Business Analytics

Model Selection and Learning Theory

Prof. Bichler

Decision Sciences & Systems

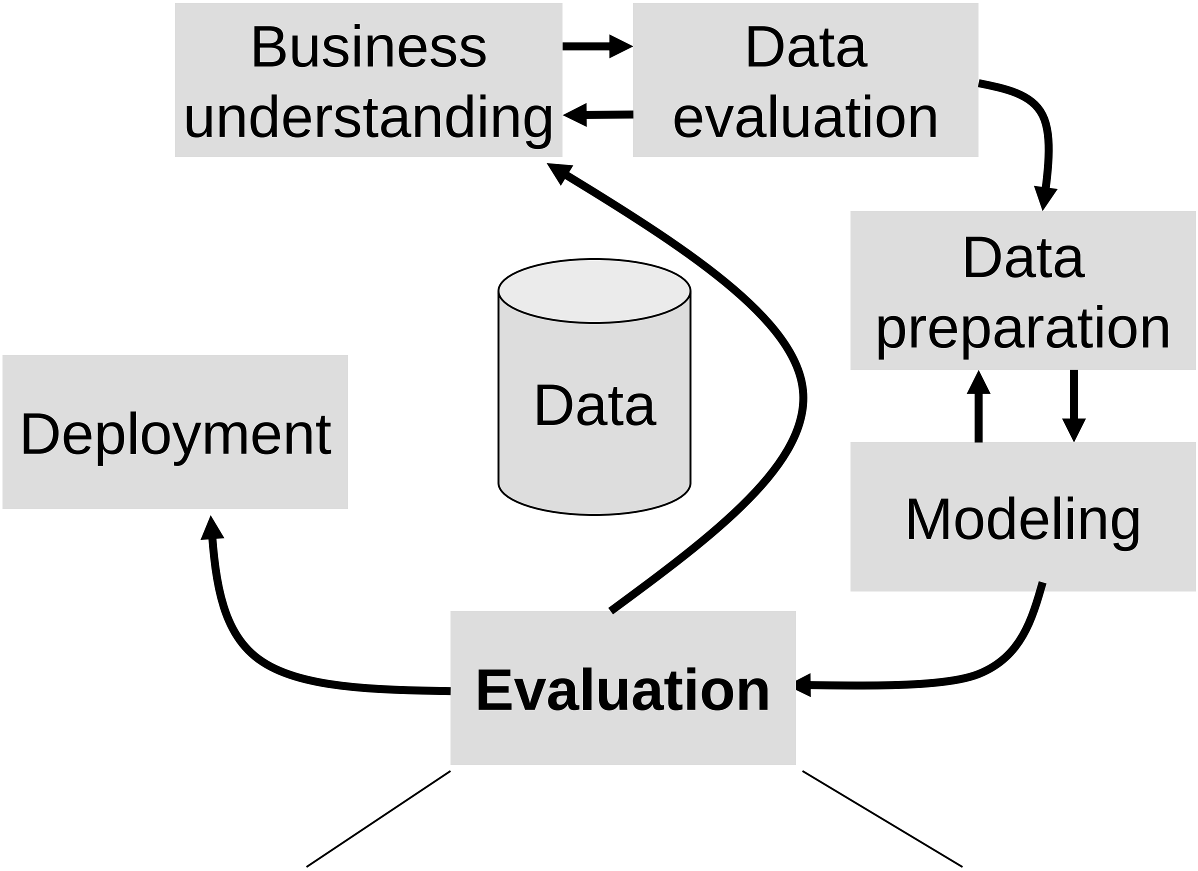
Department of Informatics

Technische Universität München

Course Content

- Introduction
- Regression Analysis
- Regression Diagnostics
- Logistic and Poisson Regression
- Naive Bayes and Bayesian Networks
- Decision Tree Classifiers
- Data Preparation and Causal Inference
- **Model Selection and Learning Theory**
- Ensemble Methods and Clustering
- High-Dimensional Problems
- Association Rules and Recommenders
- Neural Networks

The CRISP Data Mining Process



Business understanding

Data evaluation

Data preparation

Modeling

Evaluation

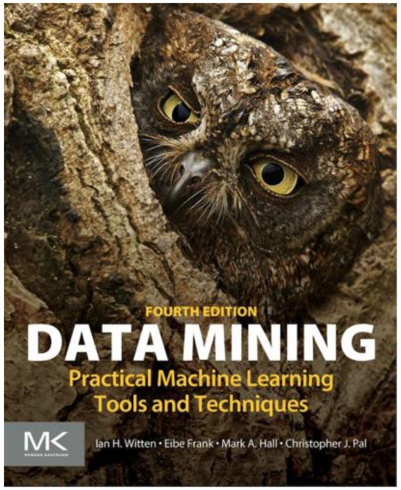
Deployment

Data

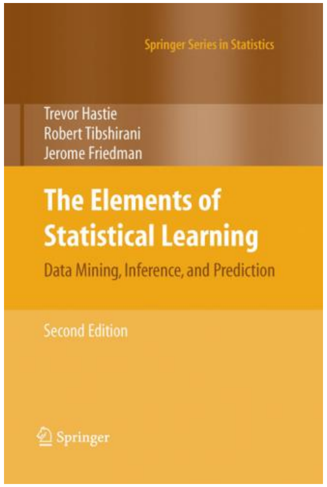
see also Witten&Frank: Chapter 5

Recommended Literature

- **Data Mining: Practical Machine Learning Tools and Techniques**
  - Ian H. Witten,Eibe Frank,Mark A. Hall
  - http://www.cs.waikato.ac.nz/ml/weka/book.html
  - Today's class: Chapter 5


FOURTH EDITION
DATA MINING
Practical Machine Learning Tools and Techniques
MK
Ian H. Witten • Eibe Frank • Mark A. Hall • Christopher J. Pal

- **The Elements of Statistical Learning**
  (Trevor Hastie, Robert Tibshirani, Jerome Friedman)
  - https://web.stanford.edu/~hastie/ElemStatLearn/
  - Section 2.9, 7: Model Assessment and Selection


Springer Series in Statistics
Trevor Hastie
Robert Tibshirani
Jerome Friedman
The Elements of Statistical Learning
Data Mining, Inference, and Prediction
Second Edition
Springer

Outline for Today

- **Bias-Variance Tradeoff**
- Resampling Methods
- Gain and ROC Curves
- Comparison Studies
- Algorithmic Information Theory
- Computational Learning Theory

