Supervised Learning Methodology

Christoph Kern

Mannheim Machine Learning Modules

c.kern@uni-mannheim.de

March 21 and 22, 2018

UNIVERSITY OF MANNHEIM

Machine Learning task

↓

Get and integrate data

↓

Explore and pre-process data

↓

Features

↓

Training and tuning

↓

Evaluation

↓

Deploy, maintain and update

Outline

1. Machine Learning basics
   - In-sample prediction error
   - Validation set, test set, CV
   - Grids and random search
   - Performance measures
2. Software Resources
3. References

Machine Learning basics

Unsupervised Learning

- Finding patterns in data using a set of input variables X

Supervised Learning

- Predicting an output variable Y based on a set of input variables X
  1. Learn the relationship between input and output using **training data** (with X and Y)
  $$Y = f(X) + \varepsilon$$
  2. Predict the output based on the prediction model (of step 1) for **new test data** ($\sim$only X available)
- continuous Y: regression, categorical Y: classification
- Focus on **prediction** ($\neq$ causation)

Machine Learning basics

Unsupervised Learning

- Finding patterns in data using a set of input variables X

Supervised Learning

- Predicting an output variable Y based on a set of input variables X
  1. Learn the relationship between input and output using **training data** (with X and Y)

$$Y = f(X) + \varepsilon$$

  2. Predict the output based on the prediction model (of step 1) for **new test data** ($\sim$only X available)

- continuous Y: regression, categorical Y: classification
- Focus on **prediction** ($\neq$ causation)

Machine Learning basics

Unsupervised Learning

- Finding patterns in data using a set of input variables X

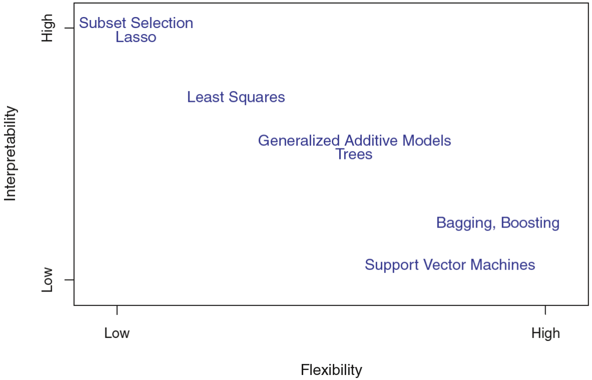
Supervised Learning

- Predicting an output variable Y based on a set of input variables X
  1. Learn the relationship between input and output using **training data** (with X and Y)
  $$Y = f(X) + \varepsilon$$
  2. Predict the output based on the prediction model (of step 1) for **new test data** ($\sim$only X available)
- continuous Y: regression, categorical Y: classification
- Focus on **prediction** ($\neq$ causation)

Table: Estimating $f(X)$

Regression methods	(tree-based) ML methods
parametric	non-parametric
linearity, additivity	flexible functional form
prior model specification	“built-in” feature selection
theory-driven	data-driven
$\rightarrow$ Inference	$\rightarrow$ Prediction

Figure: Flexibility-Interpretability Trade-Off



James et al. (2013)

In-sample prediction error

Estimating the test error with training data

- Setup: Add training optimism $\hat{\omega}$ to training error

$$\widehat{\mathrm{Err}}_{in} = \overline{\mathrm{err}} + \hat{\omega}$$

- Corrected fit measure for OLS regression

$$C_p = \overline{\mathrm{err}} + 2\frac{d}{n}\hat{\sigma}_{\varepsilon}^2$$

- Corrected fit measures for ML-based methods

$$AIC = -\frac{2}{n}LL + 2\frac{d}{n}$$

$$BIC = -2LL + \log(n)d$$

Validation set, test set, CV

Validation set approach

- Training set & validation set
  1. Fit model using one part of training data
  2. Compute test error for the excluded section

$\rightarrow$ Model assessment

- Training set, validation set & test set
  1. Fit models using training part of training data
  2. Choose best model using validation set
  3. Evaluate final model using test set

