Supervised Learning I – Intro

Big Data Analysis I

Christoph Kern[1] Frauke Kreuter Marcel Neunhoeffer Sebastian Sternberg

March 25, 2019

[1] c.kern@uni-mannheim.de

Table of contents

Machine Learning basics

What is Machine Learning?

> *A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.*

– Tom Mitchell (1997)

Introduction

Machine Learning

- “Machine Learning is the field of scientific study that concentrates on induction algorithms and on other algorithms that can be said to “learn”.” (Kohavi & Provost 1998)
    - Algorithms based on statistical criteria which focus on making predictions based on a data-driven learning process
- Combines Computer Science and Statistics

Statistical Learning

- Machine Learning from a “statistical perspective”

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

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**

ML 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**

ML 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**

ML 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**

ML basics

Supervised Learning: Find function $f(x)$ that makes optimal predictions in a new data set

Prerequisites:

- Representation: What is the *hypothesis space*, the family of functions to search over?
  - Describes possible relationships between X and Y
  - Examples: $f(x) = x'\beta$ is linear, or f is a tree.
- Evaluation: What is the criterion to choose between different functions?
  - Measures predictive performance
  - Examples: Mean Squared Error, Logistic Loss
- Computation: How is f actually calculated?
  - Speed and memory space may be limiting factors

ML basics

Supervised Learning: Find function $f(x)$ that makes optimal predictions in a **new data set**

Prerequisites:

- **Representation**: What is the *hypothesis space*, the family of functions to search over?
  - Describes possible relationships between X and Y
  - Examples: $f(x) = x'\beta$ is linear, or f is a tree.
- **Evaluation**: What is the criterion to choose between different functions?
  - Measures predictive performance
  - Examples: Mean Squared Error, Logistic Loss
- **Computation**: How is f actually calculated?
  - Speed and memory space may be limiting factors

ML basics

Supervised Learning: Find function $f(x)$ that makes optimal predictions in a **new data set**

Prerequisites:

- **Representation**: What is the *hypothesis space*, the family of functions to search over?
  - Describes possible relationships between X and Y
  - Examples: $f(x) = x'\beta$ is linear, or f is a tree.
- **Evaluation**: What is the criterion to choose between different functions?
  - Measures predictive performance
  - Examples: Mean Squared Error, Logistic Loss
- **Computation**: How is f actually calculated?
  - Speed and memory space may be limiting factors

ML basics

Supervised Learning: Find function $f(x)$ that makes optimal predictions in a **new data set**

Prerequisites:

- **Representation**: What is the *hypothesis space*, the family of functions to search over?
  - Describes possible relationships between X and Y
  - Examples: $f(x) = x'\beta$ is linear, or f is a tree.
- **Evaluation**: What is the criterion to choose between different functions?
  - Measures predictive performance
  - Examples: Mean Squared Error, Logistic Loss
- **Computation**: How is f actually calculated?
  - Speed and memory space may be limiting factors

ML basics

Table 1: 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

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 $\mathcal{T}$)

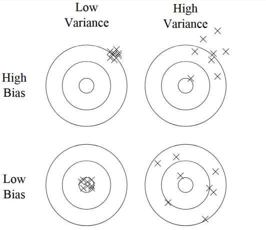
Training and test error

Expected test error decomposition

$$\mathrm{Err}(x_0) = \mathrm{Bias}^2(\hat{f}(x_0)) + \mathrm{Var}(\hat{f}(x_0)) + \mathrm{Var}(\varepsilon)$$

- Minimizing the (expected) test error requires
  - Low bias ($[\mathrm{E}\hat{f}(x_0) - f(x_0)]^2$) and
  - Low variance ($\mathrm{E}[\hat{f}(x_0) - \mathrm{E}\hat{f}(x_0)]^2$)

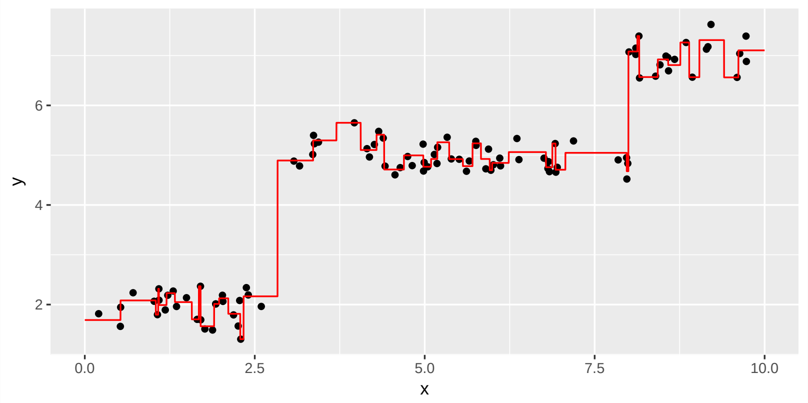
Figure 1: Bias and variance illustration



