LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Introduction to Machine Learning

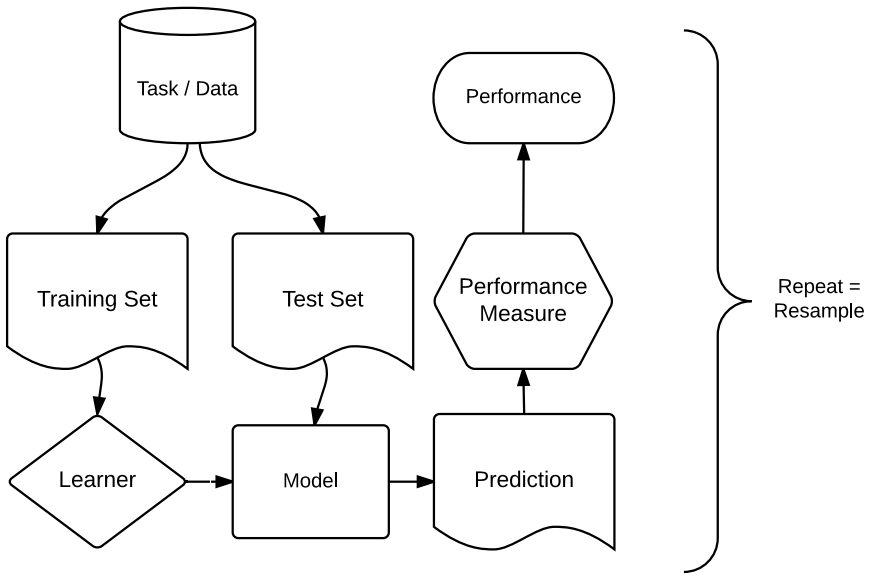
Evaluation: Resampling and Cross-Validation

Bernd Bischl, Christoph Molnar, Daniel Schalk, Fabian Scheipl

Department of Statistics – LMU Munich

RESAMPLING

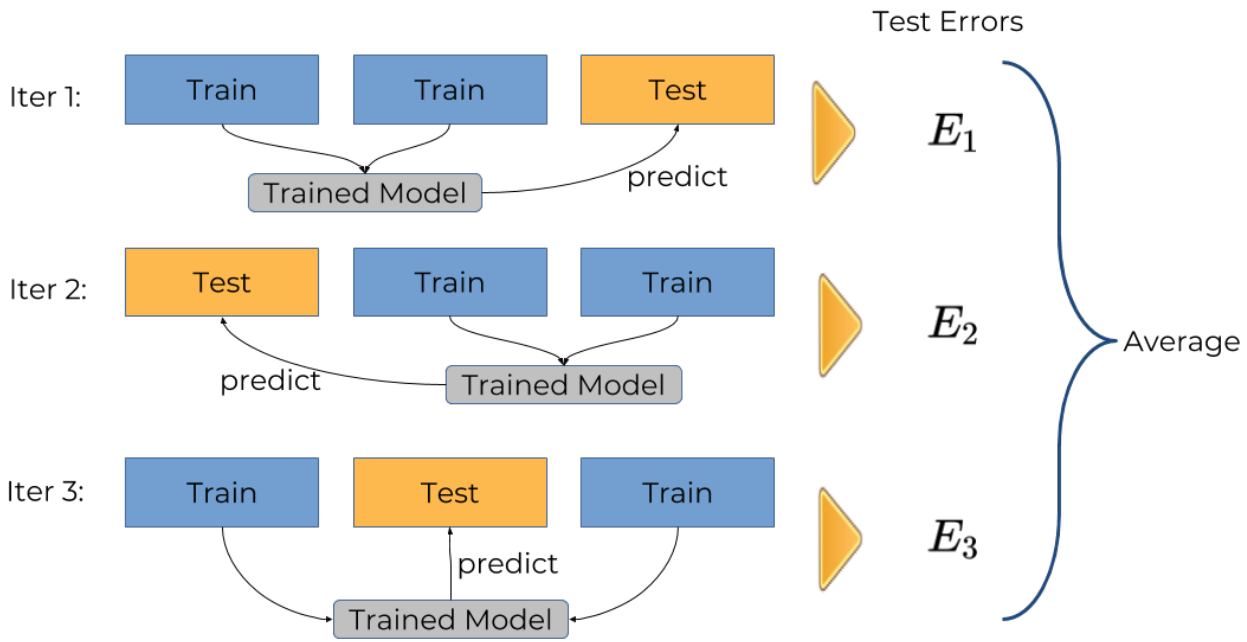
- Aim: Assess the performance of learning algorithm.
- Uses the data more efficiently then simple train-test.
- Repeatedly split in train and test, then average results.
- The usual trick is to make training sets quite larger (to keep the pessimistic bias small), and to handle the variance introduced by smaller test sets through many repetitions and averaging of results.