$\rightarrow$ Model tuning & assessment

Cross-Validation

- LOOCV (Leave-One-Out Cross-Validation)
    1. Fit model on training data while excluding one case
    2. Compute test error for the excluded case
    3. Repeat step 1 & 2 n times
- k-Fold Cross-Validation
    1. Fit model on training data while excluding one group
    2. Compute test error for the excluded group
    3. Repeat step 1 & 2 k times (e.g. $k = 5$, $k = 10$)
- Outlook: nested CV, repeated CV, ...

$$CV(\hat{f}) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, \hat{f}^{-\kappa(i)}(x_i))$$

Standard Errors for CV

$$\frac{1}{\sqrt{K}}\mathrm{sd}\{CV_1(\hat{f}^{-(1)}), \ldots, CV_K(\hat{f}^{-(K)})\}$$

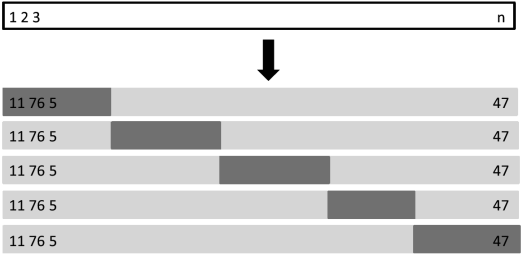
Model selection using k-Fold Cross-Validation

- Choose model with smallest cross-validated error
- Choose smallest model within one standard error of the smallest cross-validated error (1-SE Rule)

More on data splitting

- Simple random splits
    - General approach for "unstructured" data
    - Typically 75% or 80% go into training set
- Stratified splits
    - For classification problems with class imbalance
    - Sampling within each class of Y to preserve class distribution
- Splitting by groups
    - For (temporal) structured data
    - Use specific groups (temporal holdouts) for validation

Figure: 5-Fold Cross-Validation with training set and validation set (example)



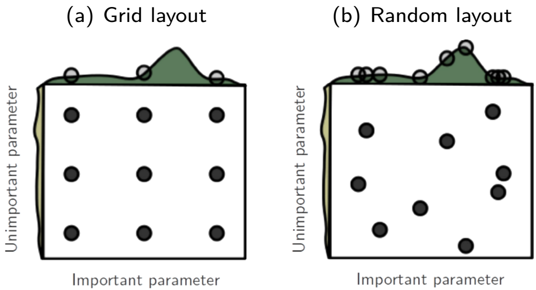
James et al. (2013)

Grids and random search

Tuning many hyperparameters

- (Exhaustive) Grid search
  - Expands a grid over all combinations of considered try-out values
  - Can become inefficient with many tuning parameters
- Random search (Bergstra & Benglio 2012)
  - Considers only a random selection of tuning parameter combinations
  - Benefit depends on method and implementation
- Adaptive search (Kuhn 2014)
  - Guided search by considering performance within the search process
  - Adaptive removal of unpromising parameter settings

Figure: Grid and random search with two tuning parameters



Bergstra & Benglio (2012)

Performance measures for regression

r^2 score:

$$r^2 = \mathrm{corr}(y_i, \hat{f}(x_i))^2$$

Mean of squared errors (MSE):

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

Root mean squared error (RMSE):

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2}$$

Mean of absolute errors (MAE):

$$\frac{1}{n}\sum_{i=1}^{n}|(y_i - \hat{f}(x_i))|$$

Median of absolute errors (MEDAE):

$$\text{median}(|y_1 - \hat{f}(x_1)|, ..., |y_n - \hat{f}(x_n)|)$$

Median of squared errors (MEDSE):

$$\text{median}((y_1 - \hat{f}(x_1))^2, ..., (y_n - \hat{f}(x_n))^2)$$

Performance measures for classification

Probabilities, thresholds and prediction for classification

$$y_i = \begin{cases} 1 & if \quad p_i > c \\ 0 & if \quad p_i \leq c \end{cases}$$

Table: Confusion matrix

		Prediction 0	Prediction 1	
Reference	0	True Negatives (TN)	False Positives (FP)	N'
	1	False Negatives (FN)	True Positives (TP)	P'
		N	P	

Performance metrics for classification

- Global performance
  - Accuracy: $\frac{TP+TN}{TP+FP+TN+FN}$
  - Misclassification rate: $\frac{FP+FN}{TP+FP+TN+FN}$
  - No Information rate
- Row / column performance
  - Sensitivity (Recall): $\frac{TP}{TP+FN}$
  - Specificity: $\frac{TN}{TN+FP}$
  - Positive predictive value (Precision): $\frac{TP}{TP+FP}$
  - Negative predictive value: $\frac{TN}{TN+FN}$
  - False positive rate: $\frac{FP}{FP+TN}$
  - False negative rate: $\frac{FN}{FN+TP}$

