Cross Validation

UBCO MDS — DATA 571

Motivation

- We've now (post DATA 570) seen some examples for statistical learning — some regression and a tiny bit of classification.
- We've yet to truly get into how to choose among various modelling options.
- For example: how would you *objectively* choose between kNN regression versus MLR?
- Sidenote: would you like to know how *this model* will behave in the long-run? Or would you like to know how *the modelling process you undertook* will? One of these is more feasible to estimate...

Recall

- Our main goal for regression methods is to reduce the Mean Squared Error for the model. We could describe this as

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

- Somewhat similarly, for classification methods we consider the misclassification rate

$$\frac{1}{N}\sum_{i=1}^{N} I(y_i \neq \hat{y}_i)$$

Recall

- For both MSE and the error rate, I use N as notation to suggest that we want to minimize these for the population rather than just the sample we have in front of us.

- Ideally, we could fit our model on a sample of data, and then compare the predictions from the estimated model on a very large testing data set to estimate the MSE.
  "But...if we have more data available...then...?"

- And if we only have a sample, (naively) it would appear impossible to attempt to estimate the MSE or error rate for the population.

Splitting Up the Sample

- The first, and perhaps most obvious, option is to randomly split our data set into non-overlapping training and testing sets:

Full Sample Index
$1, 2, 3, 4, 5, \ldots, 15, 16, 17, \ldots, 198, 199, 200$

Randomly Sample

$$\swarrow \qquad \searrow$$

Training Index	Testing Index
$2, 3, \ldots, 7, 9, \ldots, 199$	$1, 4, 5 \ldots, 8, \ldots, 198, 200$

- In this case, we fit the model on the training set, and estimate the MSE using the testing set.

- Any concerns?

Leave One Out Cross-Validation

- Another option: LOOCV
- A systematic way of creating multiple validation sets.
- We create n training sets of size $(n-1)$ wherein each set has one observation removed. This leaves us n validations of size 1 as well.

Leave One Out Cross-Validation

	Full Sample Index	
	$1, 2, 3, 4, 5, \ldots, 15, 16, 17, \ldots, 198, 199, 200$	
	Systematically divides into...	
CV Set	Training Index	Testing Index
1	$2, 3, \ldots, 200$	1
2	$1, 3, \ldots, 200$	2
3	$1, 2, \ldots, 200$	3
$\vdots$	$\vdots$	$\vdots$
200	$1, 2, 3, \ldots, 199$	200

Leave One Out Cross-Validation

- We then fit our model to each of the $i = 1, \ldots, n$ CV training sets, and receive a prediction for the i^{th} testing (or left out) observation.

- If we define $MSE_i = (y_i - \hat{y}_i)^2$ where $\hat{y}_i$ is found from the i^{th} model

- We can then estimate the MSE of the model using

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

LOOCV

- Pros/Cons of LOOCV versus a regular test set?

k-Fold CV

- We can consider other alternatives of cross validation instead of leave-one-out.
- We can randomly subdivide the sample into k approximately equally-sized and non-overlapping sets.
- Each set can be considered a validation set, with the remainder of the data used to train the model.
- Then we can calculate the MSE for each validation set j

$$MSE_j = \frac{1}{\sum_{i=1}^{n} I(i \in j)} \sum_{i \in j} (y_i - \hat{y}_i)^2$$

- And estimate the test MSE with

$$CV_{(k)} = \frac{1}{k} \sum_{j=1}^{k} MSE_j$$

k-Fold Cross-Validation

	Full Sample Index	
	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20	
	Randomly divides into...	
CV Set	Training Index	Testing Index
1	1, 2, 4, 5, 6, 8, 9, 11, 13, 14, 15, 16, 17, 18, 19, 20	3, 7, 10, 12
2	1, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 15, 16, 17, 18, 20	2, 4, 14, 19
3	1, 2, 3, 4, 6, 7, 8, 9, 10, 11, 12, 14, 15, 17, 19, 20	5, 13, 16, 18
4	1, 2, 3, 4, 5, 7, 10, 11, 12, 13, 14, 16, 17, 18, 19, 20	6, 8, 9, 15
5	2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15, 16, 18, 19	1, 11, 17, 20

