ML Basics

Machine Learning for Social Science

Recall: ML process

Machine Learning task

↓

Get and integrate data

↓

Explore and pre-process data

↓

Features

↓

Training and tuning

↓

Evaluation

↓

Deploy, maintain and update

ML basics

Supervised Learning Goal: For a given outcome (label), make optimal predictions (according to some performance metric) in a **new data set** using existing known predictors (features).

That is, we are optimizing for **predictive ability**!

Why do we split our data? Since our goal is to be able to make optimal decisions on a new data set, we evaluate our candidate models on "new data" by setting aside a **validation set** that **we do not include in the model**. We also include a **test set that we do not touch until the very end** so that we have an unbiased estimate of our final model performance.

ML basics

Supervised Learning Goal: For a given outcome (label), make optimal predictions (according to some performance metric) in a **new data set** using existing known predictors (features).

That is, we are optimizing for **predictive ability**!

Why do we split our data? Since our goal is to be able to make optimal decisions on a new data set, we evaluate our candidate models on "new data" by setting aside a **validation set** that **we do not include in the model**. We also include a **test set that we do not touch until the very end** so that we have an unbiased estimate of our final model performance.

Training and test error

Training error

$$\overline{\text{err}} = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}(x_i))$$

- Prediction error based on **training data**
- with e.g. squared error loss L

Test error

$$\text{Err}_{\mathcal{T}} = \text{E}(L(Y, \hat{f}(X))|\mathcal{T})$$

- Prediction error using **test data** (given training data $\mathcal{T}$)

Training and test error

Training error

$$\overline{\text{err}} = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}(x_i))$$

- Prediction error based on **training data**
- with e.g. squared error loss L

Test error

$$\text{Err}_{\mathcal{T}} = \text{E}(L(Y, \hat{f}(X))|\mathcal{T})$$

- Prediction error using **test data** (given training data $\mathcal{T}$)

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

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

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

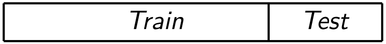
Validation set, test set, CV

Training set & test set

- Estimate prediction error on new data
  1. Fit model using one part of training data
  2. Compute test error for the excluded section

→ Model assessment

Figure: 80/20 train-test split

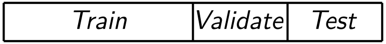


Training set, validation set & test set

- Compare models and estimate prediction error
  1. Fit models using training part of training data
  2. Choose best model using validation set
  3. Evaluate final model using test set

→ Model tuning & assessment

Figure: 50/25/25 Train-validation-test split



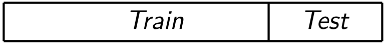
Validation set, test set, CV

Training set & test set

- Estimate prediction error on new data
  1. Fit model using one part of training data
  2. Compute test error for the excluded section

→ Model assessment

Figure: 80/20 train-test split

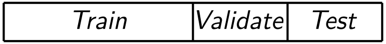


Training set, validation set & test set

- Compare models and estimate prediction error
  1. Fit models using training part of training data
  2. Choose best model using validation set
  3. Evaluate final model using test set

→ Model tuning & assessment

Figure: 50/25/25 Train-validation-test split



Leave test data untouched until the end of analysis!

Validation set, test set, CV

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, ...

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

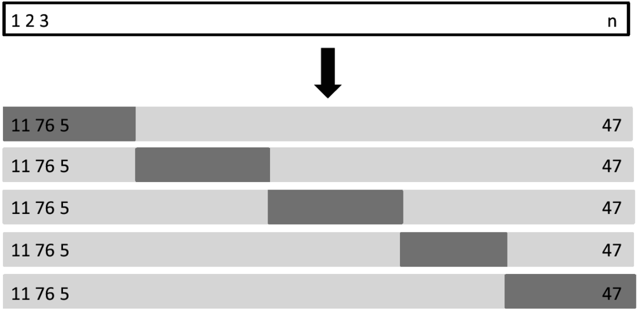
Validation set, test set, CV

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

Validation set, test set, CV

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



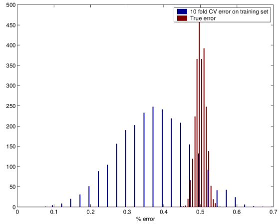
James et al. (2013)

Tuning and Cross-Validation

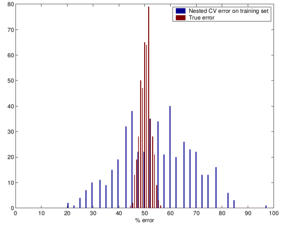
- Repeated Cross-Validation
  1. Run e.g. 10-fold CV five times
  2. Average performance scores over repetitions
  3. Different splits into folds increases robustness
- Nested Cross-Validation
  1. Split data into outer and inner folds
  2. Inner folds: Run CV within inner training fold(s) for tuning
  3. Outer folds: Evaluate best model on the outer test fold(s)
  4. Separates model selection and model assessment

Figure: Bias in CV error (Varma and Simon 2006)

(a) 10-fold CV



(b) Nested CV



Performance measures for regression

Performance measures for regression

r^2 score:

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

Residual Sum of Squares (RSS):

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

Mean of squared errors (MSE):

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

Root mean squared error (RMSE):

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

Performance measures for regression

Mean of absolute errors (MAE):

