Cross validation

- Data splitting
- Uncertainty for the expected test score
- Multiple splitting and cross validation
- R coding example and debugging

Proper scores and data splits

- Recall the the expectation of a score under a true distribution G,

$$S(F, G) = \mathsf{E}_{y \sim G}[S(F, y)].$$

- A (negatively oriented) score is *proper* if $S(F, G) \geq S(G, G)$ for all predictions F.

Before, we split the data in *observation* and *test*. Generalised split:

- Observation/Estimation/Training — Data used to estimate a model
- Validation — Data for assessing estimates and taking modelling decisions
- Test — Data used in a final step to assess the resulting model

Decisions based on scores evaluated on Validation data might lead to overestimation of the predictive ability.
Holding out a separate Test set provides a safer way of assessing predictive ability.

Basic score uncertainty

- We're interested in the expected average prediction test score $\overline{S}(F^{\text{test}}, G^{\text{test}})$
- We only have access to an estimate of $\overline{S}(F^{\text{test}}, G^{\text{test}})$, based on a training/validation data split:

$$\widehat{S}^{\text{valid}} = \frac{1}{N}\sum_{i=1}^{N} S(F_i^{\text{valid}}, y_i^{\text{valid}})$$

- Note: To investigate the *difference* in expected score between two models or methods, just replace $S(F_i, y_i)$ by the pairwise differences $S_\Delta(F_i, F_i', y_i) = S(F_i, y_i) - S(F_i', y_i)$ everywhere.
- The empirical variance estimate for $\widehat{S}^{\text{valid}}$ is

$$\widehat{\text{Var}}[\widehat{S}^{\text{valid}}] = \frac{1}{N(N-1)}\sum_{i=1}^{N}\left[S(F_i^{\text{valid}}, y_i^{\text{valid}}) - \widehat{S}^{\text{valid}}\right]^2$$

- The variance estimate may be biased due to dependence between the scores.

Cross validation

- We're interested in the expected average prediction test score $\overline{S}(F^{\text{test}}, G^{\text{test}})$ when using all the Training and Validation data to estimate the parameters of the final model.
- The Training set is a subset; may lead to overestimation of the expected score
- The Validation set is a small subset; high variability in the score estimator
- Different splits might give different score estimates and hence different modelling decisions
- Partial solution: Do multiple splits

K-fold Cross Validation: CV(K)

- Split the N data points $\mathcal{D}$ into K subsets $\mathcal{D}_k^{(\mathsf{K})}$, each of size N/K.
- Iterate over the K subsets, treating each as a Validation set, $\mathcal{D}_k^{\text{valid}} = \mathcal{D}_k^{(\mathsf{K})}$, and the remaining $K-1$ subsets as Training data $\mathcal{D}_k^{\text{train}} = \cup_{j\neq k}\mathcal{D}_j^{(\mathsf{K})}$.
- Average over the resulting K score estimates.

Cross-validation scores

- For each of the K *folds*, the estimator of the expected score is

$$\widehat{S}_k^{(\mathsf{K})} = \frac{K}{N} \sum_{i=1}^{N/K} S(F_{ki}^{\text{valid}}, y_{ki}^{\text{valid}})$$

- The combined cross-validation score is

$$\widehat{S}^{\mathsf{CV(K)}} = \frac{1}{K} \sum_{k=1}^{K} \widehat{S}_k^{(\mathsf{K})}$$

- There are many options for estimating the variance of the combined CV score. Simple:

$$\widehat{\mathsf{Var}}[\widehat{S}^{\mathsf{CV(K)}}] = \frac{1}{K(K-1)} \sum_{k=1}^{K} [\widehat{S}_k^{(\mathsf{K})} - \widehat{S}^{\mathsf{CV(K)}}]^2$$

- No universal rule for what K and splitting choices will minimise the bias and variance of the estimators. Common choice is $K = 10$ and random splitting.

Common problem-dependent splitting options

- Leave-one-out CV; LOOCV=CV(N)
  In general very expensive, but for some model classes fast approximations are possible;
  Notably in Gaussian time series and spatial models
- Structured, only partially random, cross-validation examples:
  - Leave-station-out (to assess spatial predictive ability)
  - Leave-country-out (to assess macro scale generalisability, including potentially different measurement systems)
  - Leave-timepoint out (to assess temporal interpolation ability)
  - Related non-cross-validation example: Leave-future-out (to assess forecasting ability)
- Instead of complete splitting, do multiple Validation subset selections as random subsamples with replacement (related to Bootstrap; lecture 7)

R live coding example and debugging

Need code to:

- Create random cross validation split
- Estimate each CV model
- Compute the validation score for each model
- Combine the results, make model selection
- (Compute the test score for the final model)

Some debugging tools:

- `traceback()`: Where did my code fail? What function calls did it use?
- `debugonce(fun)`: I'm feeling lucky and might find the error in `fun()` straight away!
- `debug(fun)`: The function `fun` with the problem is called in a loop and I need to run until I see the problem.
- `undebug(fun)`: Please stop debugging!
- `browser()`: Stop here and continue interactively inside the function!

```r
# Function for CV splitting:
create_split <- function(mydata, K) {
  indices <- rep(1:10, times = nrow(mydata) / 10, size = nrow(mydata))
  sample(indices, size = nrow(mydata), replace = FALSE)
}

# To simplify the example we shorten the data to a nice round number:
data <- TMINallobs[1:10000,]

# Try to construct a data split
thesplit <- create_split(data, 5)
unique(thesplit)
# [1] 1 2 3 4 5 6 7 8 9 10

# If this was someone else's function, we might use debugonce()
# to step into it to find out what's wrong (we expected 1,2,3,4,5):
debugonce(create_split)
create_split(data, 5)
# Press Enter to run each line in turn. See ?browser for more commands.
```