CROSS-VALIDATION

- Split the data into k roughly equally-sized partitions.
- Use each part once as test set and join the $k - 1$ others for training
- Obtain k test errors and average.

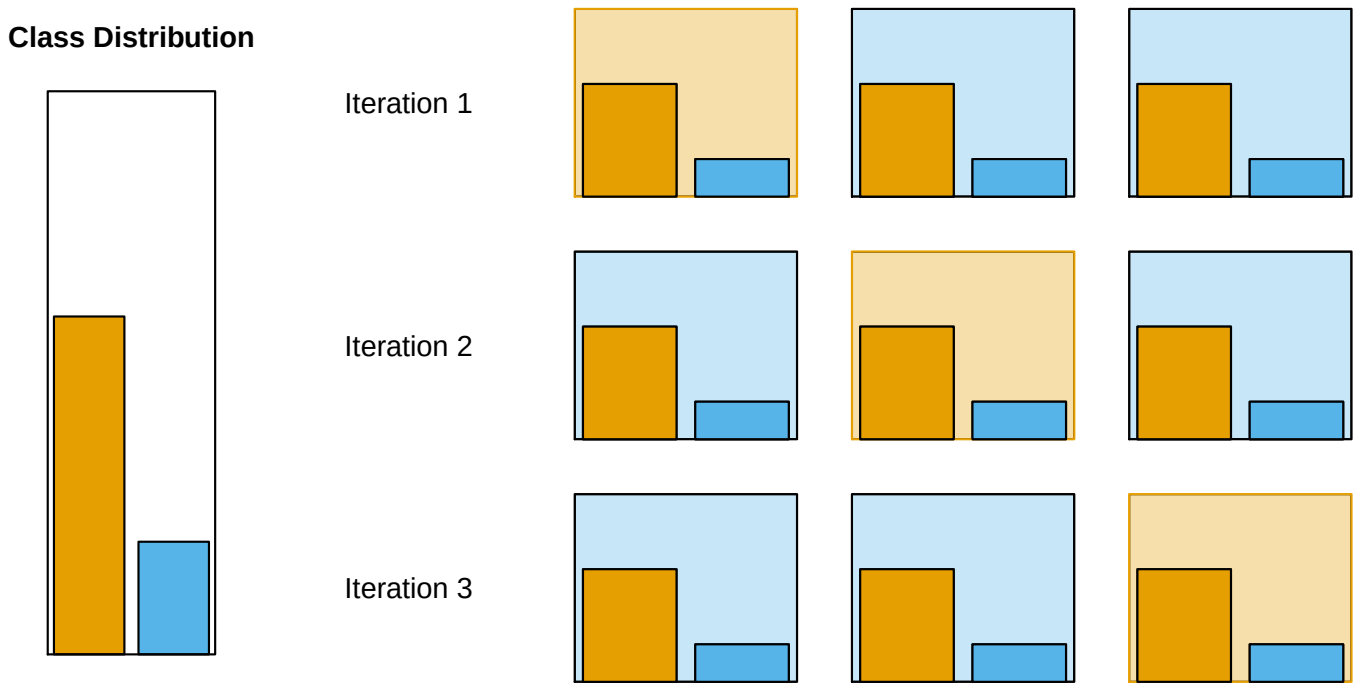
Example: 3-fold cross-validation:



CROSS-VALIDATION - STRATIFICATION

Stratification tries to keep the distribution of the target class (or any specific categorical feature of interest) in each fold.

