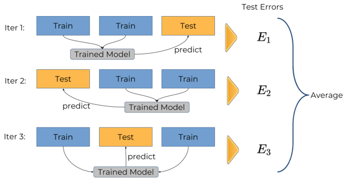
Introduction to Machine Learning

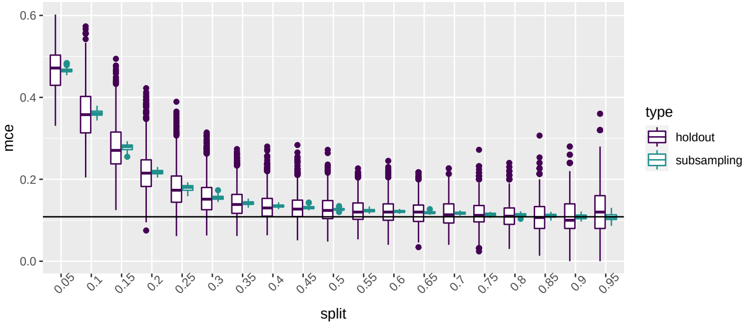
Evaluation
Resampling 2



Learning goals

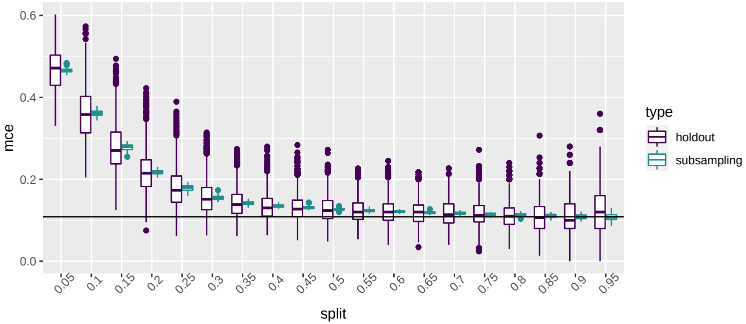
- Understand why resampling is better estimator than hold-out
- In-depth bias-var analysis of resampling estimator
- Understand that CV does not produce independent samples
- Short guideline for practical use

BIAS-VARIANCE ANALYSIS FOR SUBSAMPLING



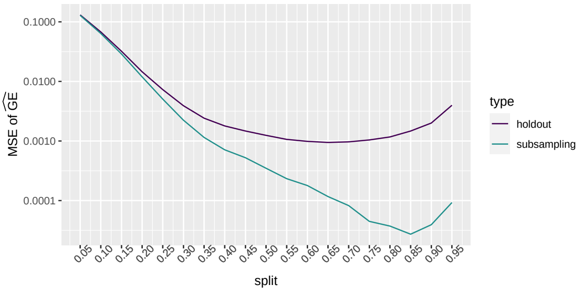
- Reconsider bias-var experiment for holdout (maybe re-read)
- Split rates $s \in \{0.05, 0.1, ..., 0.95\}$ with $|\mathcal{D}_{\text{train}}| = s \cdot 500$.
- Holdout vs. subsampling with 50 iters
- 50 replications

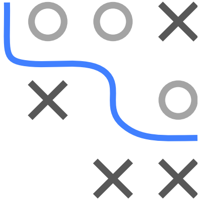
BIAS-VARIANCE ANALYSIS FOR SUBSAMPLING



- Both estimators are compared to "real" MCE (black line)
- SS same pessimistic bias as holdout for given s, but much less var

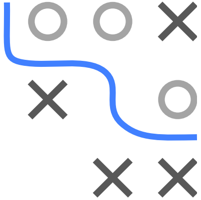
BIAS-VARIANCE ANALYSIS FOR SUBSAMPLING





- MSE of $\widehat{\mathrm{GE}}$ strictly better for SS
- Smaller var of SS enables to use larger *s* for optimal choice
- The optimal split rate now is a higher $s \approx 0.8$.
- Beyond $s = 0.8$: MSE goes up because var doesn't go down as much as we want due to increasing overlap in trainsets (see later)