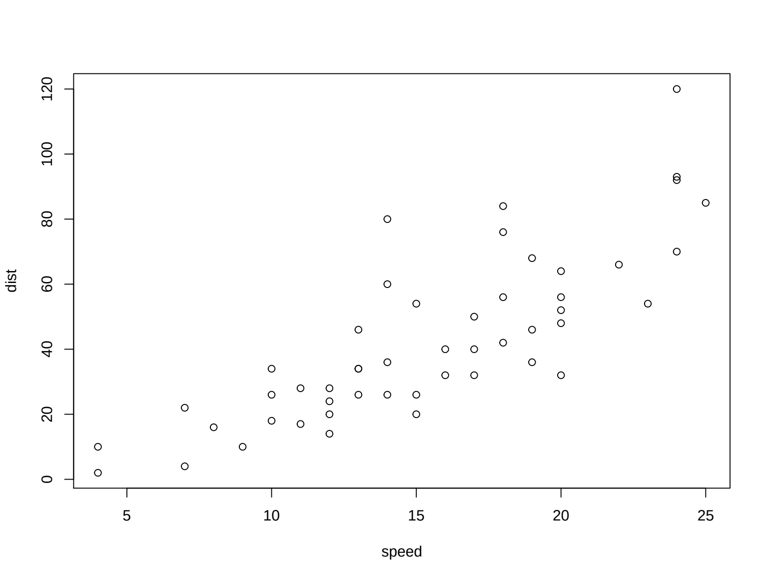
CV and Classification

- Explicitly, CV (k-fold or LOO) can be applied easily in a classification context as well.
- In these cases, we can calculate cross-validated misclassification rates for LOO
$$CV_n = \frac{\sum I(y_i \neq \hat{y}_i)}{n}$$
or in k-fold as
$$MC_j = \frac{1}{\sum_{i=1}^{n} I(i \in j)} \sum_{i \in j} I(y_i \neq \hat{y}_i) \quad \text{and} \quad CV_{(k)} = \frac{1}{k} \sum_{j=1}^{k} MC_j$$

CV and Classification

- These are the first systematic approaches we have for selecting among possible models!
- For example, remember the car speed vs stopping distance data.
- Now we can estimate which value of k for KNN fits the data best in the long term.

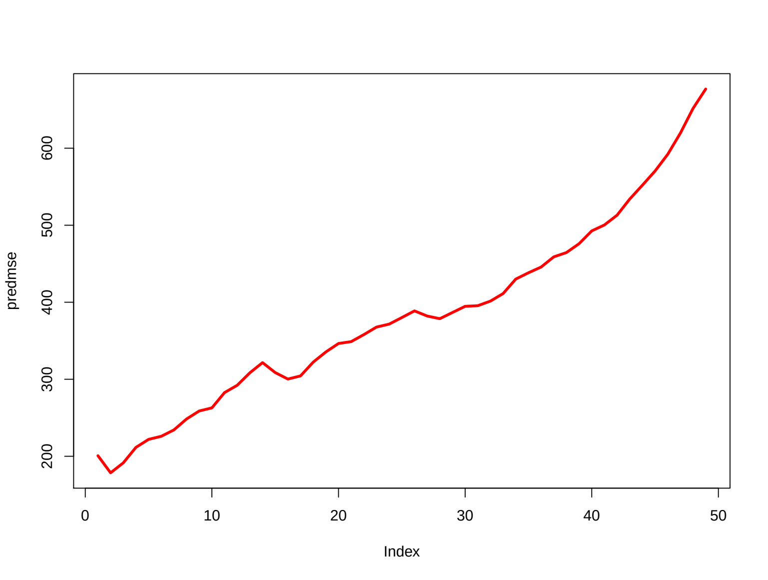
Cars example



Cars example

- We fit all possible k values (1 to 49 - the number of samples in the data)
- By default in R, knn.reg performs cross validation (help file doesn't specify what kind, but it is LOOCV)
- We plot the test, or predicted, MSE across all values of k.