> *“All models are wrong; some are useful.”*
>
> —George E. P. Box

Supervised Learning

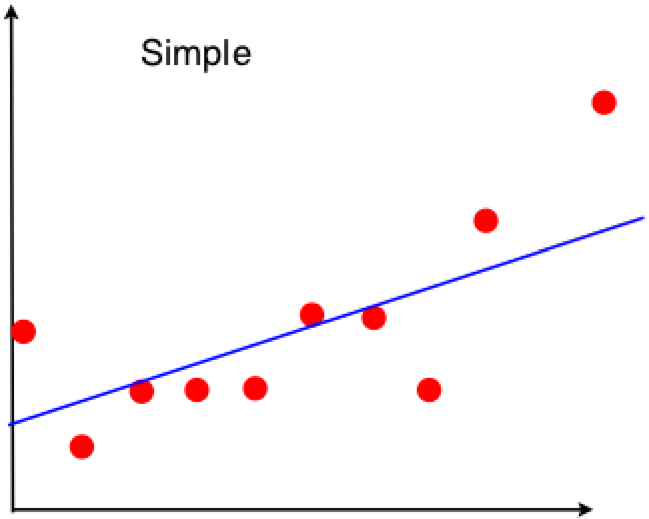
$$\hat{y} = f(\boldsymbol{x})$$

Supervised learning is inferring a function from labeled training data

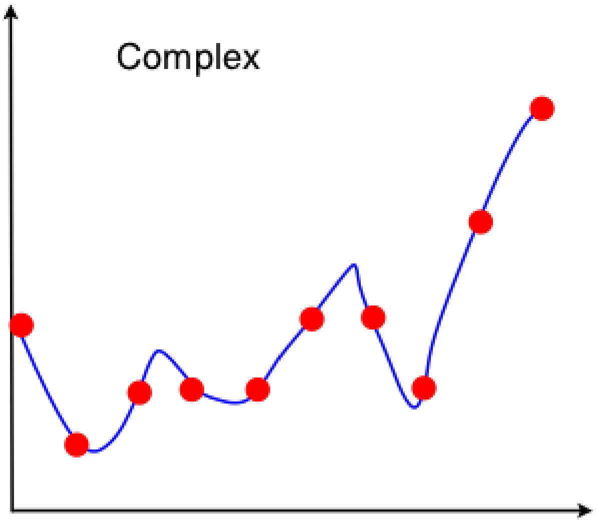
Training: given a *training set* of labeled examples $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ estimate the prediction function f by minimizing the prediction error on the training set

Testing: apply f to a never before seen *test example* $\boldsymbol{x}$ and output the predicted value $\hat{y} = f(\boldsymbol{x})$

Bias-Variance Tradeoff



Models with too few parameters are inaccurate because of a large bias (not enough flexibility).



Models with too many parameters are inaccurate because of a large variance (too much sensitivity to the sample).

Generalization Errors

Components of generalization error

- **Bias** is error from erroneous assumptions in the learning algorithm. Error might be due to inaccurate assumptions/simplifications made by the model.
- **Variance** is error from sensitivity to small fluctuations in the training set. High variance causes overfitting.

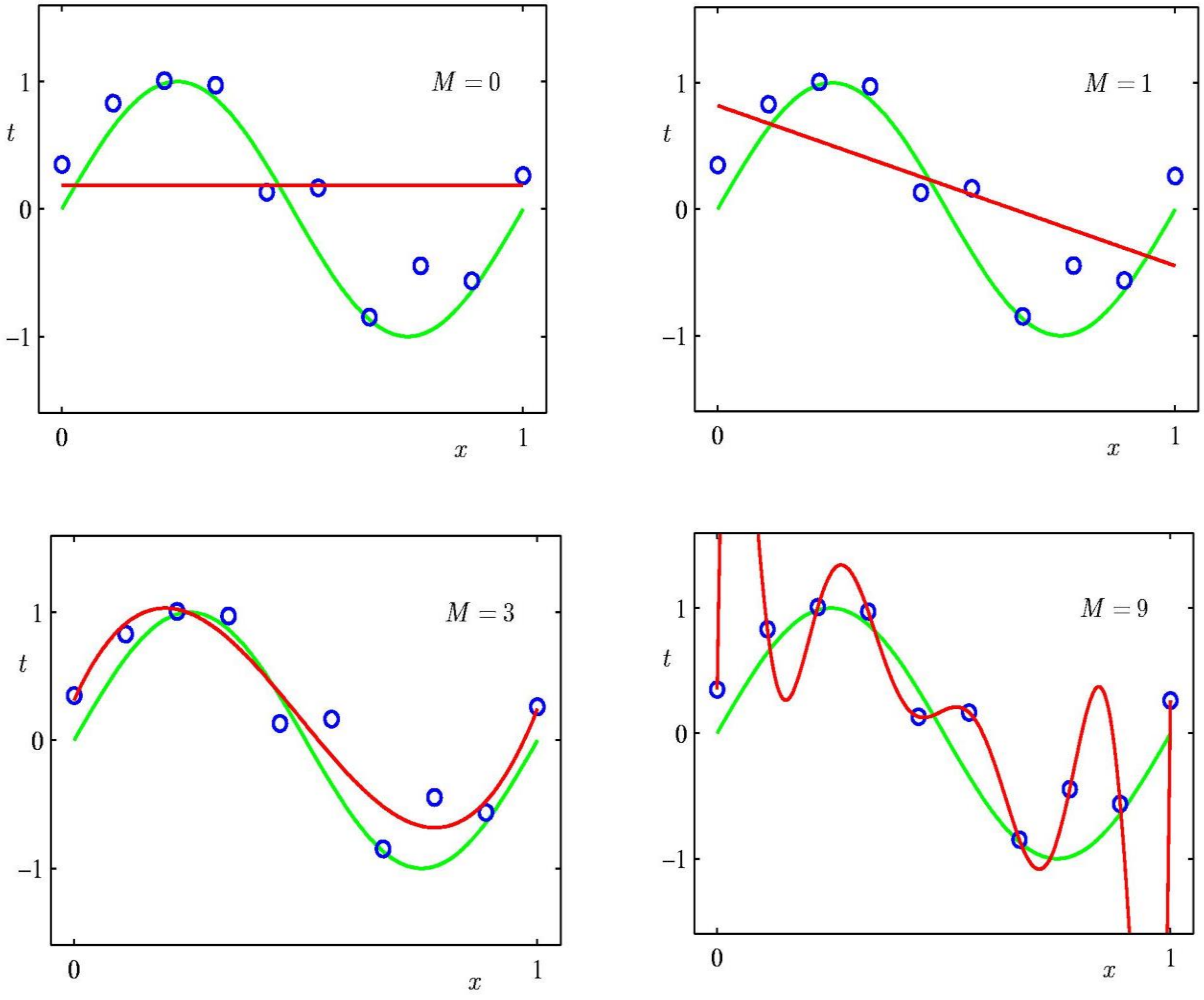
Underfitting: model is too “simple” to represent all relevant characteristics

- High bias and low variance
- High training error and high test error

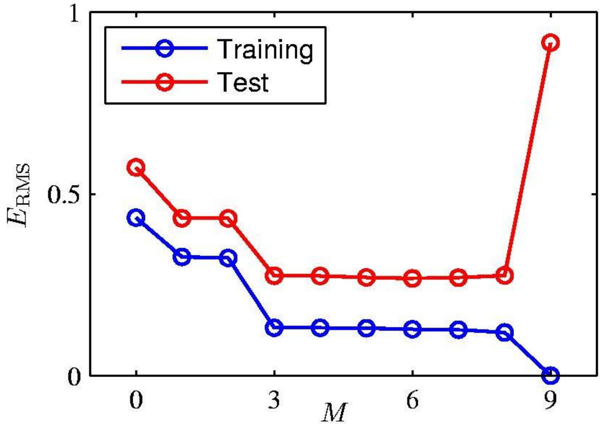
Overfitting: model is too “complex” and fits irrelevant characteristics/noise