DEDICATED TESTSET SCENARIO - ANALYSIS



- Goal: estimate $\mathrm{GE}\left(\hat{f}\right) = \mathbb{E}\left[L\left(y, \hat{f}(\mathbf{x})\right)\right]$ via

$$\widehat{\mathrm{GE}}\left(\hat{f}\right) = \frac{1}{m} \sum_{(\mathbf{x},y)\in\mathcal{D}_{\text{test}}} L\left(y, \hat{f}(\mathbf{x})\right)$$

  Here, only $(\mathbf{x}, y)$ are random, they are m i.i.d. fresh test samples
- This is: average over i.i.d $L(y, \hat{f}(\mathbf{x}))$, so directly know $\mathbb{E}$ and var. And can use CLT to approx distrib of $\widehat{\mathrm{GE}}\left(\hat{f}\right)$ with Gaussian.
- $\mathbb{E}[\widehat{\mathrm{GE}}\left(\hat{f}\right)] = \mathbb{E}[L\left(y, \hat{f}(\mathbf{x})\right)] = \mathrm{GE}\left(\hat{f}\right)$
- $\mathbb{V}[\widehat{\mathrm{GE}}\left(\hat{f}\right)] = \frac{1}{m}\mathbb{V}[L\left(y, \hat{f}(\mathbf{x})\right)]$
- So $\widehat{\mathrm{GE}}\left(\hat{f}\right)$ is unbiased estimator of $\mathrm{GE}\left(\hat{f}\right)$, var decreases linearly in testset size, have an approx of full distrib (can do NHST, CIs, etc.)
- NB: Gaussian may work less well for e.g. 0-1 loss, with $\mathbb{E}$ close to 0, can use binomial or other special approaches for other losses

PESSIMISTIC BIAS IN RESAMPLING

- Estim $\mathrm{GE}(\mathcal{I}, n)$ (surrogate for $\mathrm{GE}\left(\hat{f}\right)$ when $\hat{f}$ is fit on full $\mathcal{D}$, with $|\mathcal{D}| = n$) via resampling based estim $\widehat{\mathrm{GE}}(\mathcal{I}, n_{\mathrm{train}})$

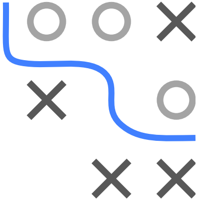
$$\widehat{\mathrm{GE}}(\mathcal{I}, \mathcal{J}, \rho, \boldsymbol{\lambda}) = \mathrm{agr}\Bigg(\rho\Big(\mathbf{y}_{J_{\mathrm{test},1}}, \boldsymbol{F}_{J_{\mathrm{test},1}, \mathcal{I}(\mathcal{D}_{\text{train},1}, \boldsymbol{\lambda})}\Big), \\ \vdots \\ \rho\Big(\mathbf{y}_{J_{\mathrm{test},B}}, \boldsymbol{F}_{J_{\mathrm{test},B}, \mathcal{I}(\mathcal{D}_{\text{train},B}, \boldsymbol{\lambda})}\Big)\Bigg),$$

- Let's assume agr is avg and ρ is loss-based, so ρ_L
- The ρ are simple holdout estims. So:

$$\mathbb{E}[\widehat{\mathrm{GE}}(\mathcal{I}, \mathcal{J}, \rho, \boldsymbol{\lambda})] \approx \mathbb{E}[\rho\Big(\mathbf{y}_{J_{\mathrm{test}}}, \boldsymbol{F}_{J_{\mathrm{test}}, \mathcal{I}(\mathcal{D}_{\text{train}}, \boldsymbol{\lambda})}\Big)]$$

- NB1: In above, as always for $\mathrm{GE}(\mathcal{I})$, both $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$ (and so $\mathbf{x} \in \mathcal{D}_{\text{test}}$) are random vars, and we take E over them
- NB2: Need $\approx$ as maybe not all train/test sets in resampling of exactly same size