Example of stratified 3-fold Cross-Validation:

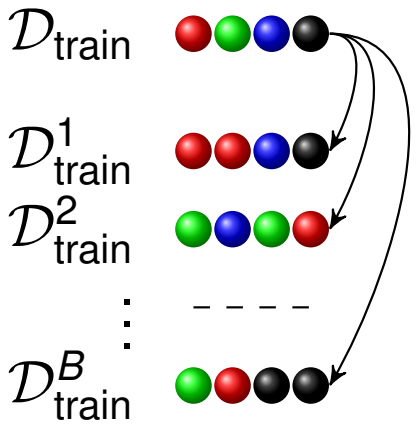


CROSS-VALIDATION - COMMENTS

- 5 or 10 folds are common, they use 80% and 90% of data in training
- $k = n$ is known as leave-one-out (LOO) cross-validation
- Estimates of the generalization error tend to be somewhat pessimistically biased (because the size of the training sets is $n - (n/k) < n$), bias increases as k gets smaller.
- The performance estimates for each fold are not independent, because of the structured overlap of the training sets. Hence, the variance of the estimator increases again for very large k (close to LOO), when training sets nearly completely overlap.
- LOO is nearly unbiased, but has high variance.
- Repeated k-fold CV (multiple random partitions) can improve error estimation for small sample size.

BOOTSTRAP

The basic idea is to randomly draw B training sets of size n with replacement from the original training set $\mathcal{D}_{\text{train}}$:



We define the test set in terms of out-of-bootstrap observations $\mathcal{D}^b_{\text{test}} = \mathcal{D}_{\text{train}} \setminus \mathcal{D}^b_{\text{train}}$.

BOOTSTRAP

- Typically, B is between 30 and 200.
- The variance of the bootstrap estimator tends to be smaller than the variance of k-fold CV, as training sets are independently drawn, discontinuities are smoothed out.
- The more iterations, the smaller the variance of the estimator.
- Tends to be pessimistically biased (because training sets contain only about 63.2% unique the observations).
- Bootstrapping framework might allow the use of formal inference methods (e.g. to detect significant performance differences between methods).
- Extensions exist for very small data sets, that also use the training error for estimation: B632 and B632+.