Domingos (2012)

Bias-Variance Trade-Off

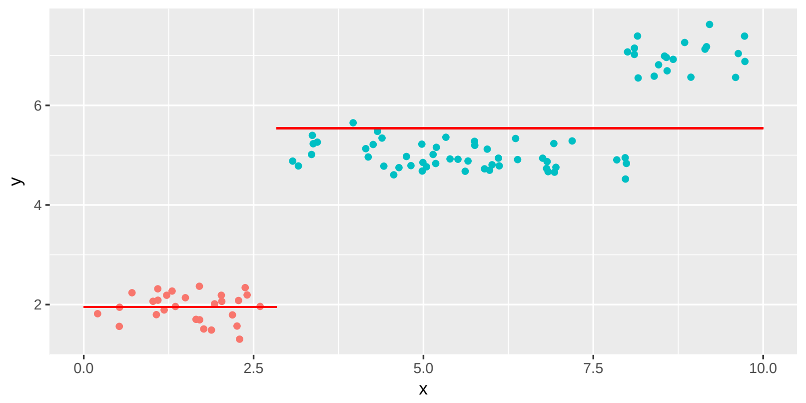
Figure 2: High Variance in Trees



- High Variance = Different data would lead to a different function
- Overfitting = Poor generalization to new data

Bias-Variance Trade-Off

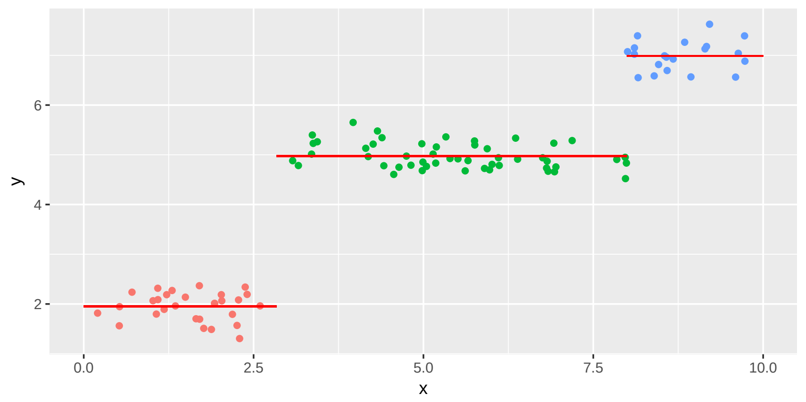
Figure 3: High Bias in Trees



- High Bias = Blue points are poorly predicted
- Underfitting = Function should adapt better to the data

Bias-Variance Trade-Off

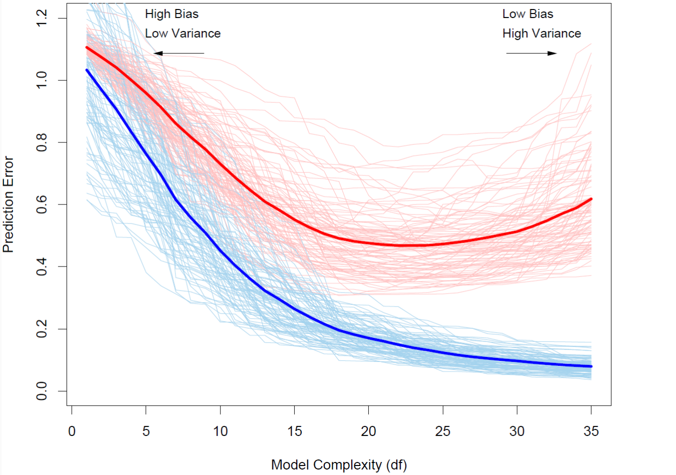
Figure 4: Optimal Solution



- Goal: Find optimal compromise between bias and variance

Bias-Variance Trade-Off

Figure 5: Training error and test error by model complexity



Hastie et al. (2009)

Goal of prediction (again): Minimal **test error**

$$\arg\min_{f \in \mathcal{F}} \mathbb{E}(L(f(x), y))$$

but we cannot simply minimize its empirical analogue in **training data**

$$\arg\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^{N} L(f(x_i), y_i)$$

because this would overfit if the capacity of f is high enough.

Bias-Variance Trade-Off

Solution: Solve (as before)

$$\arg\min_{f \in \mathcal{F}_K} \frac{1}{N} \sum_{i=1}^{N} L(f(x_i), y_i)$$

but f must come from a restricted hypothesis space (limited capacity)

- Tree with at most K leaves
- Regression with $\sum |\beta_j| < K$
- General form: $\text{Penalty}(f) < K$

This is **regularization** – in general form:

$$\arg\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^{N} L(f(x_i), y_i) + \lambda \cdot \text{Penalty}(f)$$

Quiz

If we have a high bias problem (underfitting), what can be done?

- Add more predictors (= collect more variables or transform existing ones)?
- Allow higher function capacity (= reduce regularization parameter)?
- Use more flexible algorithms (e.g., a tree instead of linear regression)?

