New York University: Machine Learning in Economics

Lecture 6: Model Selection and Averaging

Dr. George Lentzas

1. Introduction

2. Three types of Error

3. The Bias Variance Decomposition

4. Resampling Methods

5. All You Wanted to Know About CV

How to do ML/AI Responsibly

- Most practical applications of ML/AI are judged on their *predictive ability*.
- Predictive ability depends on *generalization performance*, i.e. how well the model uses the sample data to learn attributes that describe the entire population.
- Being able to evaluate a model's predictive ability is crucial for two purposes: (i) choice of model and (ii) assessment of the quality of chosen model.
- So in this class we will discuss how to do ML/AI responsibly. How to choose appropriate models, judge if the chosen model performs well and ultimately be confident that your analysis will be successful in practice.

Three Types of Error

Notation

- We have a target variable Y, a vector of inputs (a.k.a. features or explanatory variables) X, and a prediction model $\hat{f}(X)$ that is estimated on some training dataset T.
- Training is done by minimizing a loss function of our choice $L\left[Y, \hat{f}(X)\right]$, typically the squared loss.

Training Error

- Training error is the average loss over the training data.

$$\mathrm{Err}_{\mathrm{train}} = \frac{1}{N}\sum_{i=1}^{N} L\left[y_i, \hat{f}(x_i)\right]$$

Three Types of Error

Test Error

- Test error is the prediction error over an independent test sample, i.e. both X and Y drawn from their joint distribution but *given* (conditional on) the training dataset T.
- This is a sample dependent metric.

$$\mathsf{Err}_{\mathrm{T}} = \mathrm{E}\left[L\left(Y, \hat{f}\,(X)\right)|\mathrm{T}\right]$$

Expected Prediction Error

- Expected Prediction Error is the unconditional test error of the model, when estimated on any sample, with all data (training and test datasets) drawn from the joint distribution of Y and X.

$$\mathsf{Err} = \mathrm{E}\left[L\left(Y, \hat{f}\,(X)\right)\right] = \mathrm{E}\left[\mathsf{Err}_{\mathrm{T}}\right]$$

because of iterated expectations

Three Types of Error

Which Error?

- Which of these errors should we care about (that is try to minimize) if our model's performance is judged on predictive ability?

Test Error

- Ideally, we would focus on minimizing the Test Error (Err_{T}). Why?
- We would use Test Error to achieve our two goals:
  1. **Model Selection**: choose the best model among may.
  2. **Model Assessment**: having chosen a model, assess its predictive ability and hence its performance.
- In practice it is easier to analyze and estimate the Expected Prediction Error.

Three Types of Error

Training Error

- First a warning: it is not a good idea to use training error to evaluate model performance.
- This is because training error will *always* decrease with model complexity and can even be zero if the model is complex enough.
- However, a model with very low or zero training error is overfit to the training data and will generalize poorly.

Three Types of Error

Elements of Statistical Learning (2nd Ed.) ©Hastie, Tibshirani & Friedman 2009 Chap 7

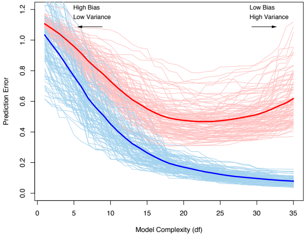


FIGURE 7.1. *Behavior of test sample and training sample error as the model complexity is varied. The light blue curves show the training error* $\overline{\text{err}}$*, while the light red curves show the conditional test error* $\text{Err}_{\mathcal{T}}$ *for* 100 *training sets of size* 50 *each, as the model complexity is increased. The solid curves show the expected test error* Err *and the expected training error* $\text{E}[\overline{\text{err}}]$.

The Bias-Variance Decomposition

- Crucial to understanding model performance is the "Bias-Variance Decomposition".
- Let us assume a regression setting where $Y = f(X) + \epsilon$ and that $\mathrm{Var}(\epsilon) = \sigma_\epsilon^2$.Then the Expected Prediction Error of a fit $\hat{f}(X)$ at an input point $X = x_0$ using square loss is

$$\begin{aligned}\text{Err}(x_0) &= \mathrm{E}\left[\left(Y - \hat{f}(x_0)\right)^2\right] \\ &= \sigma_\epsilon^2 + \text{Bias}^2\left[\hat{f}(x_0)\right] + \mathrm{Var}\left[\hat{f}(x_0)\right]\end{aligned}$$

- The first term is the the *irreducible error*, the second is the *squared bias* of the fit and the last is the *variance* of the fit. To understand this decomposition, ask yourself, what are we taking expectation with respect to in the expression above?