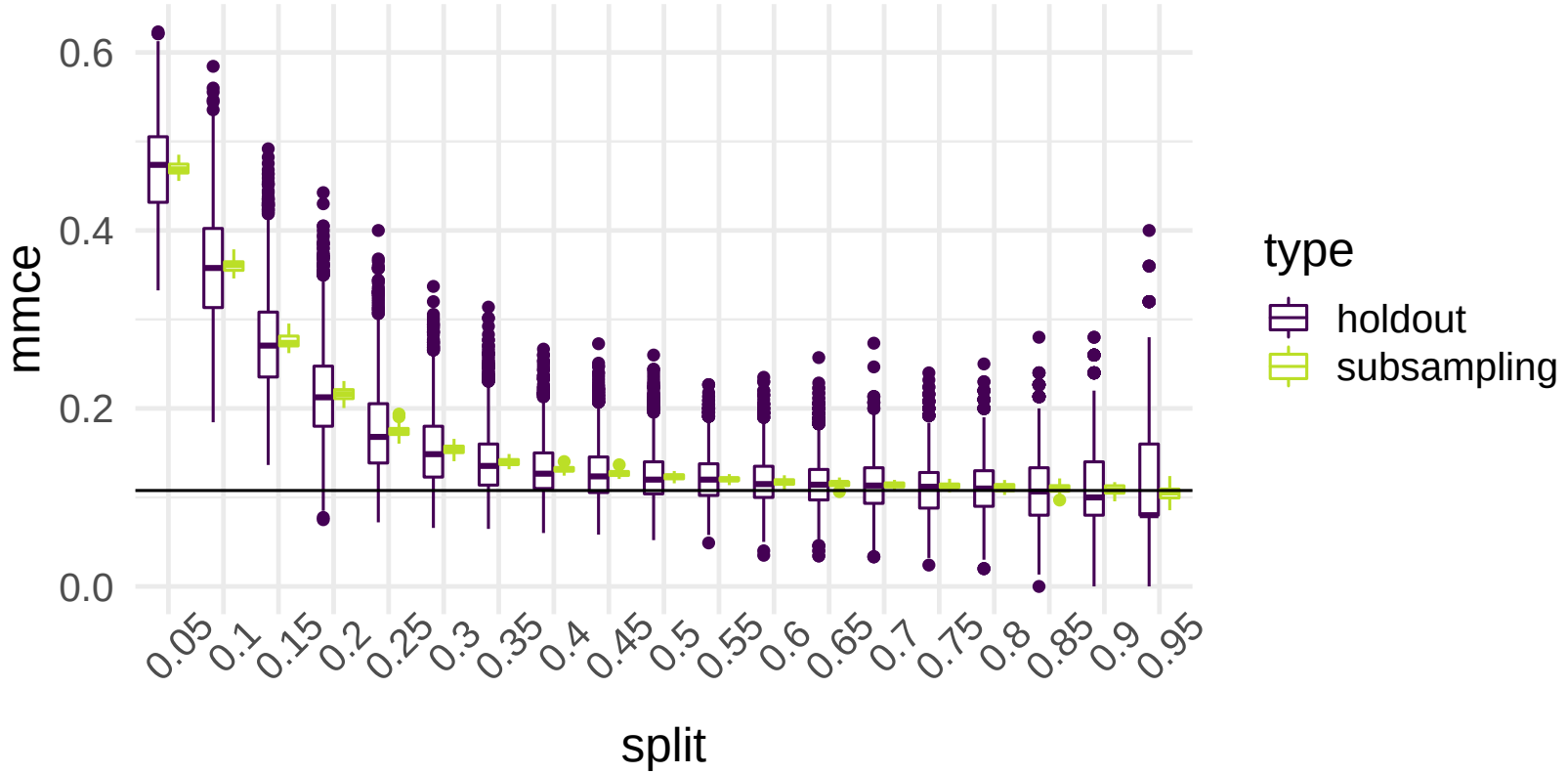
SUBSAMPLING

- Repeated hold-out with averaging, a.k.a. monte-carlo CV
- Similar to bootstrap, but draws without replacement, similar comments hold
- Typical choices for splitting: $4/5$ or $9/10$ for training
- The smaller the subsampling rate, the larger the pessimistic bias.
- The more subsampling iterations, the smaller the variance.

