AutoML: Evaluation

Benchmarking and Comparing Learners

Bernd Bischl Frank Hutter Lars Kotthoff
Marius Lindauer Joaquin Vanschoren

Benchmark Experiments

- different learning algorithms applied to one or more data sets to compare and rank their performances
- synchronized train and test sets, i.e. the same resampling method with the same train-test splits should be used to determine performance
  Example: Benchmark results (per CV-fold) of CART and random forest using 2-fold CV with MSE as performance measure:

data set	k-th fold	MSE (rpart)	MSE (randomForest)
BostonHousing	1	29.4	17.13
BostonHousing	2	20.5	8.90
mtcars	1	35.0	7.53
mtcars	2	38.9	6.73

Hypothesis Testing in Benchmarking I

We want to know if the difference in performance between models (or algorithms) is significant or only by chance.

Null Hypothesis Statistical Testing (NHST) in Benchmarking:

- formulate null hypothesis H_0 (e.g. the expected generalization error of two algorithms is equivalent) and alternative hypothesis H_1
- use hypothesis test to reject H_0 (not rejecting H_0 does not mean that we accept it)
- rejecting H_0 gives some confidence that the observed results may not be random

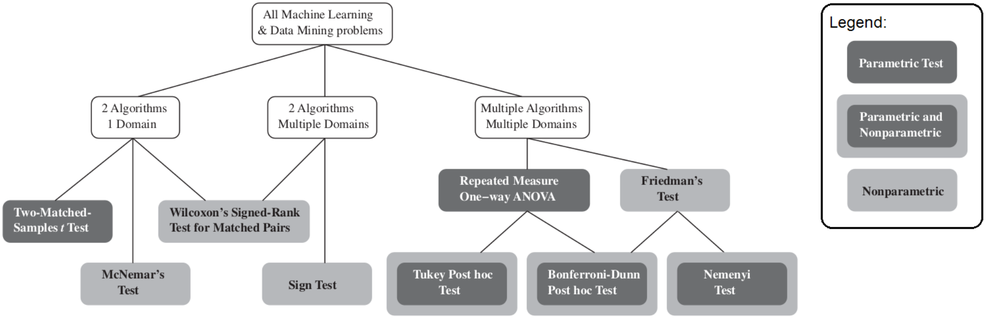
Typical example in machine learning:

- H_0: on average, model 1 does not perform better than model 2
- H_1: on average, model 1 outperforms model 2
- Aim: Reject H_0 with confidence level of $1 - \alpha$ (common values for α include 0.05 and 0.01)

Hypothesis Testing in Benchmarking II

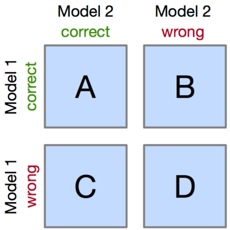
Selection of an appropriate hypothesis test is at least based on the type of problem, i.e. if the aim is to compare

- 2 models / algorithms on a single domain (i.e. on a single data set)
- 2 algorithms across different domains (i.e. on multiple data sets)
- multiple algorithms across different domains / data sets



McNemar Test I

- non-parametric test used on paired dichotomous nominal data; does not make any distributional assumptions beyond statistical independence of samples
- pairs are e.g. labels predicted by different models on the same data
- compares the classification accuracy of two **models**
- both models trained and evaluated on the exact same training and test set; **contingency table** based on two paired vectors that indicate whether each model predicted an observation correctly



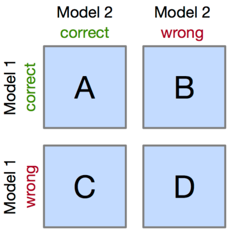
- A: #obs. correctly classified by both
- B: #obs. only correctly classified by model 1
- C: #obs. only correctly classified by model 2
- D: #obs. misclassified by both

McNemar Test II

Error of each model can be computed as follows:

- Model 1: (C+D)/(A+B+C+D)
- Model 2: (B+D)/(A+B+C+D)

Even if the models have the **same** errors (indicating equal performance), cells B and C may be different because the models may misclassify different instances.



McNemar tests the following hypothesis:

- H_0 : both models have the same performance (we expect B = C)
- H_1 : performances of the two models are not equal

The test statistic is computed as

$$\chi^2_{Mc} = \frac{(|B-C|-1)^2}{B+C} \sim \chi^2_1.$$

Note: The McNemar test should only be used if $B + C > 20$.