The Bias-Variance Decomposition

- **Variance** captures how $\hat{f}$ would change if estimated over different training data sets. The higher the flexibility of the ML method the higher its variance.
- **Bias** captures the error that arises by approximating the real data generating process by a simpler model. The more flexible the ML method, the lower the (squared) bias.
- The trade-off between these two creates the familiar U-shape curve: initially, the gain from increased flexibility leading to lower bias dominates and test MSE drops but eventually the variance introduced becomes more important and test MSE starts to increase again.

- When choosing among models we can index possible choices according to their complexity / flexibility (for example, number of regressors, number of layers, number of neurons, etc). We can see this so-called **Variance-Bias Trade-off** clearly below and why using the Training Error is not a good idea.

Bias, Variance and Model Complexity

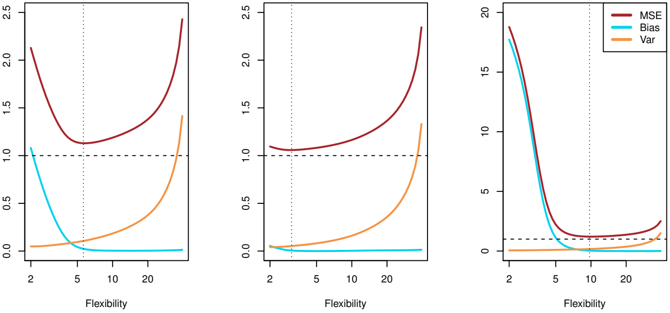


Figure: The blue curve shows the squared bias and the orange curve the variance. Test MSE is given by the red curve and the variance of the error by the dashed horizontal curve. The three graphs correspond to three different data sets.

Bias, Variance and Model Complexity

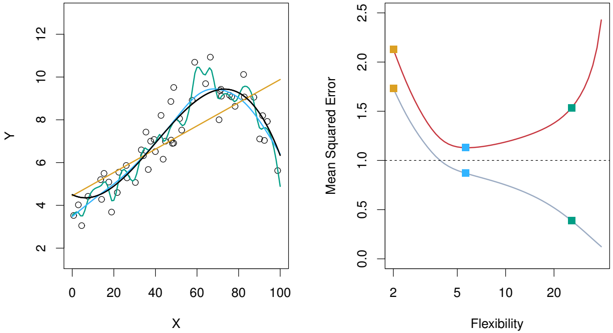


Figure: Simulated data in black with three estimates of $f(x)$, orange, blue and green in increasing flexibility. Test MSE (red curve) and training MSE (grey curve). The horizontal grey line is $\mathrm{Var}(\epsilon)$ which corresponds to the lowest possible test MSE.

Resampling Methods

Introduction

- Resampling involves repeatedly drawing samples from a training set and refitting a statistical model to each sample in order to obtain additional information. Here we will review one such method called *cross-validation*.

Cross-Validation

- We have seen that out general motivation is to choose models that minimize the test error rate, that is the average error that results from using the statistical technique to predict the output variable associated with new observations.
- The cross-validation approach is to estimate the test error rate by holding out a subset of the training data from fitting the model and then treat the left out data as a test sample and the fitted model on estimate the test error rate on the left out / test data.

Resampling Methods

The Validation Set Approach

- We start with the simplest such method, the so called Validation Set Approach. The idea is to randomly divide the available set of observations into two subsets: a training set and a test ("validation") set. The model is fitted on the former and the resulting fitted model is used to make predictions for the responses in the validation set.
- The resulting validation set error rate can provide an estimate of the test error rate.
- The main advantage of this is that it is conceptually simple! However it suffers from two significant drawbacks:
  1. The estimate of the test error rate can be highly variable depending on which or how many observations are assigned to each of the test/train sub sets.
  2. Since only a subset of observations are used to fit the model the validation estimate will over-estimate the test error rate for the model if fitted on the entire data set.