BIAS-VARIANCE ANALYSIS FOR SUBSAMPLING

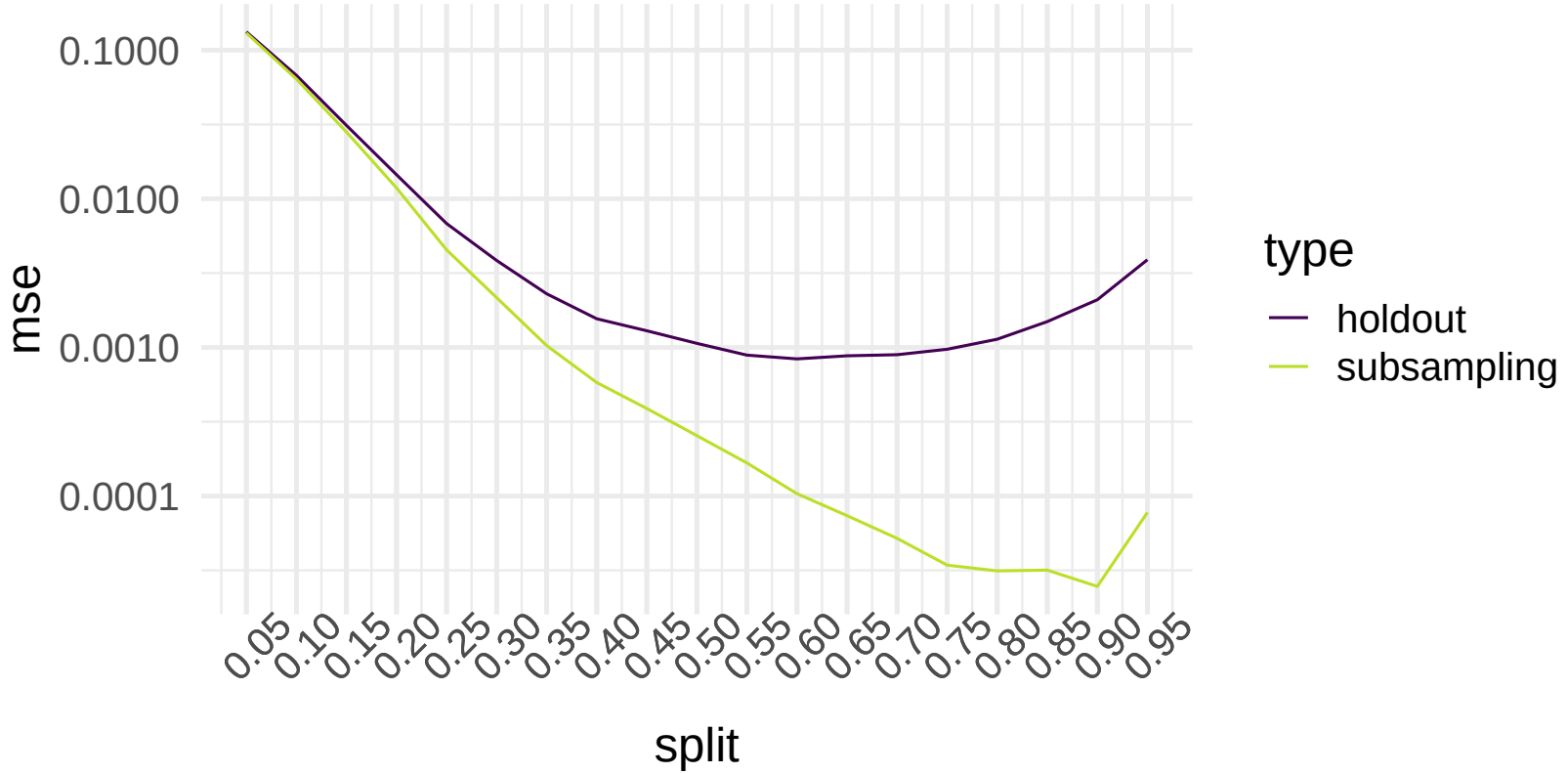
- Let's reconsider our hold-out experiment on the spiral data from the train-test unit (maybe re-read it again)
- Again, we use split-rates $s \in \{0.05, 0.1, ..., 0.95\}$ for training with $|\mathcal{D}_{\text{train}}| = s \cdot 500$.
- But now we perform 50 subsampling experiments instead of 50 hold-out experiments, so each performance estimate is much more reliable (but also more expensive)

BIAS-VARIANCE ANALYSIS FOR SUBSAMPLING



- Subsampling has the same pessimistic bias for small split rates, but much less variance overall
- This allows to use much smaller test sets with good results

BIAS-VARIANCE ANALYSIS FOR SUBSAMPLING



- The MSE is overall better for subsampling compared to hold-out
- The optimal split rate now is a higher $s \approx 0.9$
- We see an increase in variance at the end because the training sets become more overlapping and not independent

RESAMPLING DISCUSSION

- In ML we fit, at the end, a model on all our given data.
- Problem: We need to know how well this model performs in the future. But no data is left to reliably do this.
- In order to approximate this, we do the next best thing. We estimate how well the learner works when it sees nearly n points from the same data distribution.
- Holdout, CV, resampling estimate exactly this number. The "pessimistic bias" refers to when use much less data in fitting than n. Then we "hurt" our learner unfairly.
- Strictly speaking, resampling only produces one number, the performance estimator. It does NOT produce models, paramaters, etc. These are intermediate results and discarded.
- The model and parameters are obtained when we fit the learner finally on the complete data.
- This is a bit weird and complicated, but we have to live with this.

RESAMPLING DISCUSSION

- 5CV or 10CV have become standard
- Do not use Hold-Out, CV with few iterations, or subsampling with a low subsampling rate for small samples, since this can cause the estimator to be extremely biased, with large variance.
- For small data situation with less than 500 or 200 observations, use LOO or probably better repeated CV
- A $\mathcal{D}$ with $|\mathcal{D}| = 100.000$ can have small sample size properties if one class has only 100 observations ...
- For some models, computationally fast calculations or approximations for the LOO exist
- Modern results seem to indicate that subsampling has somewhat better properties than bootstrapping. The repeated observations can can cause problems in training algorithms, especially in nested setups where the “training” set is split up again.