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

Median of absolute errors (MEDAE):

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

Median of squared errors (MEDSE):

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

Performance measures for classification

Probabilities, thresholds and prediction for classification

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

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 measures for classification

Confusion matrix metrics

- 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: $\frac{FN}{FN+TP}$

Table: Confusion matrix

		Prediction		
		0	1	
Reference	0	TN	FP	N'
	1	FN	TP	P'
		N	P	

Performance measures for classification

Combined measures

- Balanced Accuracy

$$(Sensitivity + Specificity)/2$$

- $F1$

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

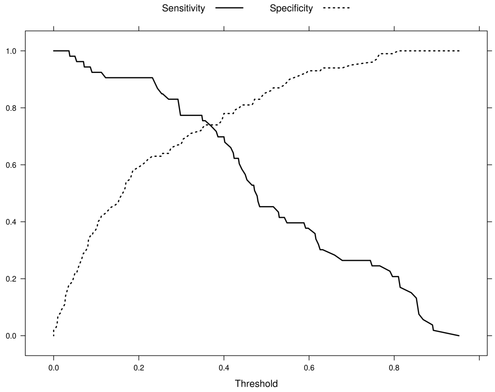
- Cohen's κ
  - Compares observed (p_0) and random (p_e) accuracy
  - $p_e = \frac{(N' \times N) + (P' \times P)}{(TP+FP+TN+FN)^2}$

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

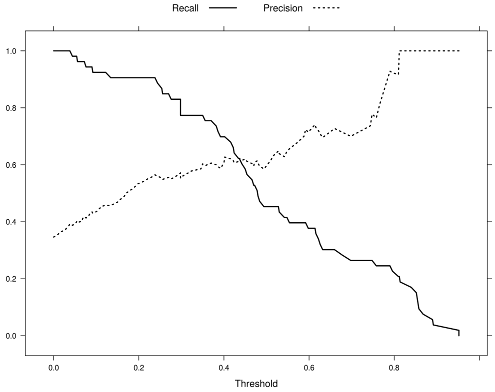
Performance measures for classification

Figure: Varying the classification threshold I

(a) Sensitivity and specificity



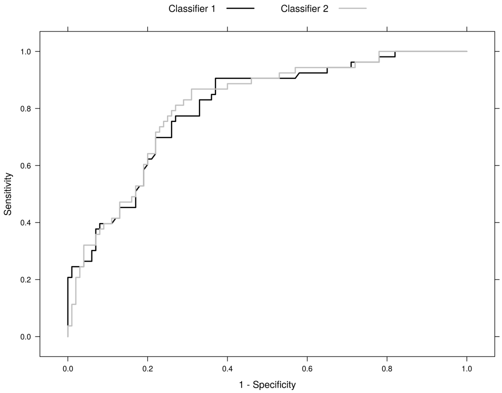
(b) Precision and recall



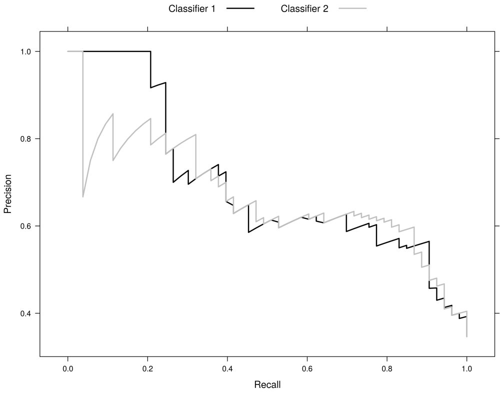
Performance measures for classification

Figure: Varying the classification threshold II

(a) ROC curves



(b) PR curves



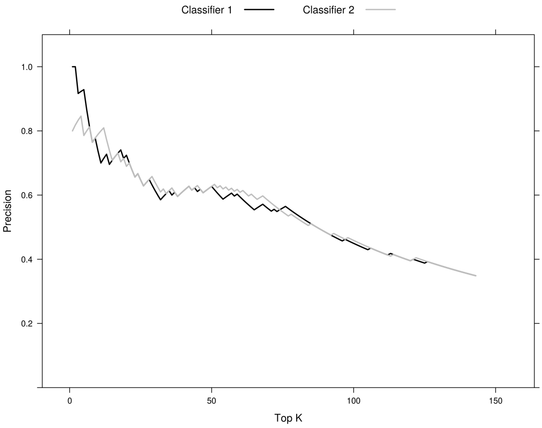
→ AUC-ROC: Area under the receiver operating characteristic curve
→ AUC-PR: Area under the precision–recall curve

Performance measures for classification

How many true positives are among the high risk observations?

1. Rank observations by risk scores
2. Classify top K % as positive/ relevant
3. Compute precision

Figure: Precision at top K



Software Resources

Resources for R

- Overview
  - https://cran.r-project.org/web/views/MachineLearning.html
- caret
  - http://topepo.github.io/caret/index.html
- mlr
  - https://mlr-org.github.io/mlr-tutorial/devel/html/
- H2O
  - http://docs.h2o.ai/

References

Buskirk, T. D., Kirchner, A., Eck, A., Signorino, C. S. (2018). An Introduction to Machine Learning Methods for Survey Researchers. *Survey Practice 11*(1).

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM 55*(10), 78–87.

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.