Same vs. other cross-validation in supervsied machine learning

Toby Dylan Hocking
toby.hocking@nau.edu

April 8, 2024

Introduction to machine learning

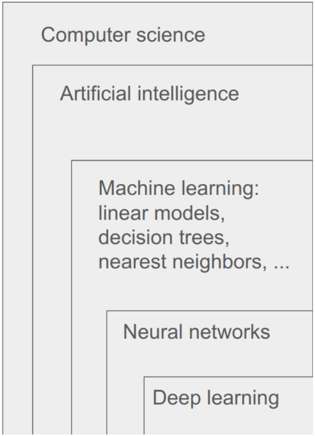
Proposed same vs. other cross-validation

Results on real data sets

Discussion and Conclusions

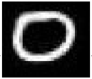
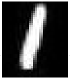
What is machine learning?

- Computer science: domain of study about efficient algorithms / computations.
- Artificial intelligence: sub-domain concerned with algorithms for accurate predictions/suggestions.
- Machine learning: sub-domain concerned with algorithms for large data.
- Machine learning is widely used in search engines, automatic translation, image analysis, ...



Machine learning intro: image classification example

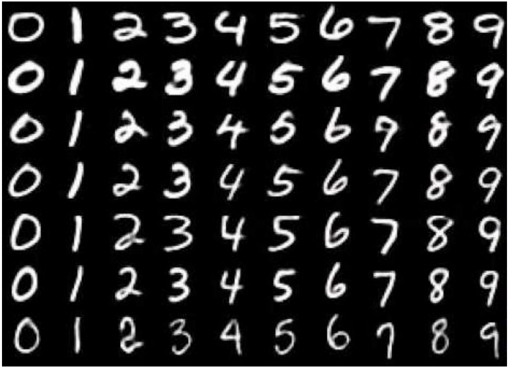
ML is all about learning predictive functions $f(x) \approx y$, where

- Inputs/features x can be easily computed using traditional algorithms, e.g. matrix of pixel intensities in an image.
- Outputs/labels y are what we want to predict, easy to get by asking a human, but hard to compute using traditional algorithms, e.g. image class.
- Input $x =$ image of digit, output $y \in \{0, 1, \ldots, 9\}$,
  – this is a classification problem with 10 classes.
  $f($$) = 0$, $f($$) = 1$
- Traditional/unsupervised algorithm: I give you a pixel intensity matrix $x \in \mathbb{R}^{16\times 16}$, you code a function f that returns one of the 10 possible digits. Q: how to do that?

Supervised machine learning algorithms

I give you a training data set with paired inputs/outputs, e.g.

$$y = 0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9$$

$X =$ 

Your job is to code an algorithm that learns the function f from the training data. (you don't code f)

Source: github.com/cazala/mnist

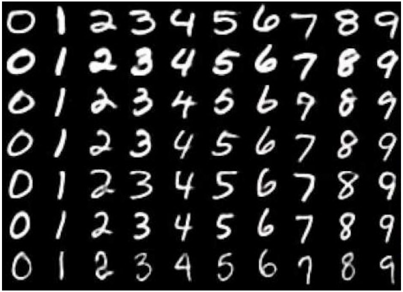
Supervised machine learning algorithms

Can be used whenever a knowledgeable/skilled human can easily/quickly/consistently create a large database of labels for training.

Should be used if it is not easy to code the function f for predicting the labels (using traditional/unsupervised techniques).

Accurate if the test data, on which you want to use f, is similar to the train data (input to learning algorithm).

Advantages of supervised machine learning

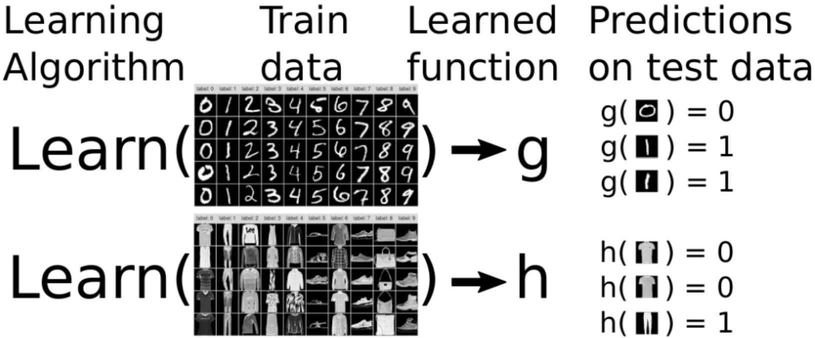


- Input $x \in \mathbb{R}^{16\times 16}$, output $y \in \{0, 1, \ldots, 9\}$ types the same!
- Can use same learning algorithm regardless of pattern.
- Pattern encoded in the labels (not the algorithm).
- Useful if there are many un-labeled data, but few labeled data (or getting labels is long/costly).
- State-of-the-art accuracy (if there is enough training data).