- Low bias and high variance
- Low training error and high test error

More Complex Models, Less Bias, More Variance

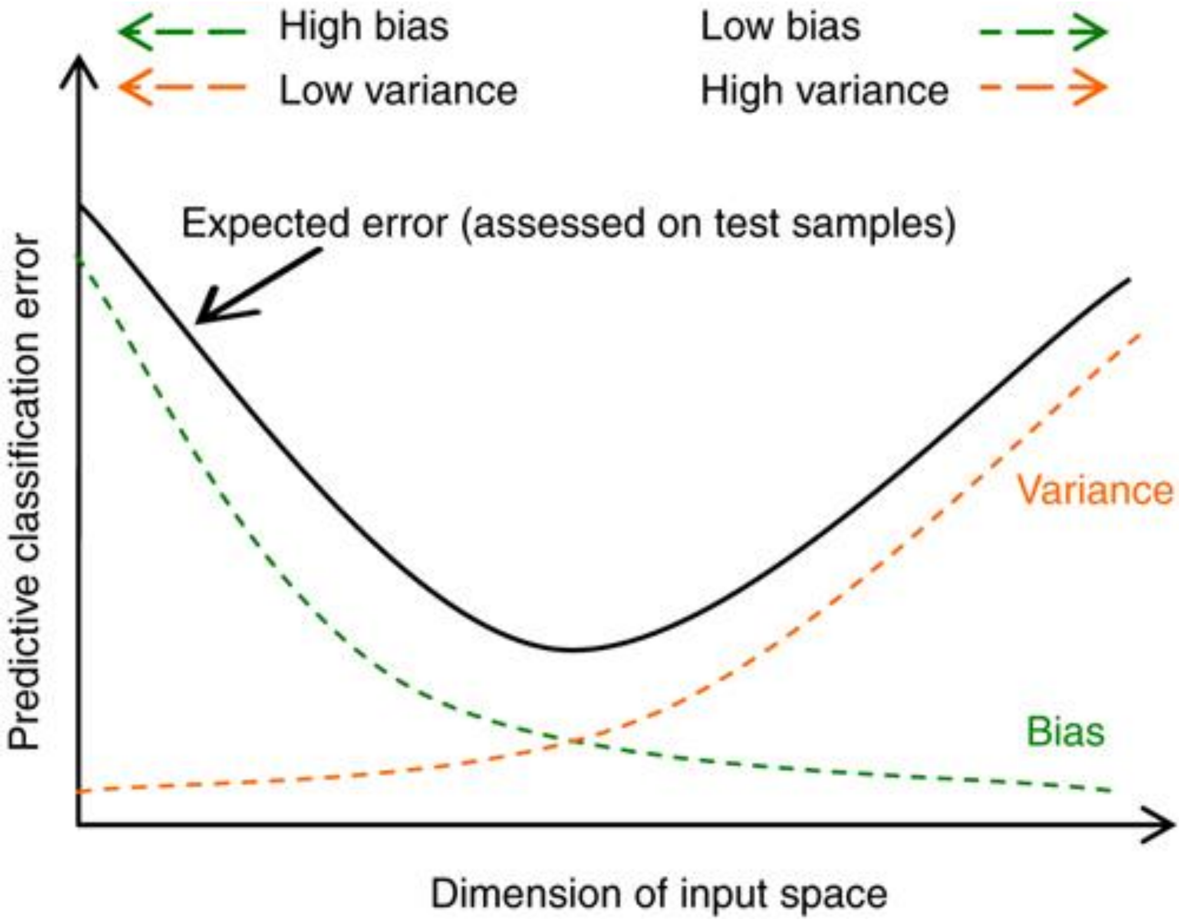


Fitting a polynomial



Root mean squared error

Bias-Variance Tradeoff



Which Model Should be Selected?

The bias-variance tradeoff provides a conceptual framework for determining a good model (but it is not directly useful)

We need a *practical method* for optimizing bias-variance tradeoff
Practical aim is to pick a model that minimizes a criterion:

f(fitting error from given data) + g(model complexity)

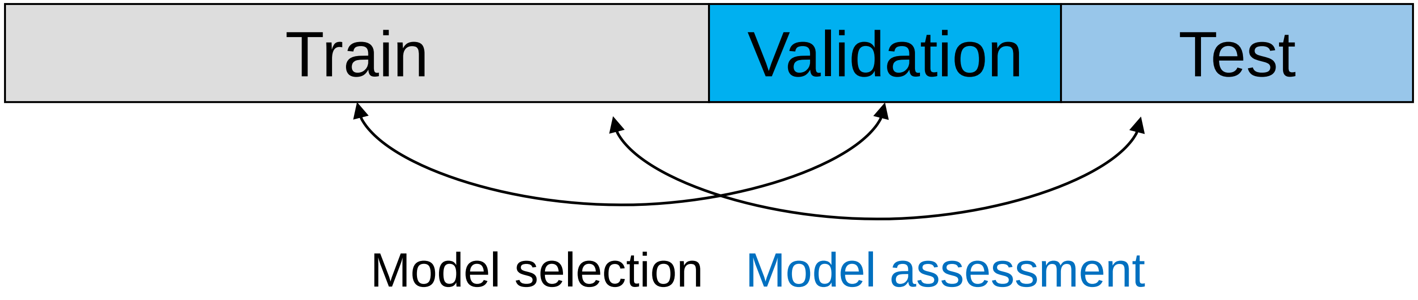
where f and g are increasing functions

Most methods are based on a tradeoff between *fitting error* (high variance) and *model complexity* (low bias)

Model Selection and Model Assessment

Model selection: Estimating performances of different models to choose the best one (produces the minimum of the test error)

Model assessment: Having chosen a model, estimating the prediction error on new data



Model Selection

Wide-spread methods for model selection are:

- Akaike Information Criterion (AIC)
  - $AIC = 2k - 2\ln(L)$, already discussed in the context of log. regression
  - k is the number of parameters, $\ln(L)$ the log likelihood

- Minimum description length (Risannen, 1978)
  - discussed later in this class

- Resampling methods
  - Cross validation, jackknife, bootstrap, etc.

- etc.

Outline for Today

- Bias-Variance Tradeoff
- **Resampling Methods**
- Gain and ROC Curves
- Comparison Studies
- Algorithmic Information Theory
- Computational Learning Theory

"All models are wrong; some are useful."