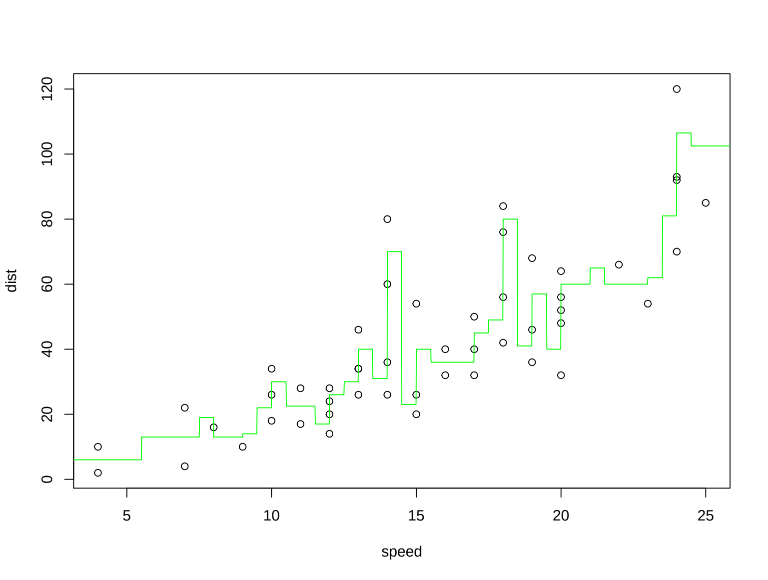
Cars example



Cars example

- The minimum of that plot suggests the best value of $k = 2$
- Which gives the following...

Cars example

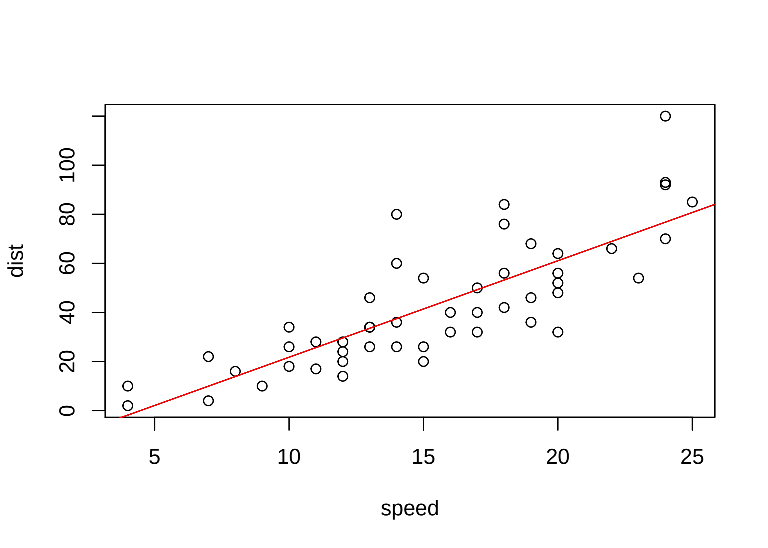


Cars example

- Not only can we select k using CV, but furthermore we can compare that best kNNreg model's predictive performance with any other potential model
- Since CV predicts the long-run MSE, there's no reason we cannot provide that prediction in the context of, say, simple linear regression...

Cars example

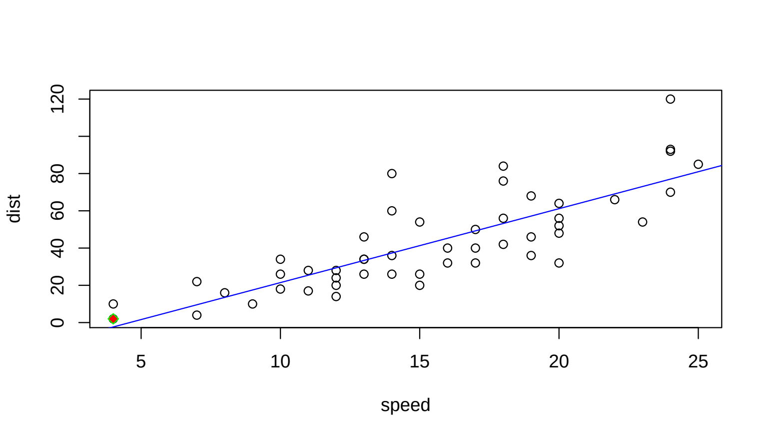
Car data, simple linear regression on full sample