McNemar Test III

Example:

		Random Forest 0	Random Forest 1
Tree	0	30	5
	1	17	42

Calculating the test statistic:

$$\chi^2_{Mc} = \frac{(|5-17|-1)^2}{5+17} = 5.5 > 3.841 = \chi^2_{1,0.95}$$

We can reject H_0 at a significance level of 0.05, i.e. we reject the hypothesis that the tree and the random forest have the same performance.
Significance level must be chosen before applying the test (avoid p-value hacking).

Two-Matched-Samples t-Test I

- two-matched-samples t-test (i.e. a paired t-test) is the simplest hypothesis test to compare two **models** on a single test set based on arbitrary performance measures
- parametric test and distributional assumptions must be made (which are often problematic):

  (pseudo-)normality usually met when sample size > 30

  i.i.d. samples usually met if loss of individual observations from single test set considered

  equal variances of populations can be investigated through plots

Two-Matched-Samples t-Test II

Compare two different models $\hat{f}_1$ and $\hat{f}_2$ w.r.t. performance measure calculated on test set of size n_{test}:

- H_0: $GE(\hat{f}_1) = GE(\hat{f}_2)$ vs. H_1: $GE(\hat{f}_1) \neq GE(\hat{f}_2)$
- test statistic $T = \sqrt{n_{\text{test}}}\frac{\bar{d}}{\sigma_d}$ where
  - mean performance difference of both models is $\bar{d} = \hat{GE}_{\mathcal{D}_{\text{test}}}(\hat{f}_1) - \hat{GE}_{\mathcal{D}_{\text{test}}}(\hat{f}_2)$
  - standard deviation of this mean difference is

$$\sigma_d = \sqrt{\frac{1}{n_{\text{test}} - 1} \sum_{i=1}^{n_{\text{test}}} \left(d_i - \bar{d}\right)^2},$$

where $d_i = L(y^{(i)}, \hat{f}_1(\mathbf{x}^{(i)})) - L(y^{(i)}, \hat{f}_2(\mathbf{x}^{(i)}))$ and $\bar{d} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} d_i$

Note: d_i is the difference of the outer loss of individual observations from the test set between the two models to be compared.

Two-Matched-Samples t-Test III

- could also use a k-**fold CV paired t-test** to compare two **algorithms** (instead of two models) on single data set
- instead of comparing outer loss of individual observations, compare generalization errors per CV fold (i.e. k generalization errors for k CV folds)
- performance differences are not independent across CV folds due to overlapping training sets (which violates the assumption of i.i.d. samples)
- to partly overcome issue of overlapping training sets across folds, Dietterich suggests using 5 times 2-fold CV so that at least within each repetition neither training nor test sets overlap [Dietterich. 1998]

Friedman Test I

Compare multiple classifiers on multiple data sets:

- H_0 : all algorithms are equivalent in their performance and hence their average ranks should be equal
- H_1 : the average ranks for at least one algorithm is different

To evaluate n data sets and k algorithms:

- rank each algorithm on each data set from best-performing algorithm (rank 1) to worst-performing algorithm using any performance measure
- R_{ij} is the rank of algorithm j on data set i
- if there is a d-way tie after rank r, assign rank of $\left[(r+1)+(r+2)+...+(r+d)\right]/d$ to each tied classifier

Friedman Test II

Can now compute:

- overall mean rank $\bar{R} = \frac{1}{nk} \sum_{i=1}^{n} \sum_{j=1}^{k} R_{ij}$
- sum of squares total $SS_{Total} = n \sum_{j=1}^{k} (\bar{R}_{.j} - \bar{R})^2$ where $\bar{R}_{.j} = \frac{1}{n} \sum_{i=1}^{n} R_{ij}$
- sum of squares error $SS_{Error} = \frac{1}{n(k-1)} \sum_{i=1}^{n} \sum_{j=1}^{k} (R_{ij} - \bar{R})^2$

Test statistic calculated as:

$$\chi_F{}^2 = \frac{SS_{Total}}{SS_{Error}} \sim \chi^2_{k-1} \text{ for large n (>15) and k (>5)}$$

For smaller n and k, the χ^2 approximation is imprecise and a look up of χ^2_F values that were approximated specifically for the Friedman test is suggested.