—George E. P. Box

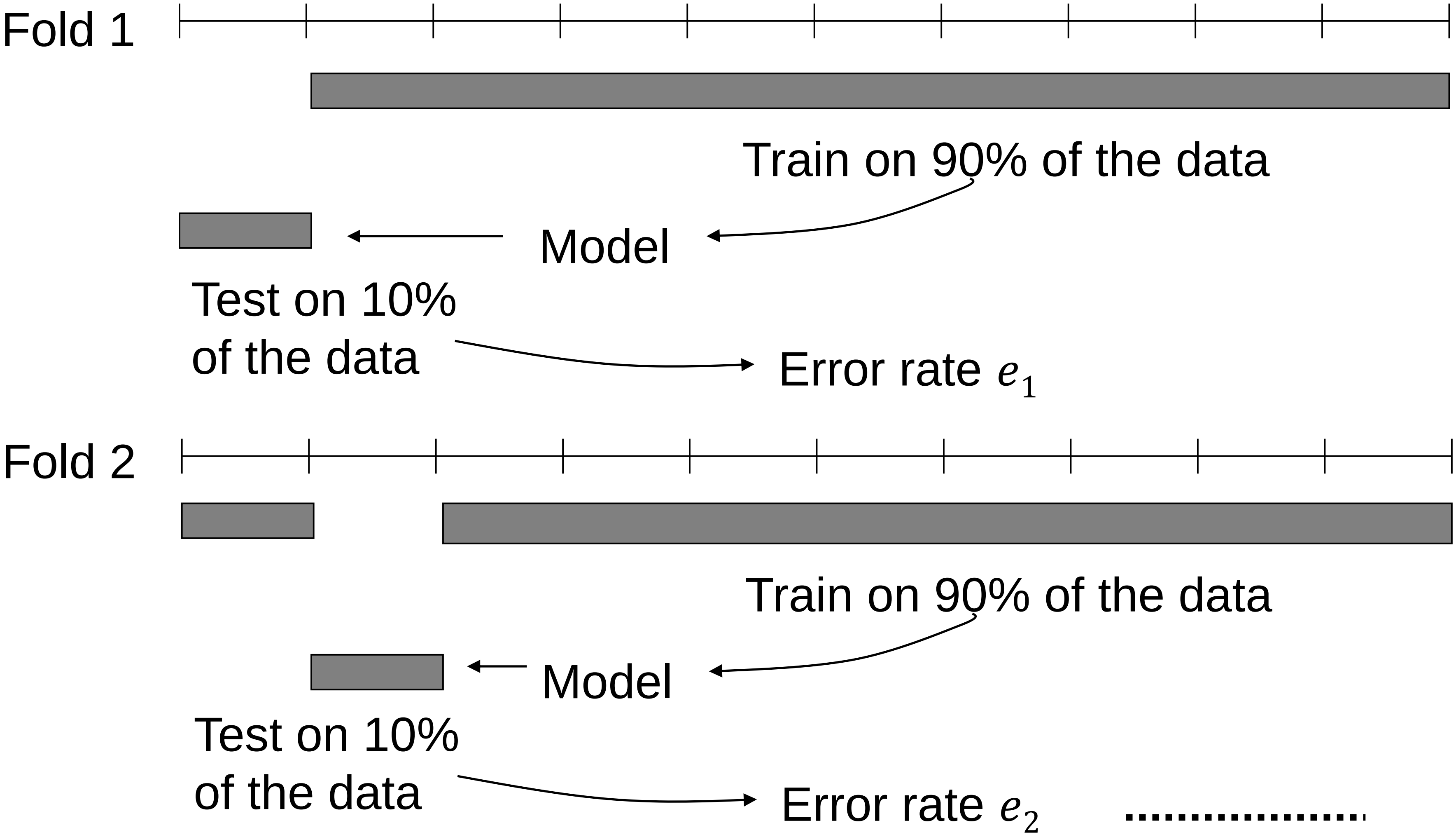
Testing and Training

- Holdout procedure (i.e., training and testing)
  - Reserve some data for testing (usually ~1/3)
  - Use remaining data for training
  - Use test data set to estimate the error rate (select a model)

- Stratified holdout
  - Guarantee that classes are (approximately) proportionally represented in the test and training set

- Repeated holdout (in addition)
  - Randomly select holdout set several times and average the error rate estimates

Cross Validation



Fold 1

Train on 90% of the data

Model

Test on 10% of the data

Error rate e_1

Fold 2

Train on 90% of the data

Model

Test on 10% of the data

Error rate e_2 ....

k-fold Cross-Validation

- Fixed number of k partitions of the data (folds)
- In turn: each partition is used for testing and the remaining instances for training
  - Finally each instance is used for testing once
- May use stratification
- Standard practice: stratified ten-fold cross-validation
- Error rate is estimated by taking the average of error rates
- *Select model* that performs best over all test subsets

$$\hat{e} = \frac{1}{k}\sum_{i=1}^{k} e_i$$

Comparing Error Rates

- Suppose we have two algorithms
  - Obtain two different models
  - Estimate the error rates for the two models
  - Compare estimates

$$\hat{e}^{(1)} < \hat{e}^{(2)} \; ?$$

  - Select the better one

- Problem?
  - Are there “significant” differences in the error rates?

Comparing Error Rates

- Estimated error rate is just an estimate (random)
- Student's paired t-test tells us whether the means of two samples are significantly different
- Construct a t-test statistic
  - Need variance as well as point estimates

$$t = \frac{\bar{d}}{s_d/\sqrt{k}}$$

Average of differences of error rates

Observed standard deviation of diff. in error rate

H$_0$: Difference = 0

If only the difference of correct classifications of two algorithms (i.e., not quantitative differences) are compared on the same test set, then a *binomial test* may be preferable.

Leave-One-Out Holdout (Jackknife)

- n-Fold Cross-Validation
  - n instances are in the data set
  - Use all but one instance for training
  - Each iteration is evaluated by predicting the omitted instance

- Advantages / Disadvantages
  - Maximum use of the data for training
  - Deterministic (no random sampling of test sets)
  - High computational cost
  - Non-stratified sample!

The Bootstrap

- Bootstrap works by sampling several times *with replacement* from training set to form a “bootstrap” data set.
- As a result some observations may appear more than once in a given bootstrap data set and some not at all.
- Each bootstrap data set contains n observations, sampled with replacement from the original data set.
- Then the model is estimated on a bootstrap data set, and predictions are made for original training set.
- This process is repeated many times and the resulting statistics are averaged.

Bootstrap as an Alternative to Cross-Validation

1. Number your observations $1,2,3, \dots n$
2. Draw a random sample of size n WITH REPLACEMENT.
3. Calculate your statistic (e.g. error rate, mean) with these data.
4. Repeat steps 1-3 many times (e.g., 500 times).
5. Calculate the variance of your statistic (e.g. error rate or the mean) directly from your sample of 500 statistics to learn about population statistics.
6. You can also calculate confidence intervals directly from your sample of 500 statistics. Where do 95% of statistics fall?