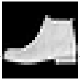
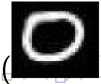
Sources: github.com/cazala/mnist, github.com/zalandoresearch/fashion-mnist

Learning two different functions using two data sets

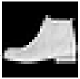
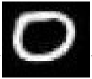
Figure from chapter by Hocking TD, *Introduction to machine learning and neural networks* for book *Land Carbon Cycle Modeling: Matrix Approach, Data Assimilation, and Ecological Forecasting* edited by Luo Y (Taylor and Francis, 2022).



Learn is a learning algorithm, which outputs g and h.

Q: what happens if you do $g($$)$, or $h($$)$?

Learning two different functions using two data sets

- What if you do $g($$)$, or $h($$)$?
- This is a question about **generalization**: how accurate is the learned function on a new/test data set?
- "Very accurate" if test data are similar enough to train data (best case is i.i.d. = independent and identically distributed)
- Predicting childhood autism (Lindly *et al.*), train on one year of surveys, test on another.
- Predicting carbon emissions (Aslam *et al.*), train on one city, test on another.
- Predicting presence of trees/fires in satellite imagery (Shenkin *et al.*, *Thibaut et al.*), train on one geographic area/image, test on another.
- Predicting fish spawning habitat in sonar imagery (Bodine *et al.*), train on one river, test on another.
- But how do we check if "very accurate" in these situations?

Introduction to machine learning

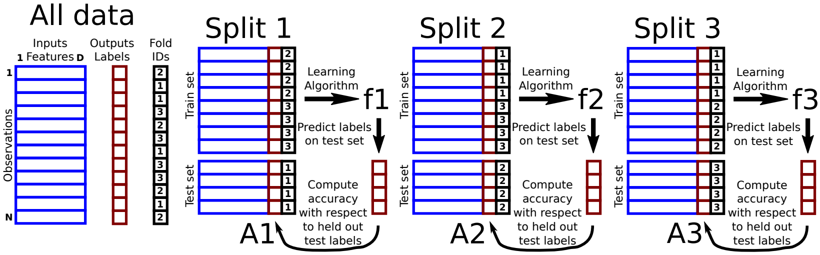
Proposed same vs. other cross-validation

Results on real data sets

Discussion and Conclusions

K-fold cross-validation: a standard algorithm used to estimate the prediction accuracy in machine learning

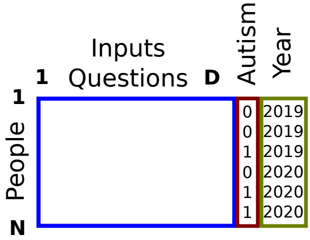
- $K = 3$ folds shown in figure below, meaning three different models trained, and three different prediction/test accuracy rates computed.
- It is important to use several train/test splits, so we can see if there are statistically significant differences between algorithms.
- Rows/observations are people, inputs/features are survey questions, and output/label is Autism response (Yes or No).



Hocking TD *Intro. to machine learning and neural networks* (2022).

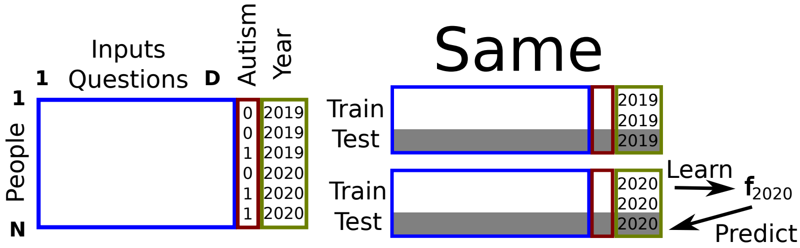
Proposed Same Other Cross-Validation

- Example: childhood autism prediction data set.



