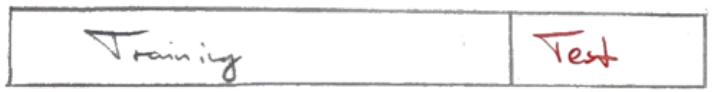
MSA 220 / UVE440

Lecture 3, 2019-01-29

Splitting data

Hold out method

Randomize order of samples before you do any splitting!

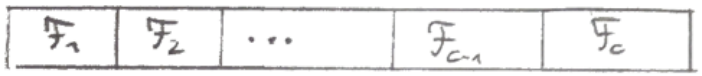


~> The model parameters are estimated from the training set.

~> The test set can be used to tune hyperparameters or assess model generalizability (see note after cross-validation below)

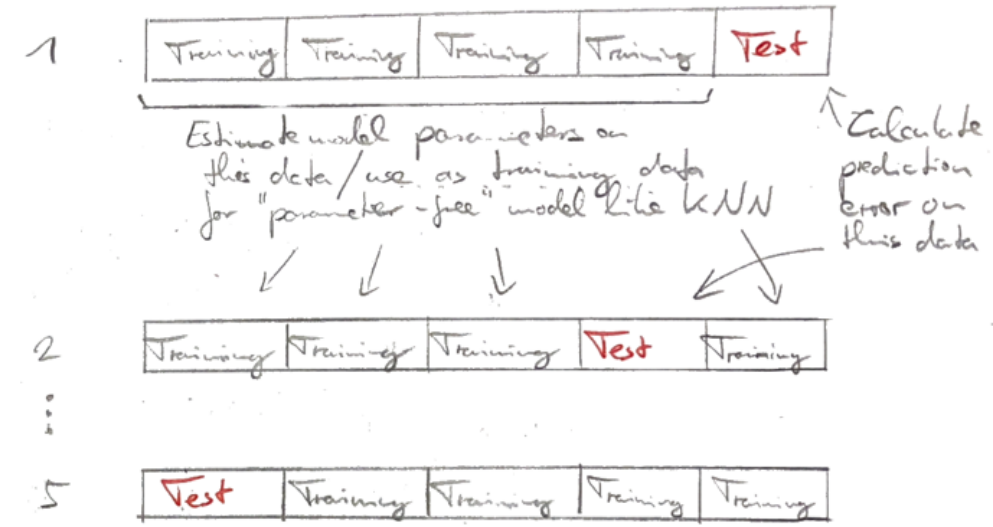
c-fold cross-validation

Randomize order of samples before splitting!



1) Divide randomized samples into c equally large subsets, so-called folds

Example: 5-fold cross validation



Notes on tuning parameter / method / model selection

We are looking at a hierarchy of estimation processes:

Training Data
- Estimation of individual methods/models
- Models with different tuning parameters can be seen as different models, e.g. kNN with $k=1$ as one model and $k=5$ another

Test Data
- Determination which tuning parameter/model/method leads to best performance on test data
- In a sense we are performing an estimation process here as well

How do we determine the generalizability of this selected model?

2 In an ideal, data rich world we could create another split:

Training	Validation	Test
Estimate individual models	Compare models/methods	Calculate generalizability of final model

- Prediction error on the test set should be interpreted in the context of the error returned in the validation step.

  If test error is much worse than validation then the model doesn't generalize well. However, this doesn't tell you what to change. A more in-depth model/data analysis is necessary.

- In a setting where we do not have enough data to split twice, cross-validation error is usually taken as a surrogate value.

  However note that model/method selection can overfit the test set and the prediction error from cross-validation might be lower than what you would get from a separate test set.

Stratification during data splitting

Classification example

Two class dataset

Random splitting of this dataset without paying attention to class labels could easily lead to a training set with only × and no ○ or a test set without ○

However, the underlying assumption of the hold out method and cross-validation is that training & test set come from the same distribution.

Having very different class frequencies violates this.