Resampling Methods

The Validation Set Approach

- Below we see the Validation Set Approach used on the 'Auto' data example from ISLR. We can see the variability issue on the right hand panel.

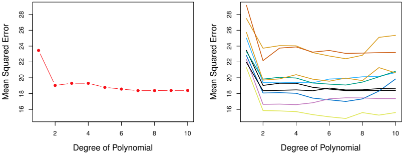


Figure: On the left we see vallidation test error estimates depending on the degree of polynomial used in the mode. On the right the approach was repeated ten times, each using a different data split.

Resampling Methods

The Leave-One-Out Validation (LOOTA/LOOCV) Approach

- One of the obvious way to deal with the above issues is to decrease the size of the test subset. LOOTA takes this idea to the limit and uses only one test observation.
- The procedure is then repeated n times and the LOOTA estimate of the test error rate is the average of the n test error rates estimated,

$$LOOTA(n) = \frac{1}{n}\sum_{i=1}^{n} MSE_i$$

- This approach deals with both problems of the Validation Set Approach. It will not majorly over-estimate the test error rate (after all we are using $n-1$ versus n observations). In addition if you perform LOOTA many times you get the same result, as there is no randomness associated with splitting the data.
- The catch is that LOOTA is potentially very computationally expensive as the model of choice needs to be fitted n times.

Resampling Methods

K-Fold Cross Validation Approach (kCV)

- An alternative approach is kCV which involves randomly dividing the set of observations into k groups of approximately equal size. The first set is used as a test/validation sample, having fit the model on the remaining $k-1$ sets.
- The MSE is then calculated using the test set and the process is repeated k times, each time using a different subset as the test/validation sample on which to estimate the MSE.
- The k-fold CV estimate of the test error rate is then given by the average of the k different values:

$$CV\left(k\right)=\frac{1}{k}\sum_{i=1}^{k}MSE_i$$

- Typically k is set to 5 or 10.

Resampling Methods

K-Fold Cross Validation Approach (kCV)

- The obvious advantage of kCV is computational; we only need to fit the model k times (as opposed to n times with LOOTA). The below figure shows the kCV estimates of test error rates from applying a smoothing spline model to the simulated data considered previously (see Classification slides).

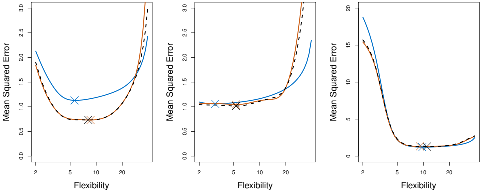


Figure: True (blue) and estimated (LOOTA in black) and 10CV in (orange) test error rates.

Resampling Methods

Comparison

- If computational issues are not a concern, should we choose LOOTA? The answer has to do with the familiar bias-variance trade-off.
- LOOTA will give asymptotically unbiased estimates of the test errors each training set contains $n-1$ observation, very close to the true n observations in the sample. On the contrary kCV will use $\frac{(k-1)n}{k}$ *observations*. So in terms of bias reduction it should be obvious that LOOTA is superior to kCV.
- However we know that in any estimation bias reduction is only half the story. What about the variance? LOOTA tends to have higher variance than kCV! Intuitively this is because in LOOTA we are averaging n correlated estimates while in kCV we are averaging k less correlated estimates.
- As the mean of highly correlated random variables has higher variance than that of non highly correlated ones we expect kCV to be superior in terms of variance reduction. Which brings us back to the choice of $k = 5$ or $k = 10$ as these have been shown to give test error estimates that successfully balance bias and variance.

kCV > LOOTA

Resampling Methods

k CV for Classification

- So far we have set the discussion in terms of a regression problem. But what about Classification? The same concepts extend naturally to the classification setting.
- Now instead of using MSE to capture test error we instead use the number of misclassified observations in the test data,

$$Err_i = I\left(y_i \neq \hat{y}_i\right)$$

- Using this definition we calculate the k-fold CV error rate by

$$CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} Err_i$$

- In the next slide you can see an example where we use 10-fold CV for polynomial logistic regression (choice of polynomial order) and kNN (choice of k). We observe that the training error keeps declining as the model becomes more flexible and hence cannot be relied upon to select the optimal k in kNN. Although 10-fold CV slightly underestimates the true test MSE it produces an optimal k that is close to the truly best value.