Example: re-sample 500 samples of n=50 with replacement, run logistic regression and examine the distribution of error rates (or other metrics)

Measuring Errors

		Predicted class	
		Yes	No
Actual class	Yes	True positive (TP)	False negative (FN) (Type I error)
	No	False positive (FP) (Type II error)	True negative (TN)

Error rate = # of errors / # of instances = (FN+FP) / N

Recall (hit rate) = # of found positives / # of positives
= TP / (TP+FN)

Precision = # of found positives / # of found
= TP / (TP+FP)

Specificity = TN / (TN+FP)

False alarm rate= FP / (FP+TN) = 1 - Specificity

Counting the Costs

- In practice, different types of classification errors often incur different costs
- Examples:
  - Predicting when customers leave a company for competitors (churn prognosis)
    - Much more costly to lose a valuable customer
    - Much less costly to act on a false customer
  - Loan decisions
  - Oil-slick detection
  - Fault diagnosis
  - Promotional mailing

		Predicted class	
		Yes	No
Actual class	Yes	True positive	False negative
	No	False positive	True negative

Cost-Sensitive Learning

- Most learning schemes minimize total error rate
  - Costs were not considered at training time
  - They generate the same classifier no matter what costs are assigned to the different classes
  - Example: standard decision tree learner

- Simple methods for cost-sensitive learning
  - Weighting of instances according to costs
  - Resampling of instances according to costs
    - E.g. increase the “*no*” instances in training, which yields a model that is biased towards avoiding errors on “*no”* instances. When testing on the original test data set, there will be fewer false positives.

Outline for Today

- Bias-Variance Tradeoff
- Resampling Methods
- **Gain and ROC Curves**
- Comparison Studies
- Algorithmic Information Theory
- Computational Learning Theory

> *“All models are wrong; some are useful.”*
>
> —George E. P. Box

Direct Marketing

- Find most likely prospects to contact

- Not everybody needs to be contacted

- Number of targets is usually much smaller than number of prospects

- Typical applications
  - retailers, catalogues, direct mail (and e-mail)
  - customer acquisition, cross-sell, churn prediction
  - ...

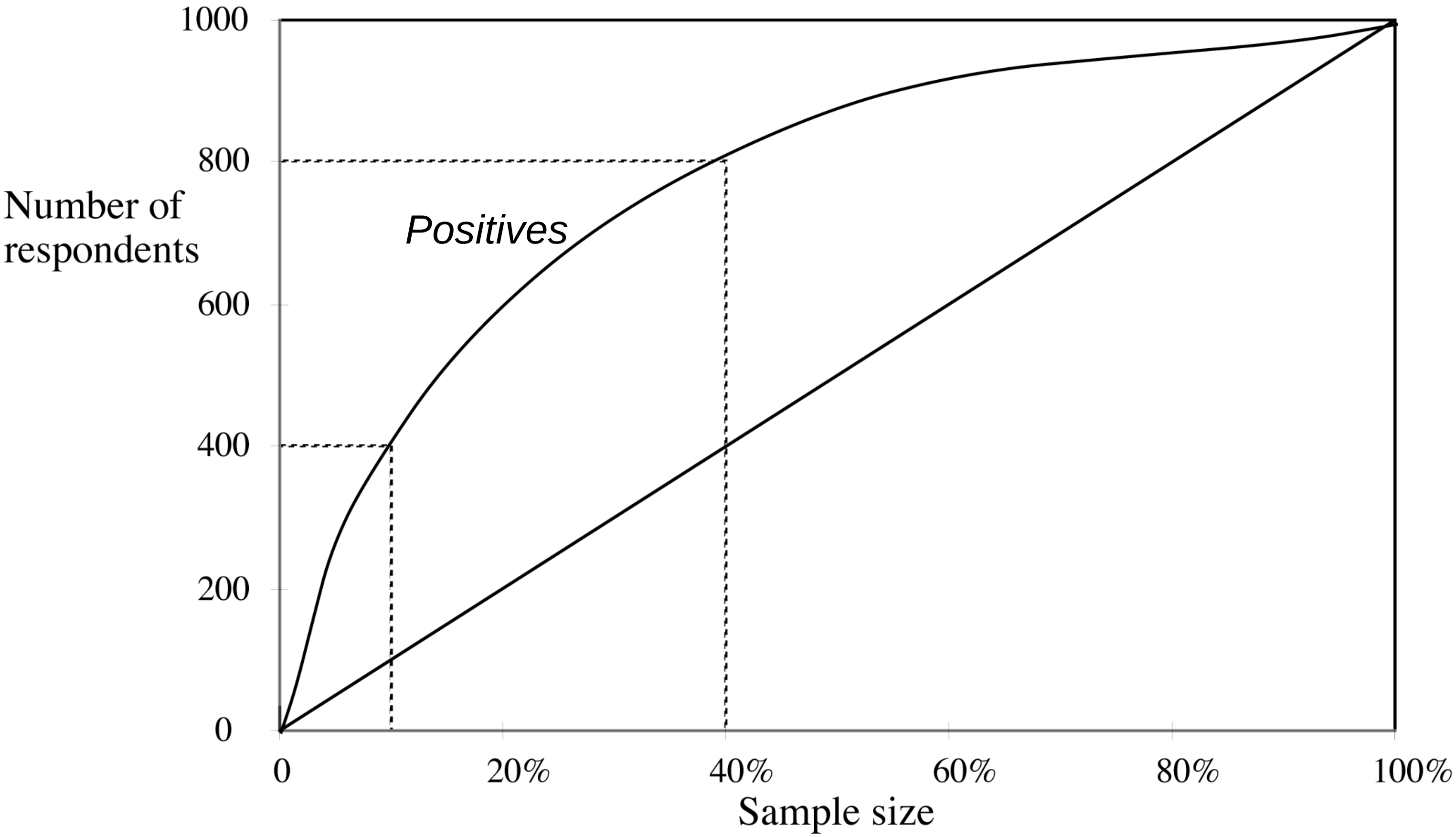
Direct Marketing Evaluation

- Accuracy on the entire dataset is not the right measure

- Approach
  - develop a target model
  - score all prospects and rank them by decreasing score
  - select top q % of prospects for action

- How to decide what is the best selection?

A Hypothetical Gain Curve



Generating a Gain Curve

- Instances are sorted according to their predicted probability:

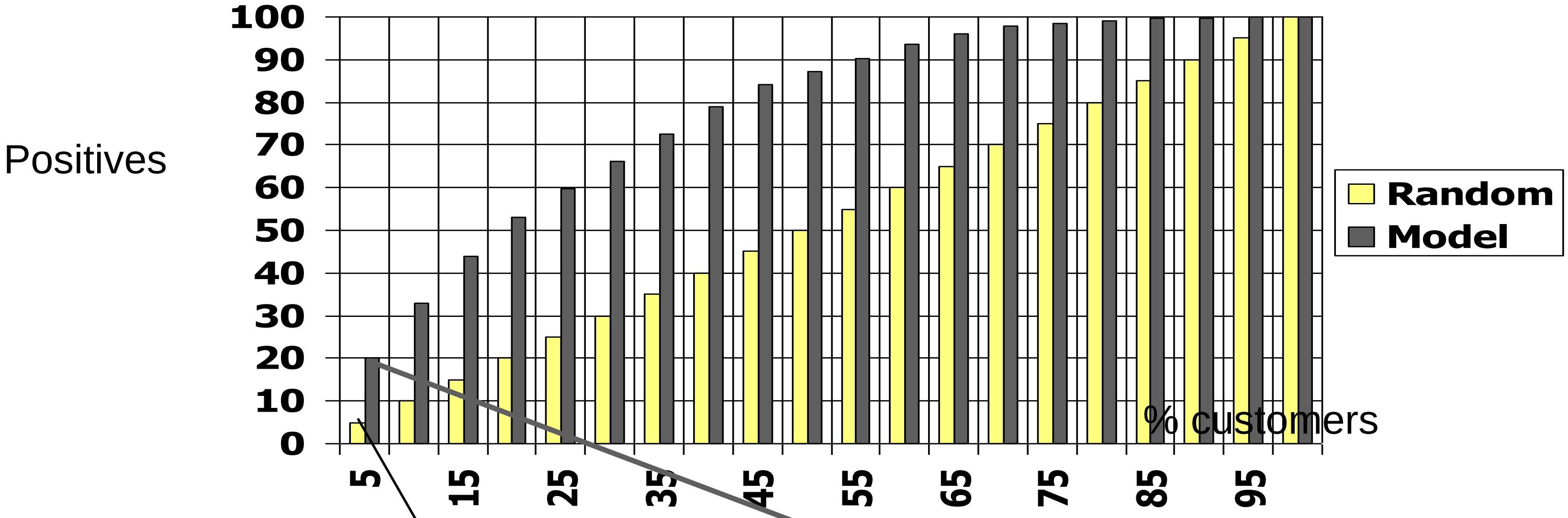
Rank	Predicted probability	Actual class
1	0.95	Yes
2	0.93	Yes
3	0.93	No
4	0.88	Yes
…	…	…

3 hits in top 5% of the list

If there are 15 targets overall, then top 5 has 3/15=20% of targets

- In a gain curve
  - x axis is sample size
  - y axis is number of positives

Gain Curve: Random List vs. Model-Ranked List



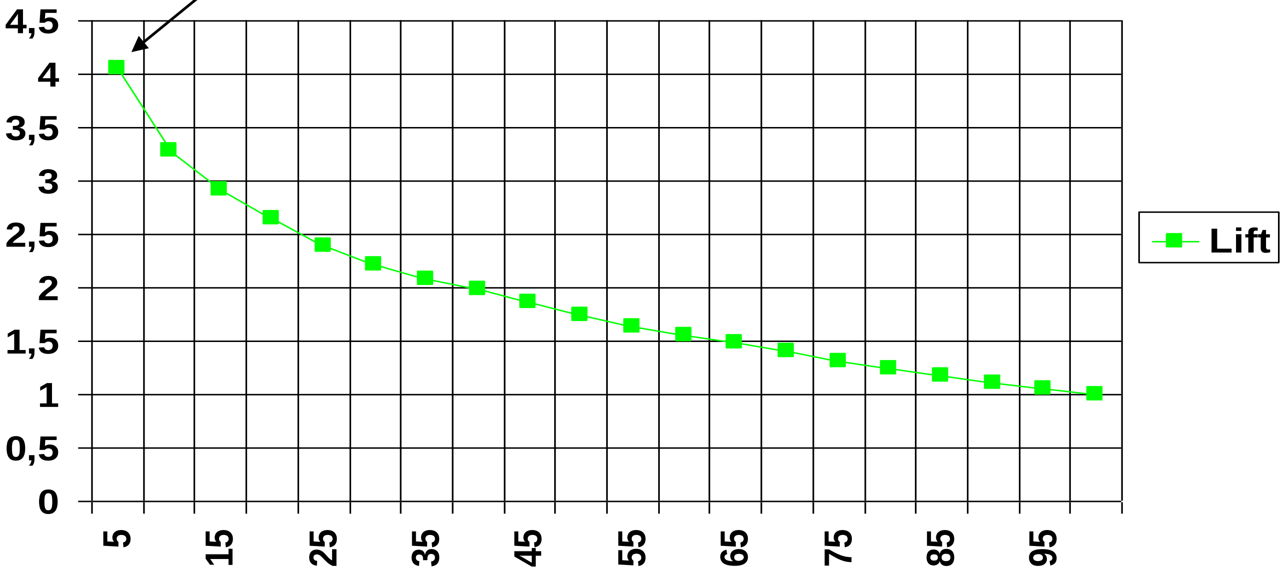
5% of random list have 5% of targets,

but 5% of model ranked list have 20% of targets Gain(5%)=20%.

Lift Curve

Lift(at 5%) = 20% / 5% = 4.0 times better than random

Lift(q) $= \text{Gain}(q)/q$



q (percent of the list)

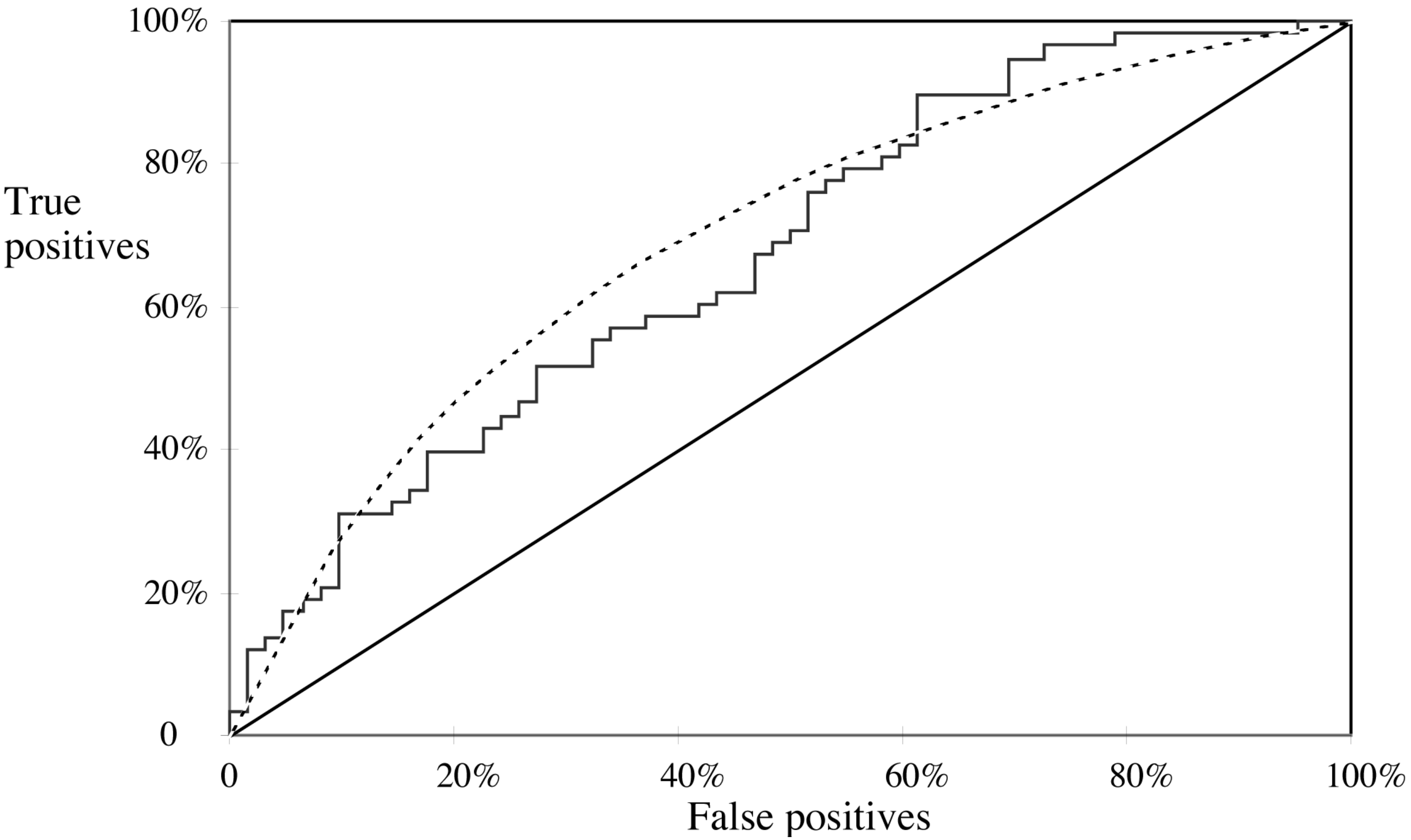
Note: Some (including Witten & Frank) use "Lift" for what we call Gain.

