\theta_m|\mathcal{M}_m)$ 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

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

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

VC dimension

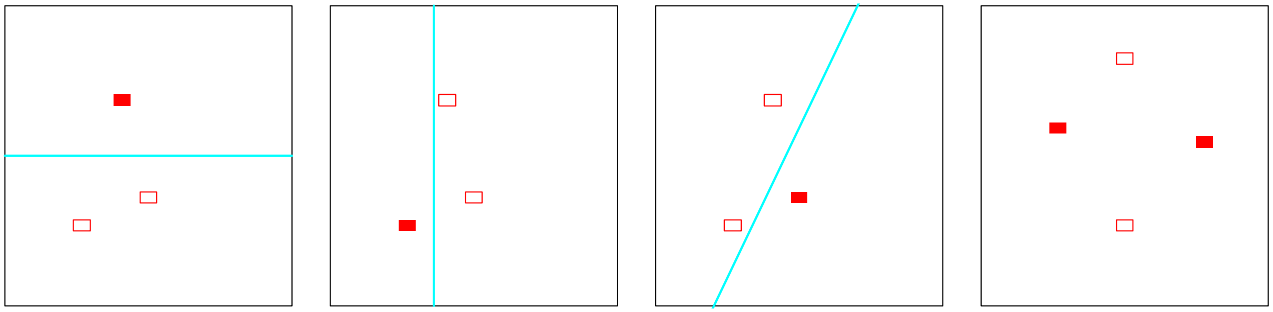
- class of indicator functions $f(x, \alpha)$, indexed by a parameter α. eg $f_1 = I(\alpha_0 + \alpha_1 x > 0)$, or $f_2 = I(\sin(\alpha x) > 0)$.
- VC dimension of a class $\{f(x, \alpha)\}$ is defined to be the largest number of points (in some configuration) that can be *shattered* by members of $\{f(x, \alpha)\}$.

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

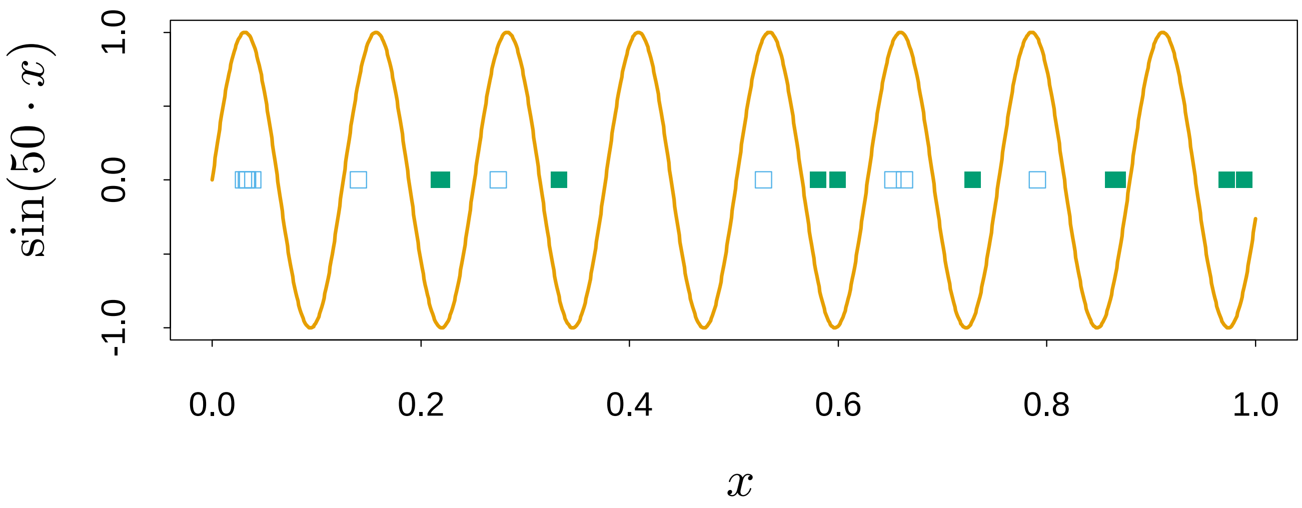
- VC dim(f_1)=2, VC dim (f_2)=∞
- VC bounds

  sets:

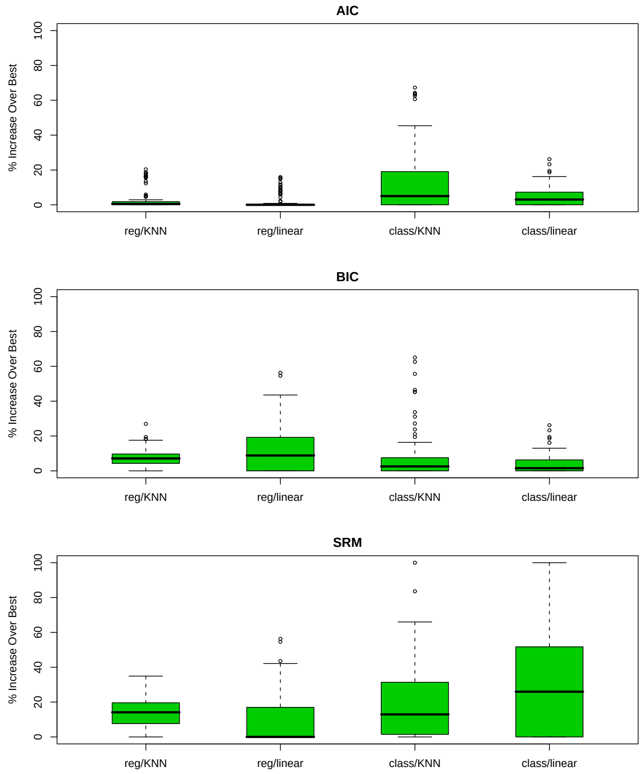
$$\begin{aligned}\mathrm{Err} &\leq \overline{\mathrm{err}} + \frac{\epsilon}{2}\Big(1 + \sqrt{1 + \frac{4 \cdot \overline{\mathrm{err}}}{\epsilon}}\Big) \\ &\text{where} \quad \epsilon = a_1 \frac{h[\log{(a_2 N/h)} + 1] - \log{(\eta/4)}}{N}.\end{aligned}$$



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.



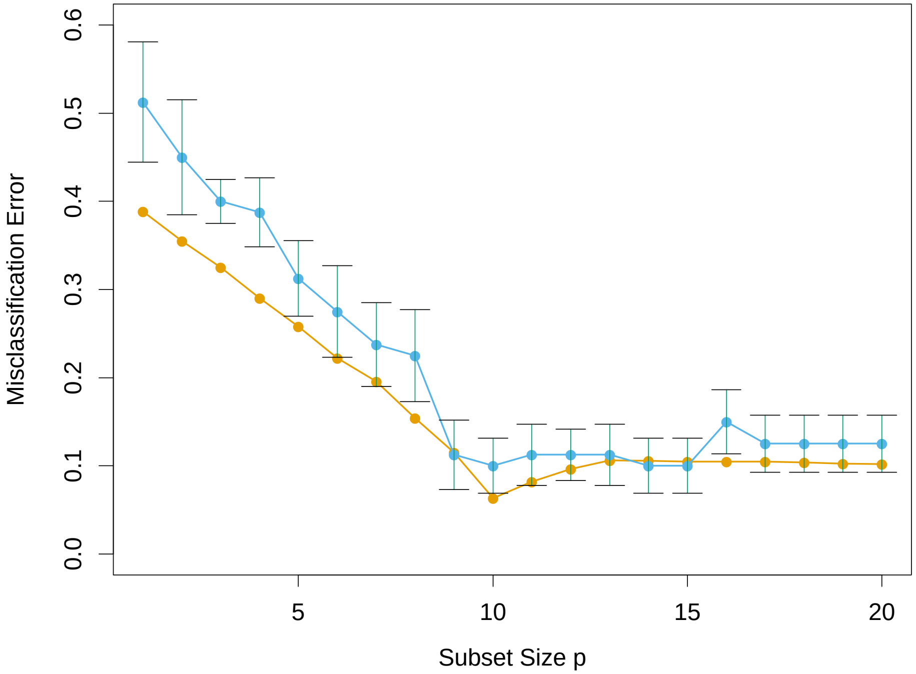
The solid curve is the function $\sin(50x)$ *for* $x \in [0, 1]$*. The blue (solid) and green (hollow) points illustrate how the associated indicator function* $I(\sin(\alpha x) > 0)$ *can shatter (separate) an arbitrarily large number of points by choosing an appropriately high frequency* α*.*



Boxplots show the distribution of the relative error $100 \times [\mathrm{Err}(\hat{\alpha}) - \min_\alpha \mathrm{Err}(\alpha)]/[\max_\alpha \mathrm{Err}(\alpha) - \min_\alpha \mathrm{Err}(\alpha)]$ *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* 50 *represented in each boxplot, with the errors computed on test sets of size 500.*