Combined measures

- Balanced Accuracy

$$(Sensitivity + Specificity)/2$$

- $F1$

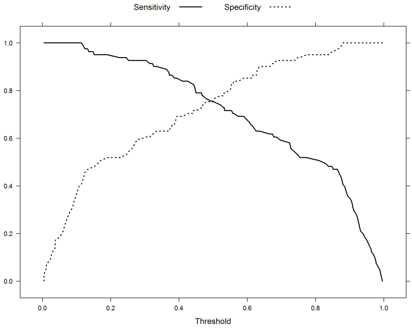
$$2 \times \frac{Precision \times Recall}{Precision + Recall}$$

- Cohen's κ
  - Compares observed (p_0) and random (p_e) accuracy
  - $p_e = \frac{(N' \times N)+(P' \times P)}{(TP+FP+TN+FN)^2}$

$$1 - \frac{1 - p_0}{1 - p_e}$$

Figure: Varying the classification threshold I

(a) Sensitivity and specificity



(b) Precision and recall

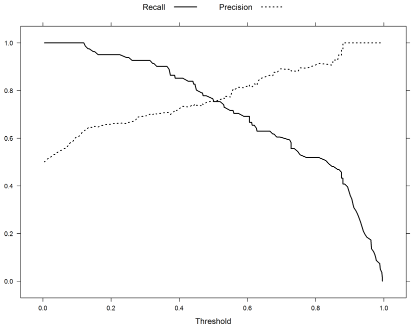
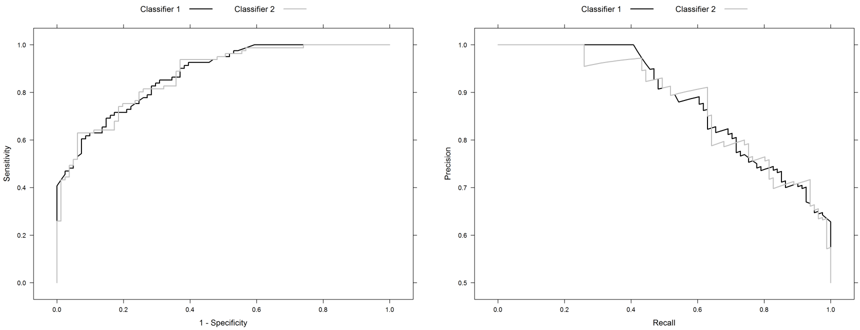


Figure: Varying the classification threshold II

(a) ROC curves (b) PR curves



→ AUC-ROC: Area under the receiver operating characteristic curve

→ AUC-PR: Area under the precision–recall curve

Software Resources

Resources for R

- Classification and Regression Training: `caret`
  - `https://topepo.github.io/caret/`
- Machine Learning in R: `mlr`
  - `https://mlr-org.github.io/mlr-tutorial/devel/html/`
- Collection of performance metrics: `MLmetrics`, `verification`
- ROC and PR curves: e.g. `PRROC`

References

Bergstra, J. and Bengio, Y. (2012). Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research, 13*, 281–305

Ghani, R. and Schierholz, M. (2017). Machine Learning. In: Foster, I., Ghani, R., Jarmin, R. S., Kreuter, F., Lane, J. (Eds.). *Big Data and Social Science: A Practical Guide to Methods and Tools*. Boca Raton, FL: CRC Press Taylor & Francis Group.

Hastie, T., Tibshirani, R., Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York, NY: Springer.

James, G., Witten, D., Hastie, T., Tibshirani, R. (2013). *An Introduction to Statistical Learning*. New York, NY: Springer.

Kuhn, M. (2014). *Futility Analysis in the Cross-Validation of Machine Learning Models*. `https://arxiv.org/abs/1405.6974`.

Kuhn, M. and Johnson, K. (2013). *Applied Predictive Modeling*. New York, NY: Springer.

Varian, H. R. (2014). Big Data: New Tricks for Econometrics. *Journal of Economic Perspectives, 28*(2), 3–28.