Post-Hoc Tests I

- Friedman test checks if all algorithms are ranked equally
- does not allow to identify best-performing algorithm

$\rightarrow$ post-hoc tests

Post-hoc Nemenyi test:

- compares all pairs of algorithms to find best-performing algorithm after H_0 of the Friedman-test was rejected
- for n data sets and k algorithms, $\frac{k(k-1)}{2}$ comparisons
- calculate average rank of algorithm j on all n data sets: $\bar{R}_{.j} = \frac{1}{n} \sum_{i=1}^{n} R_{ij}$

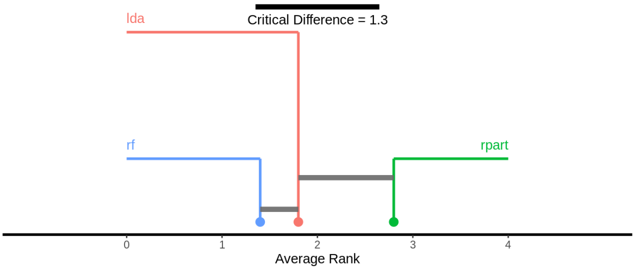
For any two algorithms j_1 and j_2, test statistic computed as:

$$q = \frac{\bar{R}_{.j_1} - \bar{R}_{.j_2}}{\sqrt{\frac{k(k+1)}{6n}}}$$

Post-Hoc Tests II

Critical Difference Plot:

- quick way to see what differences are significant across all compared learners
- all learners that do not differ by at least the critical difference are connected by line
- a learner not connected to another learner and of lower rank can be considered better according to the chosen significance level



Post-Hoc Tests III

Post-hoc Bonferonni-Dunn test:

- compares all algorithms with baseline (i.e. $k-1$ comparisons)
- used after Friedman test to find which algorithms differ from the baseline significantly
- uses Bonferonni correction to prevent randomly accepting one of the algorithms as significant due to multiple testing

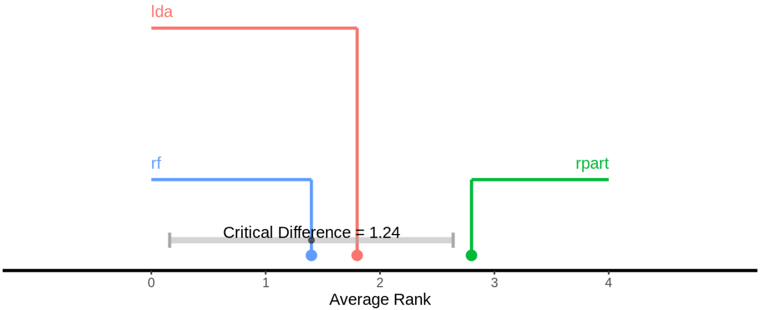
The test statistic is the same as before:

$$q = \frac{\bar{R}_{.j_1} - \bar{R}_{.j_2}}{\sqrt{\frac{k(k+1)}{6n}}}.$$

The performance of j_1 and j_2 are significantly different when $|q| > q_\alpha$, where the critical value q_α is obtained from a table of the studentized range statistic divided by $\sqrt{2}$.

Post-Hoc Tests IV

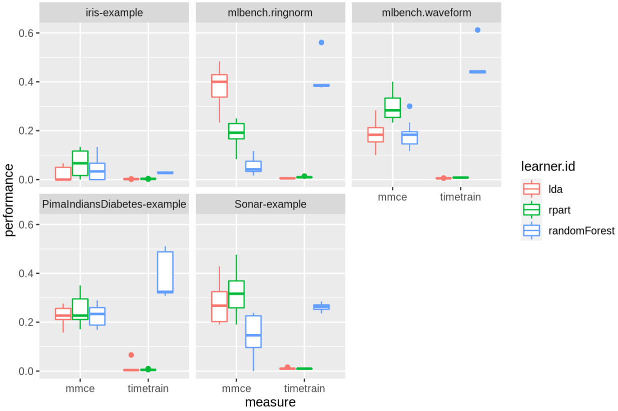
- learners within the baseline interval (gray line) perform not significantly different from the baseline



Comparing Visually I

It can be helpful to inspect distributions visually for additional insights, e.g.

Boxplots



Comparing Visually II

Rank plots