Cars example

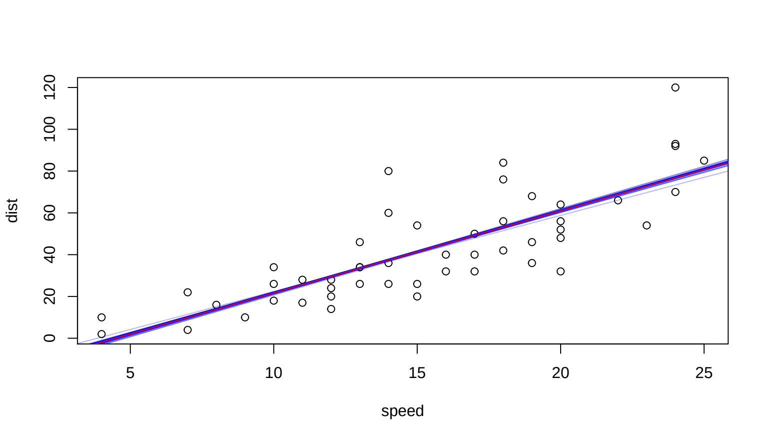
- Note that under the inferential assumptions for simple linear regression, we have a theoretical (unbiased) estimate of the true MSE via $\frac{\text{RSS}}{n-2}$
- For the cars data, that gives us: $\frac{11353.52}{48} = 236.53$
- Assuming all our diagnostics hold and we're willing to make those inferential assumptions, we could compare that value to a different model's CV predicted MSE. Note that kNNreg with $k = 2$ gives a predicted MSE of 178.54
- But for an even more direct comparison, we could of course get the CV MSE by applying LOOCV to the linear model!

LOOCV SLR on Cars



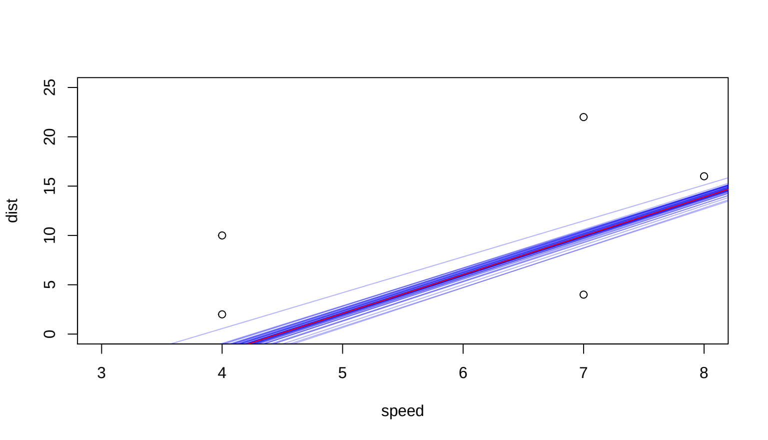
LOOCV SLR on Cars

- LOOCV gives $\hat{MSE} = 246.4$

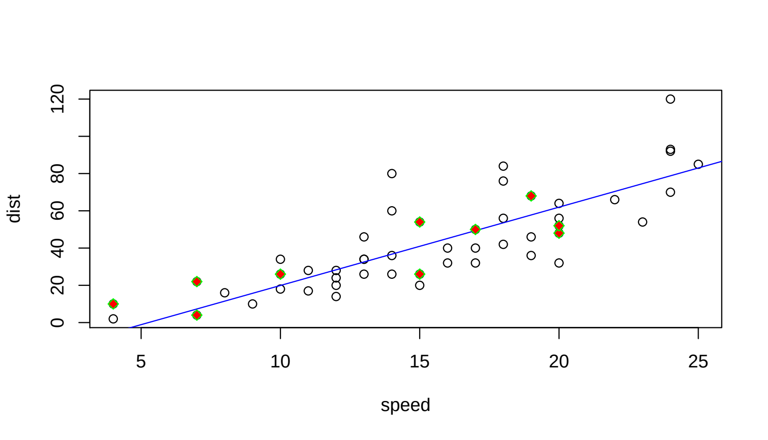


LOOCV SLR on Cars (zoomed in)