ROC Curves

- Differences to gain chart: recall vs. false alarm rate
  - y axis shows percentage of true positives in sample (rather than absolute number):
    - $TP\ rate\ =\ tp\ =\ 100\ *\ \mathrm{TP}/(\mathrm{TP}+\mathrm{FN})$
  - x axis shows percentage of false positives in sample (rather than sample size):
    - $FP\ rate\ =\ fp\ =\ 100\ *\ \mathrm{FP}/(\mathrm{FP}+\mathrm{TN})$

- ROC curves are similar to gain curves
  - "ROC" stands for "receiver operating characteristic"
  - Used in signal detection to show tradeoff between hit rate and false alarm rate over noisy channel
  - Go through all sizes of a sample and plot tp vs. fp

A Sample ROC Curve

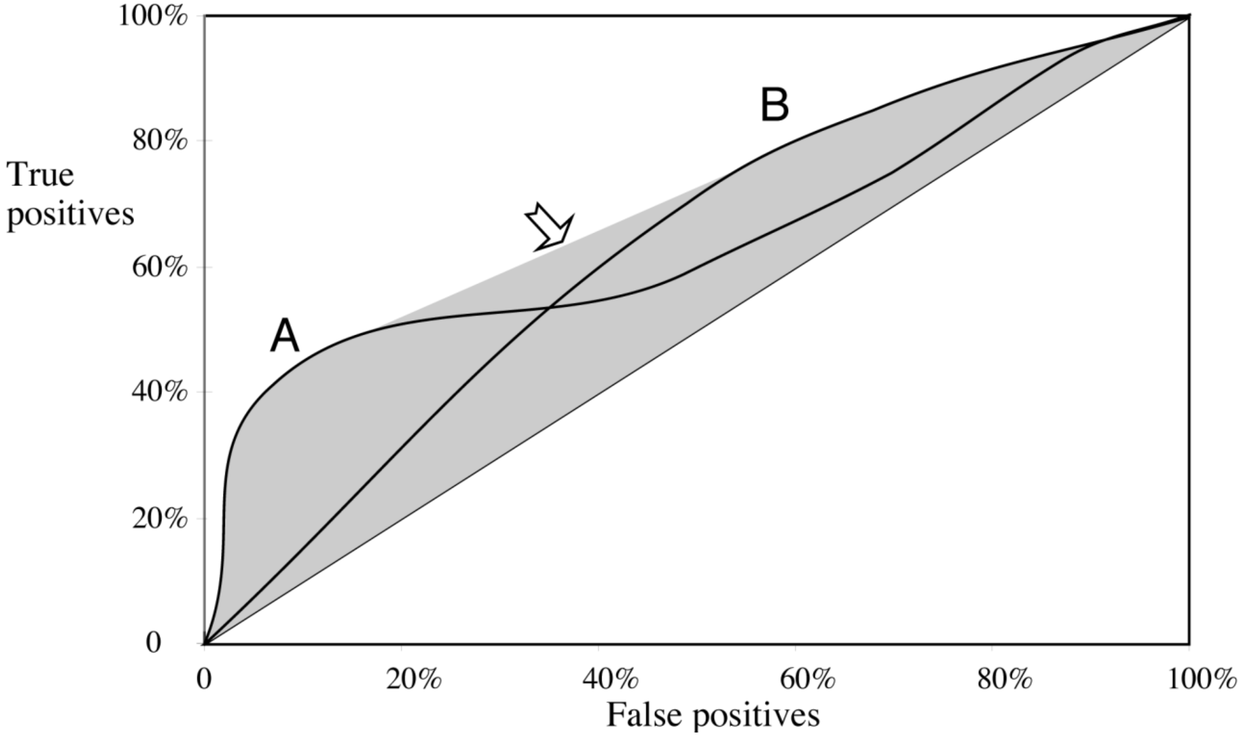


Jagged curve - one set of test data

Smooth curve - use cross-validation and average

ROC Curves for Two Classifiers

- For a small, focused sample, use method A (e.g., only interested in 40% of true positives)
- For a larger one, use method B



Outline for Today

- Bias-Variance Tradeoff
- Resampling Methods
- Gain and ROC Curves
- **Comparison Studies**
- Algorithmic Information Theory
- Computational Learning Theory

“All models are wrong; some are useful.”