Cross-validation

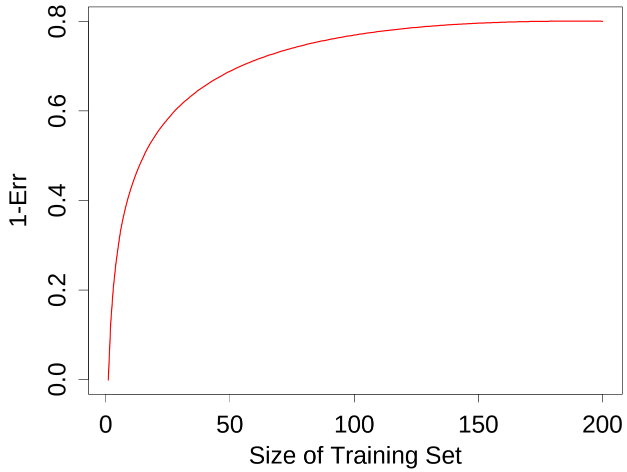
Simple, and best overall- see chapter 3



Prediction error (orange) and tenfold cross-validation curve (green) estimated from a single training set, from the class/linear 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 $1 - \mathrm{Err}$ *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 $\mathbf{Z}$.
- 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 is useful for estimating the standard error of a statistic $s(\mathbf{Z})$: we use the standard error of the bootstrap values $s(\mathbf{Z}^{*1}), s(\mathbf{Z}^{*2}), \ldots s(\mathbf{Z}^{*B})$
- Eg $s(\mathbf{Z})$ 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- eg 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 dist'n 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 paramater from the percentiles for the bootstrap values $s(\mathbf{Z}^{*1}), s(\mathbf{Z}^{*2}), \ldots s(\mathbf{Z}^{*B})$. There are other, more sophisticated ways to form 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 $\widehat{\mathrm{Err}}_{\mathrm{boot}} = 0.5(1 - .632) = .184$; true value is 0.5!

- Leave-one out bootstrap:

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

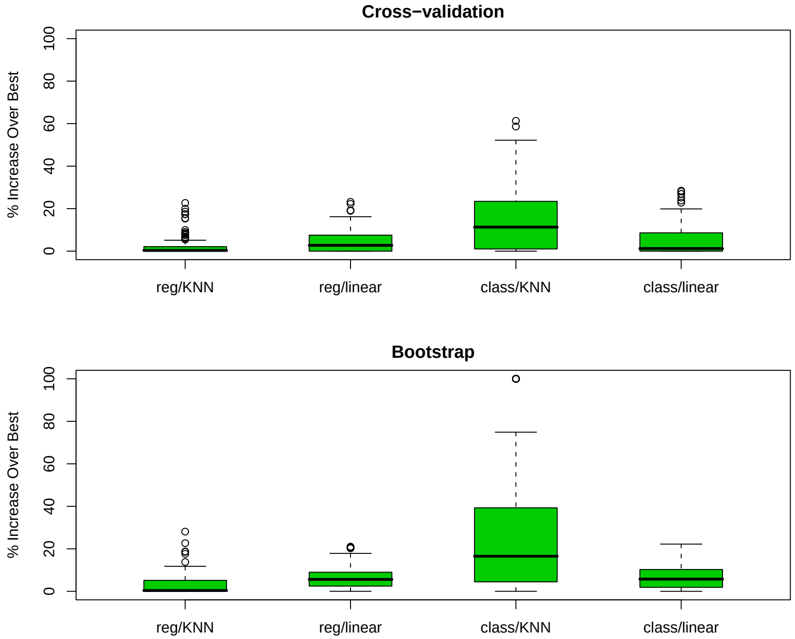
- .632 bootstrap estimator:

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

- .632+ bootstrap estimator:

$$\widehat{\text{Err}}^{(.632+)} = (1-w)\cdot\overline{\text{err}} + w\cdot\widehat{\text{Err}}^{(1)}.$$

  where $w = .632/(1-.368R)$, $R=$ “overfitting rate”



Boxplots show the distribution of the relative error $100 \cdot [\mathrm{Err}_{\hat{\alpha}} - \min_\alpha \mathrm{Err}(\alpha)]/[\max_\alpha \mathrm{Err}(\alpha) - \min_\alpha \mathrm{Err}(\alpha)]$ *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.*