- LOOCV gives $\hat{MSE} = 246.4$

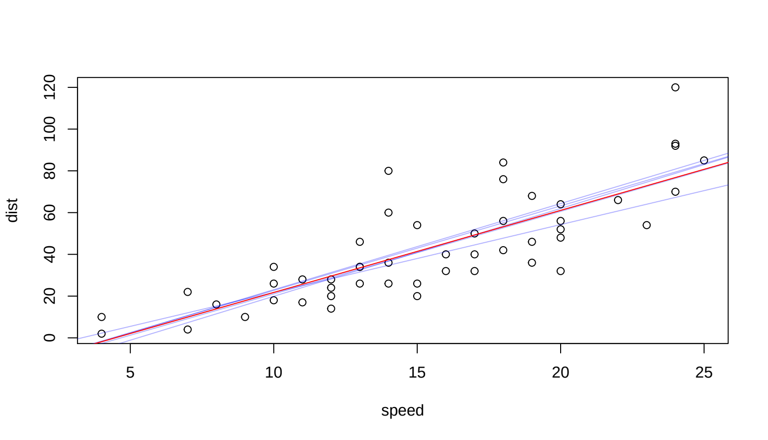


5-fold CV SLR on Cars



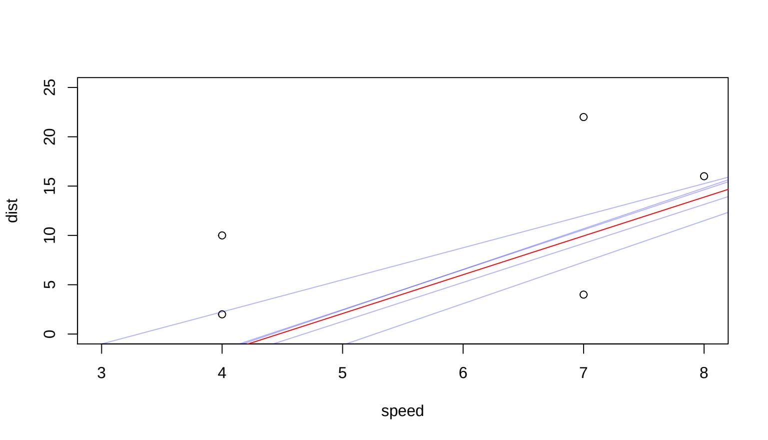
5-fold CV SLR on Cars

- (One run of) 5-fold CV gives $\hat{MSE} = 293.0$

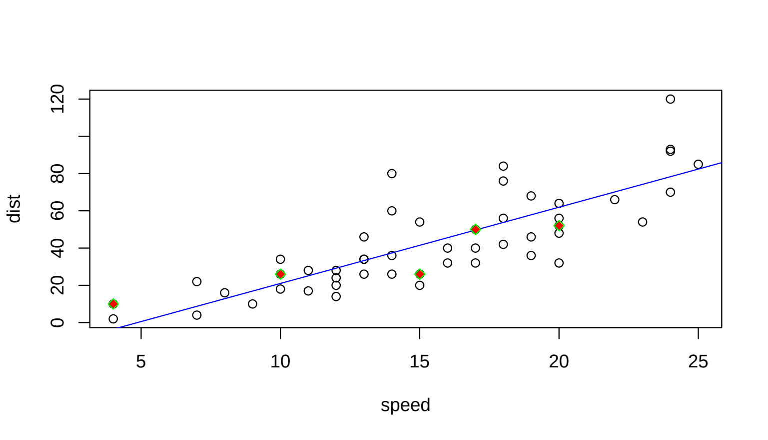


5-fold CV SLR on Cars (zoomed in)