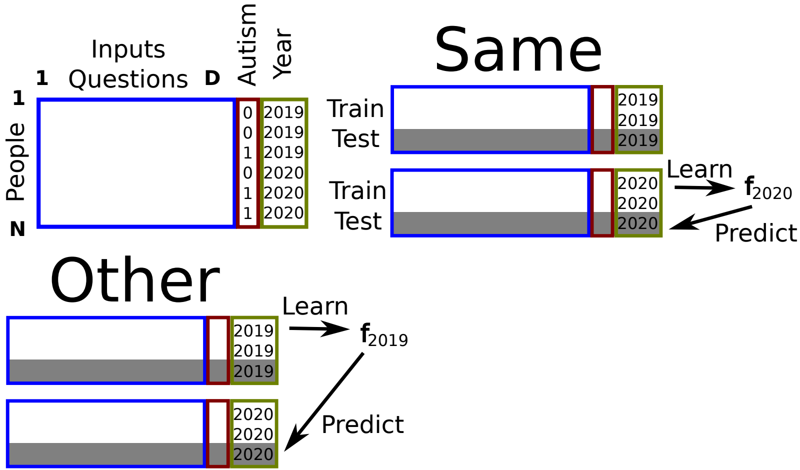
Proposed Same Other Cross-Validation

- One train/test split shown out of K folds.



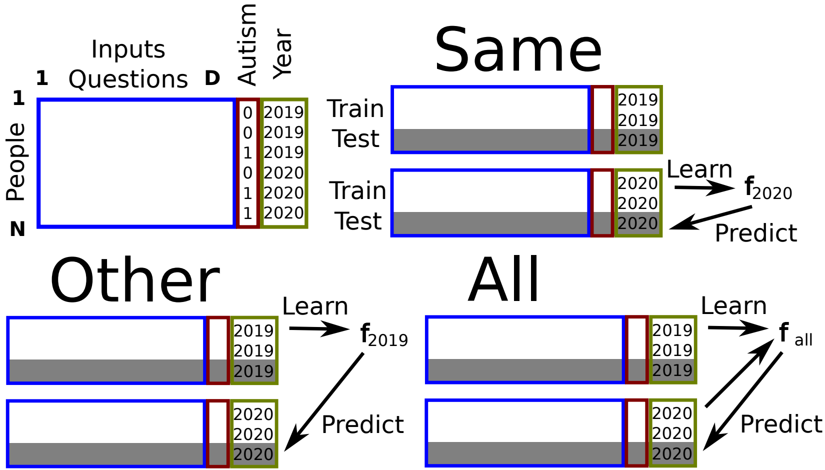
Proposed Same Other Cross-Validation

- One train/test split shown out of K folds.



Proposed Same Other Cross-Validation

- One train/test split shown out of K folds.



Proposed Same Other Cross-Validation

For a fixed test set from one group:
If train/test are similar/iid,

- **All** should be most accurate.
- **Same/Other** should be less accurate, because there is less data available (if other is larger than same, then other should be more accurate than same, etc).

If train/test are different (not iid),

- **Same** should be most accurate.
- **Other** should be substantially less accurate.
- **All** accuracy between same and other.

Introduction to machine learning

Proposed same vs. other cross-validation

Results on real data sets

Discussion and Conclusions

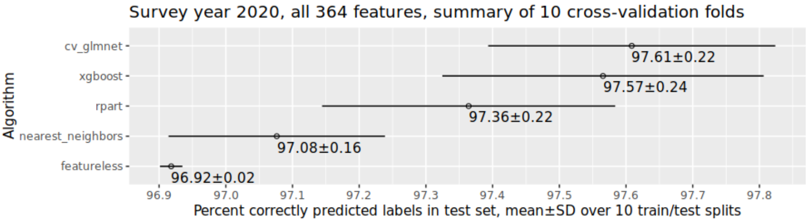
Learning algorithms we consider

We use R packages that implement the following learning algorithms, in the mlr3 R package framework:

- **cv_glmnet** L1-regularized linear model (feature selection). Friedman, *et al.* (2010).
- **xgboost** Extreme gradient boosting (non-linear). Chen and Guestrin (2016).
- **rpart** Recursive partitioning, decision tree (non-linear, feature selection). Therneau and Atkinson (2023).
- **nearest_neighbors** classic non-linear algorithm, as implemented in kknn R package. Schliep and Hechenbichler (2016).
- **featureless** un-informed baseline, ignores all inputs/features, and always predicts the most frequent label in train data (Autism=No in our case). Nomenclature from mlr3 R package, Lang, *et al.*, (2019).

Each learning algorithm has different properties (non-linear, feature selection, etc). For details see Hastie, *et al.* (2009) textbook.