Resampling Methods

k CV for Classification

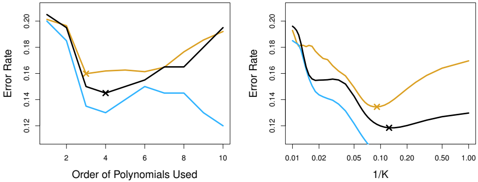


Figure: Since generally we do not know the true Bayes decision boundary nor the test error rate we use 10-fold CV. True test error rate (brown), training error (blue) and 10CV error (balck) are shown for logistic regression (left) and kNN (right).

Resampling Methods

R Considerations

- R makes Cross Validation easy to implement with a number of ready-made functions and indexing options.
- set.seed(): sets a seed for the random number generator so that you can reproduce results.
- sample(): creates a sample of the specified size either with or without replacement.
- cv.glm(): [part of "boot" library] calculates the estimated K-fold cross-validation prediction error for generalized linear models.
- knn(): [part of "class" library] performs kNN

Resampling Methods

Best Case Scenario

- The ideal scenario is that have enough data to *randomly* divide the dataset into three parts: a **training set**, a **validation set**, and a **test set**.
- The training set is used to fit the possible models, the validation set is used to estimate prediction error to select a model and the test set is used (once) to evaluate the performance of the final chosen model.
- One word of caution: there is still some risk of under-estimating the true error rate in the presence of *ephemeral predictors*, that is features whose predictive power fades over time (giving rise to fading correlations).
- Ideally you would want a validation set that is "separated" from the training data (in a time sense or otherwise). This is why in k-fold CV the folds are continuous (vs random) although this is a second best approach.

Resampling Methods

The Gold Standard: Cross Validation

- In many situations we do not have enough data to adequately do training, validation and testing on separate samples. By far the most commonly used technique in such situations is Cross-Validation (CV).
- The Cross-Validation approach is to estimate the test error rate by holding out a subset of the training data from fitting the model and then treat the left out data as a test sample.

k-Fold Cross Validation Approach (k-CV)

- k-fold CV involves randomly dividing the set of observations into k groups of approximately equal size. The first set is used as a test/validation sample, having fit the model on the remaining $k-1$ sets. The prediction error is then calculated using the test set and the process is repeated k times.
- The k-fold CV estimate of the error rate is then given by:

$$CV\left(\hat{f}\right) = \frac{1}{N}\sum_{i=1}^{N} L\left[y_i, \hat{f}^{-\kappa(i)}\left(x_i\right)\right]$$

k-Fold Cross Validation Approach (k-CV)

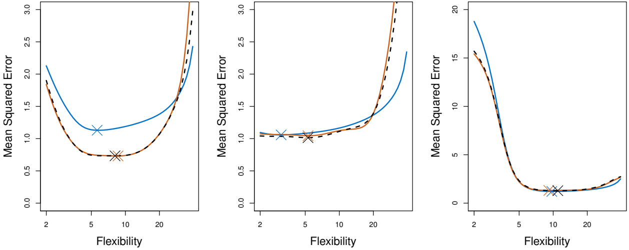


Figure: True (blue) and estimated (1-CV in black and 10-CV in orange) error rates.

All You Wanted to Know About CV

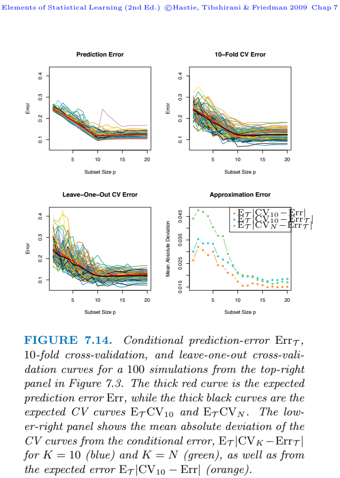
What does CV Estimate?