PESSIMISTIC BIAS IN RESAMPLING / 2



$$\mathbb{E}[\widehat{\mathrm{GE}}(\mathcal{I}, \mathcal{J}, \rho, \boldsymbol{\lambda})] \approx \mathbb{E}[\rho\left(\mathbf{y}_{J_{\mathrm{test}}}, \boldsymbol{F}_{J_{\mathrm{test}}, \mathcal{I}(\mathcal{D}_{\text{train}}, \boldsymbol{\lambda})}\right)] =$$

$$\mathbb{E}\left[\frac{1}{m} \sum_{(\mathbf{x}, y) \in \mathcal{D}_{\text{test}}} L(y, \mathcal{I}(\mathcal{D}_{\text{train}})(\mathbf{x}))\right] = \mathrm{GE}(\mathcal{I}, n_{\mathrm{train}})$$

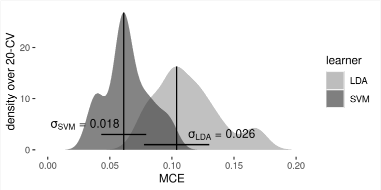
$\Rightarrow$

- So when we use $\widehat{\mathrm{GE}}(\mathcal{I}, \mathcal{J}, \rho, \boldsymbol{\lambda})$ to to estimate $\mathrm{GE}(\mathcal{I}, n)$, our expected value is nearly correct, it's $\mathrm{GE}(\mathcal{I}, n_{\mathrm{train}})$
- But fitting $\mathcal{I}$ on less data (n_{train} vs full n) usually results in model with worse perf, hence estimator is pessimistically biased
- Bias the stronger, the smaller our training splits in resampling.

NO INDEPENDENCE OF CV RESULTS

- Similar analysis as before holds for CV
- Might be tempted to report distribution or SD of individual CV split perf values, e.g. to test if perf of 2 learners is significantly different
- But k CV splits are not independent

A t-test on the difference of the mean GE estimators yields a highly significant p-value of $\approx 7.9 \cdot 10^{-5}$ on the 95% level.

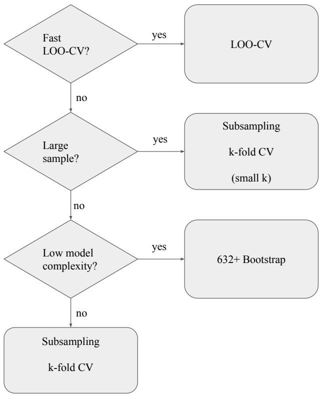


LDA vs SVM on `spam` classification problem, performance estimation via 20-CV w.r.t. MCE.

NO INDEPENDENCE OF CV RESULTS

- $\mathbb{V}[\widehat{\mathrm{GE}}]$ of CV is a difficult combination of
  - average variance as we estim on finite trainsets
  - covar from test errors, as models result from overlapping trainsets
  - covar due to the dependence of trainsets and test obs appear in trainsets
- Naively using the empirical var of *k* individual $\widehat{\mathrm{GE}}$s (as on slide before) yields biased estimator of $\mathbb{V}[\widehat{\mathrm{GE}}]$. Usually this underestimates the true var!
- Worse: there is no unbiased estimator of $\mathbb{V}[\widehat{\mathrm{GE}}]$ [Bengio, 2004]
- Take into account when comparing learners by NHST
- Somewhat difficult topic, we leave it with the warning here

SHORT GUIDELINE



- 5-CV or 10-CV have become standard.
- Do not use hold-out, CV with few folds, or SS with small split rate for small n. Can bias estim and have large var.
- For small n, e.g. $n < 200$, use LOO or, probably better, repeated CV.
- For some models, fast tricks for LOO exist
- With $n = 100.000$, can have "hidden" small-sample size, e.g. one class very small
- SS usually better than bootstrapping. Repeated obs can cause problems in training, especially in nested setups where the “training” set is split up again.