Stratification as a solution:

Here: frequency of ×: $\frac{24}{28} = \frac{6}{7}$

frequency of ○: $\frac{4}{28} = \frac{1}{7}$

Say we want to use 70% of the data for training & 30% for testing and we to maintain class frequencies:

28 data points → $0.7 \cdot 28 = 19.6$, i.e. 20 training points

→ $28 - 20 = 8$ test points

Then: $\frac{6}{7} \cdot 20 \approx 14.1$ i.e. 14 from class × } training

and 1 from class ○ }

Summary: 70/30 split per class! Not on the whole dataset

In test set: 7 from class ×

1 from class ○

[1]

Bias - Variance decomposition & tradeoff

For purposes of illustration assume that the response y fulfills $\mathbb{E}_{p(y|x)}[y] = f(x)$ for some function f, and $Var_{p(y|x)}[y] = \sigma^2$.

(This is like the standard regression model: $y = f(x) + \varepsilon$ with $\mathbb{E}[\varepsilon] = 0$, $Var[\varepsilon] = \sigma^2$)

Then: For the total expected prediction error with squared error loss holds

$$R = \mathbb{E}_{p(T)}\left[\mathbb{E}_{p(x,y)}\left[(y - \hat{f}(x))^2\right]\right] =$$

distribution over training samples (T); $\hat{f}$: estimated from data; regression function, giving a prediction at x

① x, y indep. of T ⇒ $p(T,x,y) = p(T)p(x,y)$

$$= \mathbb{E}_{p(T,x)}\left[\mathbb{E}_{p(y|x)}\left[(y - \hat{f}(x))^2\right]\right] =$$

② $p(T,x,y) = p(T,x)p(y|x)$

$$= \mathbb{E}_{p(T,x)}\left[\mathbb{E}_{p(y|x)}\left[(y - \mathbb{E}_{p(y|x)}[y] + \mathbb{E}_{p(y|x)}[y] - \hat{f}(x))^2\right]\right] =$$

$$= \mathbb{E}_{p(T,x)}\left[\mathbb{E}_{p(y|x)}\left[(y - \mathbb{E}_{p(y|x)}[y])^2\right]\right. +$$

$= Var_{p(y|x)}[y] = \sigma^2$

$$+ 2 \cdot \mathbb{E}_{p(y|x)}\left[(y - \mathbb{E}_{p(y|x)}[y])\right]\left(\mathbb{E}_{p(y|x)}[y] - \hat{f}(x)\right)$$

$= 0$ (*); constant w.r.t. $p(y|x)$

$$+ \left.(\mathbb{E}_{p(y|x)}[y] - \hat{f}(x))^2\right] =$$

$\mathbb{E}_{p(y|x)}[y] = f(x)$

$$= \sigma^2 + \mathbb{E}_{p(T,x)}\left[(f(x) - \hat{f}(x))^2\right] =$$

$$= \sigma^2 + \mathbb{E}_{p(T,x)}\left[(f(x) - \mathbb{E}_{p(T)}[\hat{f}(x)] + \mathbb{E}_{p(T)}[\hat{f}(x)] - \hat{f}(x))^2\right] =$$

$$= \sigma^2 + \mathbb{E}_{p(T,x)}\left[(f(x) - \mathbb{E}_{p(T)}[\hat{f}(x)])^2\right. +$$

Does not depend on T

$$+ 2\,(f(x) - \mathbb{E}_{p(T)}[\hat{f}(x)])(\mathbb{E}_{p(T)}[\hat{f}(x)] - \hat{f}(x)) +$$

zero in expectation like (*)

$$+ \left.(\mathbb{E}_{p(T)}[\hat{f}(x)] - \hat{f}(x))^2\right] =$$

$$= \sigma^2 + \mathbb{E}_{p(x)}\left[(f(x) - \mathbb{E}_{p(T)}[\hat{f}(x)])^2\right] +$$

$$+ \mathbb{E}_{p(T,x)}\left[(\hat{f}(x) - \mathbb{E}_{p(T)}[\hat{f}(x)])^2\right] \text{, therefore}$$

total pred. error

$$R = \sigma^2 + \mathbb{E}_{p(x)}\left[(f(x) - \mathbb{E}_{p(T)}[\hat{f}(x)])^2\right] + \mathbb{E}_{p(x)}\left[Var_{p(T)}[\hat{f}(x)]\right]$$

σ^2: irreducible error; $= \text{Bias}^2$; $=$ Variance of regressor / classifier

Bias^2 & Variance are averaged over all x values to get the 'total prediction error'