- What type of error does CV estimate? Is it **Test Error** (Err_T) or **Expected Prediction Error** (Err)?
- You might expect that the higher the number of folds, the closer k-fold CV gets to using the full sample to fit new test points and thus the better it estimates Test Error (Err_T).
- You might similarly expect that with a low number of folds it averages over somewhat different training sets and thus estimates Expected Prediction Error ($\text{Err} = \text{E}\left[\text{Err}_\text{T}\right]$).

Not What You Might Expect

- Actually, (i) 10-fold CV is thought to do a better job than N-fold CV in estimating in estimating Err_T and (ii) 10-fold CV does a better job in estimating Err than Err_T.
- In other words, **CV estimates the expected prediction error**.

All You Wanted to Know About CV



Elements of Statistical Learning (2nd Ed.) ©Hastie, Tibshirani & Friedman 2009 Chap 7

FIGURE 7.14. *Conditional prediction-error* $\mathrm{Err}_{\mathcal{T}}$, 10*-fold cross-validation, and leave-one-out cross-validation curves for a* 100 *simulations from the top-right panel in Figure 7.3. The thick red curve is the expected prediction error* Err, *while the thick black curves are the expected CV curves* $\mathrm{E}_{\mathcal{T}}\mathrm{CV}_{10}$ *and* $\mathrm{E}_{\mathcal{T}}\mathrm{CV}_N$. *The lower-right panel shows the mean absolute deviation of the CV curves from the conditional error,* $\mathrm{E}_{\mathcal{T}}|\mathrm{CV}_K - \mathrm{Err}_{\mathcal{T}}|$ *for* $K = 10$ *(blue) and* $K = N$ *(green), as well as from the expected error* $\mathrm{E}_{\mathcal{T}}|\mathrm{CV}_{10} - \mathrm{Err}|$ *(orange).*

All You Wanted to Know About CV

k-fold CV Does a Poor Job for $\mathrm{Err}_{\mathrm{T}}$

- The next figure shows scatter-plots of 10-fold and N-fold CV error estimates (for various best subset regressions) compare to the true $\mathrm{Err}_{\mathrm{T}}$.
- The bottom right panel shows that for the most part the correlations are negative!

How do we choose k?

- Since k-fold CV error is an estimate of the expected prediction error (Err) then the Bias-Variance Trade-off applies here too. As k increases the CV estimator becomes more unbiased but has higher variance since the training sets become more similar to one another.
- What matters is how the performance of the model varies with the size of the training set.
- To visualize this consider a learning curve for a classifier as in the figure below. 5-fold CV on 200 observations uses 160 observations which seems to introduce little bias.
- If the learning curve has considerable slope then small-k-fold CV will overestimate the true Err (introduced bias) although it is unclear that this is an problem. The choice of 5 or 10 folds is considered a good compromise for most situations.

All You Wanted to Know About CV

Elements of Statistical Learning (2nd Ed.) ©Hastie, Tibshirani & Friedman 2009 Chap 7

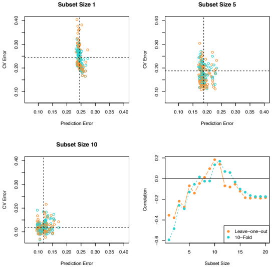


FIGURE 7.15. *Plots of the CV estimates of error versus the true conditional error for each of the* 100 *training sets, for the simulation setup in the top right panel Figure 7.3. Both* 10*-fold and leave-one-out CV are depicted in different colors. The first three panels correspond to different subset sizes* p*, and vertical and horizontal lines are drawn at* $\text{Err}(p)$*. Although there appears to be little correlation in these plots, we see in the lower right panel that for the most part the correlation is* negative.

All You Wanted to Know About CV

Elements of Statistical Learning (2nd Ed.) ©Hastie, Tibshirani & Friedman 2009 Chap 7

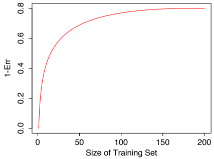


FIGURE 7.8. *Hypothetical learning curve for a classifier on a given task: a plot of* $1-\text{Err}$ *versus the size of the training set* N*. With a dataset of* 200 *observations,* 5*-fold cross-validation would use training sets of size* 160*, which would behave much like the full set. However, with 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.*