If we have a high variance problem (overfitting), what can be done?

- Add more predictors (= collect more variables or transform existing ones)?
- Allow higher function capacity (= reduce regularization parameter)?
- Use more flexible algorithms (e.g., a tree instead of linear regression)?
- Collect more training data?

Validation set, test set, CV

In-sample prediction error

Estimating the test error with training data

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

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

- Corrected fit measure for OLS regression

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

- 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

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, ...

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

Cross-Validation

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)

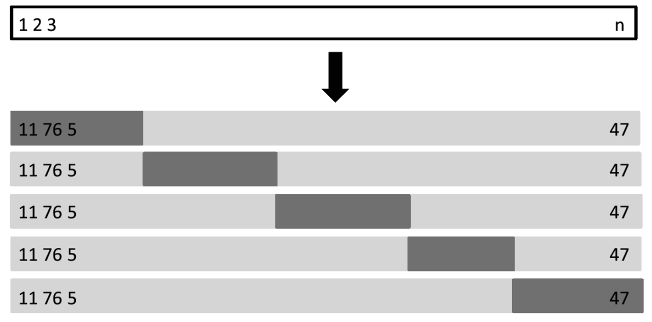
Cross-Validation

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

Cross-Validation

Figure 6: 5-Fold Cross-Validation with training set and validation set (example)



James et al. (2013)

Performance measures

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):

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

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):

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

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 2: 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 3: Confusion matrix

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

Performance measures for classification

Combined measures

- Balanced Accuracy

$$(Sensitivity + Specificity)/2$$

- $F1$

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

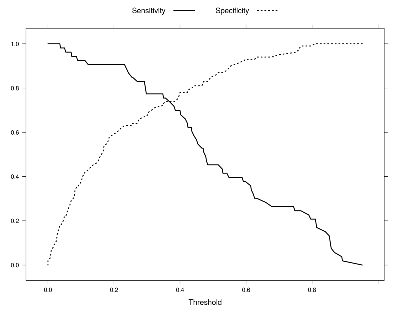
- Cohen's κ
    - Compares observed (p_0) and random (p_e) accuracy
    - $p_e = \frac{(N' \times N) + (P' \times P)}{(TP+FP+TN+FN)^2}$

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

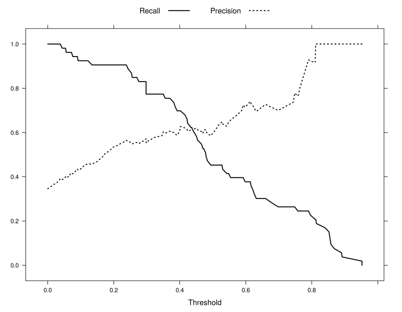
Performance measures for classification

Figure 7: Varying the classification threshold I

(a) Sensitivity and specificity



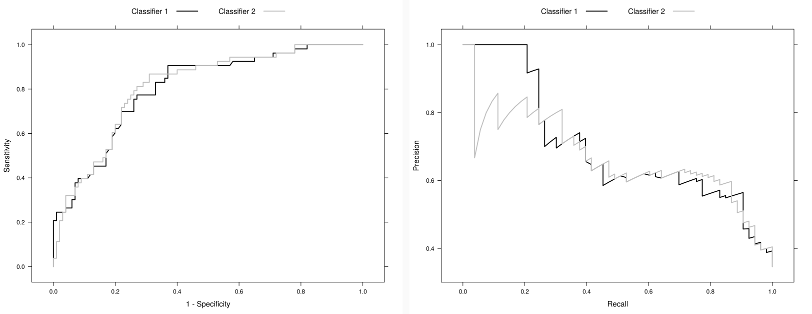
(b) Precision and recall



Performance measures for classification

Figure 8: Varying the classification threshold II

(a) ROC curves (b) PR curves



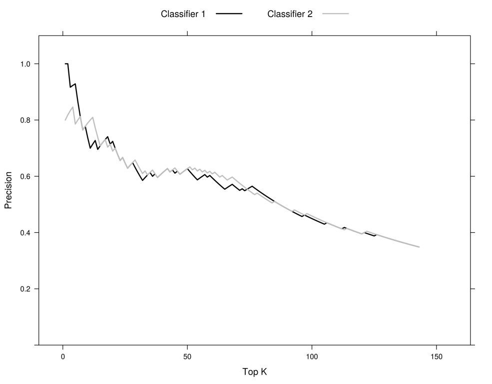
→ AUC-ROC: Area under the receiver operating characteristic curve
→ AUC-PR: Area under the precision–recall curve

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 9: 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/`

References

References

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.

Kohavi, R., Provost, F. (1998). Glossary of Terms. *Machine Learning 30*(2), 271–274.

Mitchell, T. M. (1997). *Machine Learning*. Maidenhead: McGraw-Hill.