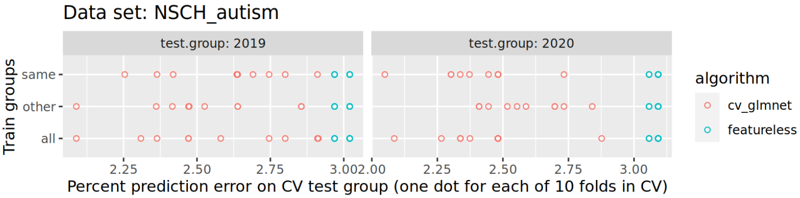
Summarize 10 folds with mean and standard deviation



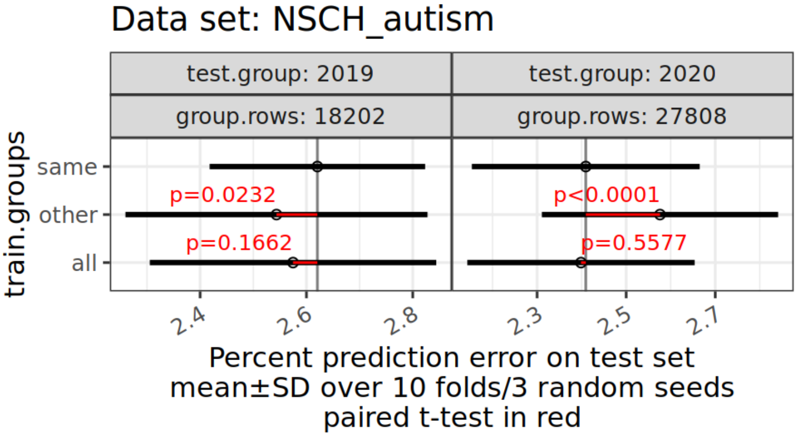
Learning algorithms we consider:

- cv_glmnet — L1-regularized linear model (feature selection).
- xgboost — Extreme gradient boosting (non-linear).
- rpart — Recursive partitioning, decision tree (non-linear, feature selection).
- nearest_neighbors — classic non-linear algorithm.
- featureless — un-informed baseline, ignores all inputs/features, and always predicts the most frequent label in train data (Autism=No in our case).

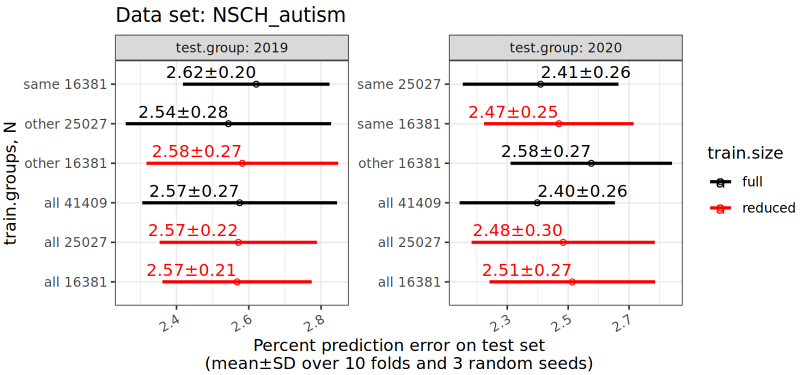
Same Other for Autism data



Same Other for Autism data



Same Other for Autism data



Proposed Same Other Cross-Validation

- 18,202 rows in 2019, whereas 27,808 in 2020.
- For predicting in 2019 (left), training on only 2019 (same) is slightly less accurate than training on only 2020 (other), and 2019+2020 (all). This suggests 2020 data are consistent with the pattern in 2019, which is too complex to learn from the limited 2019 data alone (there is a slight advantage to combining years when training).
- For predicting in 2020 (right), training on 2019 (other) is slightly less accurate than training on 2020 (same), and 2019+2020 (all). This again suggests that 2019/2020 data are consistent, but there are not enough data in 2019 alone.

Discussion and Conclusions

- Proposed Same Other Cross-Validation can be used to see if it is beneficial to learn using data from different groups (train on one group, test/predict on another).
- Free/open-source software available: mlr3resampling R package on `https://github.com/tdhock/mlr3resampling`.
- These slides are reproducible, using the code in `https://github.com/tdhock/cv-same-other-paper`
- Contact: toby.hocking@nau.edu, toby.hocking@r-project.org