- (One run of) 5-fold CV gives $\hat{MSE} = 293.0$

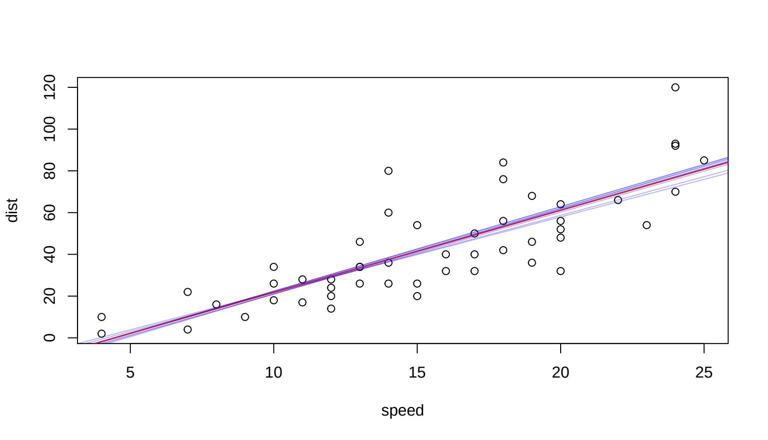


10-fold CV SLR on Cars



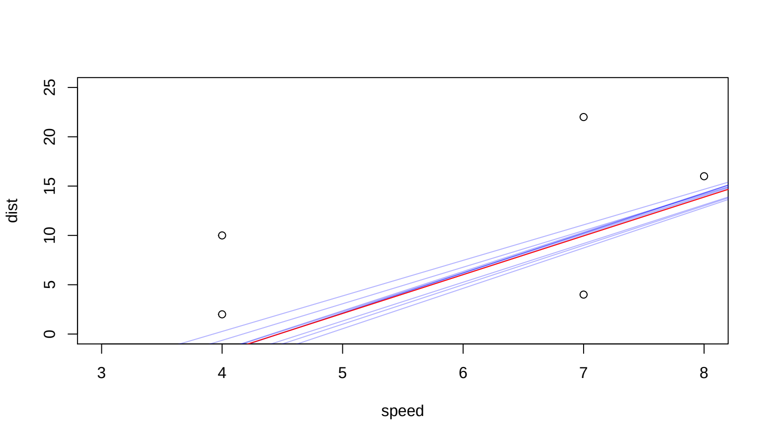
10-fold CV SLR on Cars

- (One run of) 10-fold CV gives $\hat{MSE} = 255.5$



10-fold CV SLR on Cars (zoomed in)

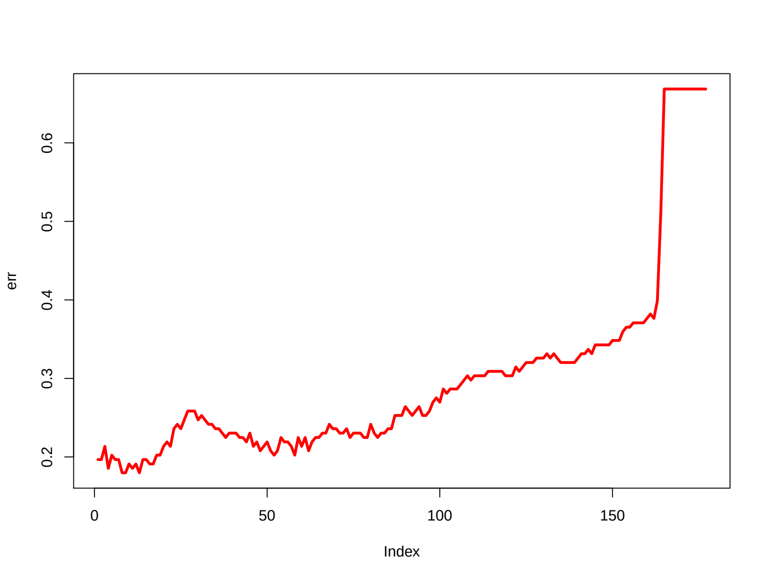
- (One run of) 10-fold CV gives $\hat{MSE} = 255.5$



Wine example

- 27 measurements on 178 samples of red wine. Samples originate from the same region of Italy (Piedmont), but are of different varietals (Barolo, Barbera, Grignolino).
- We can do KNN classification, using cross validation to choose the number of neighbours...

Wine example



Wine example

- In this case the minimum is at $k = 8$.
- We can provide a cross-validated classification table

	1	2	3
1	52	2	5
2	3	60	8
3	2	12	34

CV and Training-Testing

- Testing/validation is being used somewhat interchangeably in this lecture, which is a bit misleading
- As the models we discuss get more complex, with more tuning parameters to consider, it becomes more important how you train/validate$\rightarrow$test your models.
- Many approaches will use a training set, with CV implemented on that set to select tuning/hyper parameters, and then report the results of that model on a *completely new* test set.
- The critical importance is that a true ‘test set’ is NOT used to select any aspect of the trained model. Otherwise, the resulting error estimates are likely optimistic.

What is CV estimating?

- Avoiding notation/proofs/details (see ESL Ch 7 and a recent manuscript[1]), what does CV estimate?
- It turns out, it's good at estimating the long-run error of your *modeling process*
- Other words: "if I fit a linear model on any sample of the same size from this population, how might that perform in the longrun."
- This is different than "I fit my linear model on this sample, got $Y = 15.3 + 2.8X_1$...what's the expected long-run error of *THAT MODEL*"

[1] Bates, Hastie & Tibshirani (2023) 'Cross-validation: what does it estimate and how well does it do it?', *Journal of the American Statistical Association*

What is CV estimating?

- This also means that, in general, whatever processing you do to the data prior to analysis needs to be incorporated WITHIN the CV folds — not before.
- We'll explore this a little bit via Lab 1 and/or Assignment 1...


UBC
THE UNIVERSITY OF BRITISH COLUMBIA