—George E. P. Box

Comparison Studies of Classification Methods

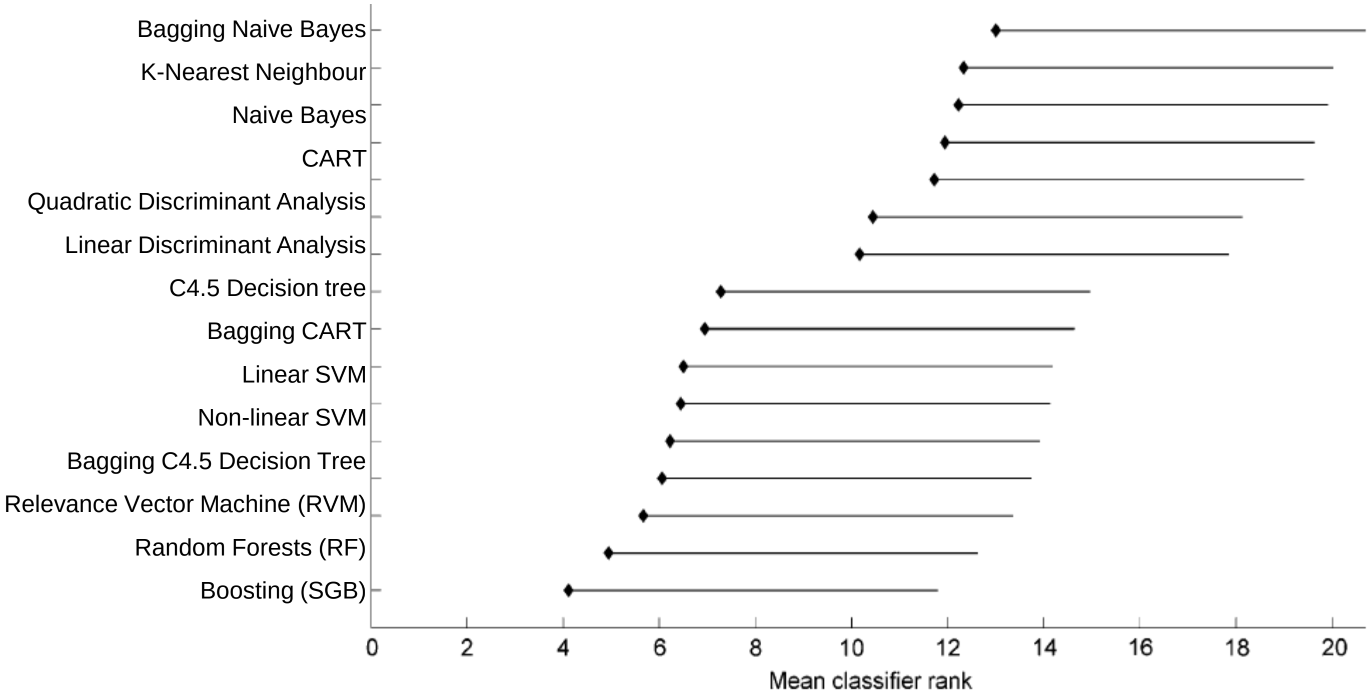
- Large number of classification techniques from the field of ML, NN and Statistics
- Many empirical comparisons in the 80‘s and 90‘s with contradictory results
- StatLog Project (mid 90‘s)
  - More than 20 methods and 20 datasets
  - Performance of methods depends very much on the data set
  - No general guidelines
  - Comparison study is advisable in special cases
- State-of-the-practice
  - Comparisons of several methods on a particular data set
  - Sometimes even automated „mass modeling“

Recommendations from StatLog

- Following methods should be considered in real-world comparison studies
  - Logistic Regression (Discriminant Analysis)
  - Decision trees
  - *K*-Nearest Neighbour
  - Non-parametric statistical methods
  - Neural Networks

- Decision trees and logistic regression are widely used in practice
  - High performance / low error rate
  - Speed of model building and classification
  - Easy to explain
    - as compared to NN or *k*NN

Another Study: Modern vs. Traditional Classifiers



Lessmann, Stefan, and Stefan Voß. "Customer-centric decision support." *Business & Information Systems Engineering* 2.2 (2010): 79-93.

Outline for Today

- Bias-Variance Tradeoff
- Resampling Methods
- Gain and ROC Curves
- Comparison Studies
- **Algorithmic Information Theory**
- Computational Learning Theory

“All models are wrong; some are useful.”

—George E. P. Box

Theoretical Foundations

Various theoretical streams aim for a fundamental understanding of machine learning:

- **Algorithmic information theory**
  - Kolmogorov complexity
  - Minimum description length principle
  - Etc.

- **Computational learning theory**
  - Probably approximately correct (PAC) learning
  - Vapnik–Chervonenkis (VC) theory
  - Bayesian inference
  - Algorithmic learning theory
  - Etc.

In the remainder of this class we cover important concepts from algorithmic information theory and discuss PAC learning.

Traditional Model Selection Criteria in Science

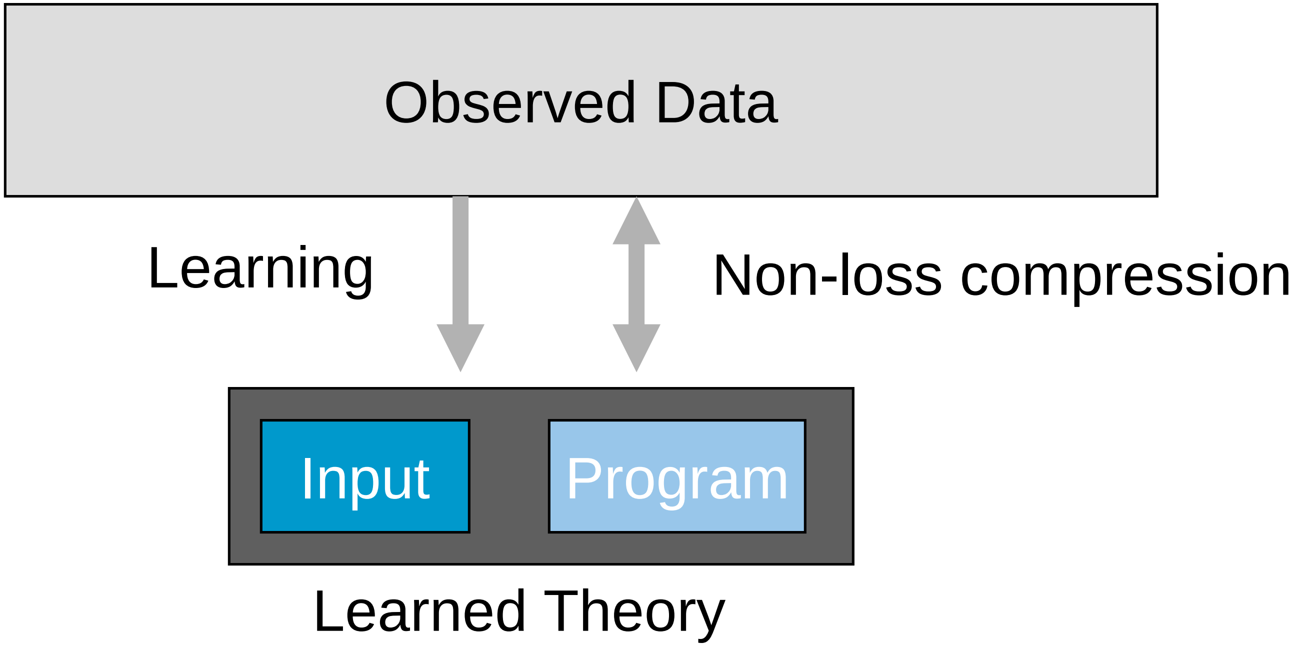
- Can we find a single ‘best’ model / classifier?
- Model selection criteria attempt to find a good compromise between:
  - The complexity of a model
  - Its prediction accuracy on the training data
- Reasoning: a good model is a simple model that achieves high accuracy on the given data
- Also known as Occam’s Razor: the best theory is the smallest one that describes all the facts

William of Ockham, born in the village of Ockham in Surrey (England) about 1285, was the most influential philosopher of the 14th century and a controversial theologian.

Elegance vs. Errors

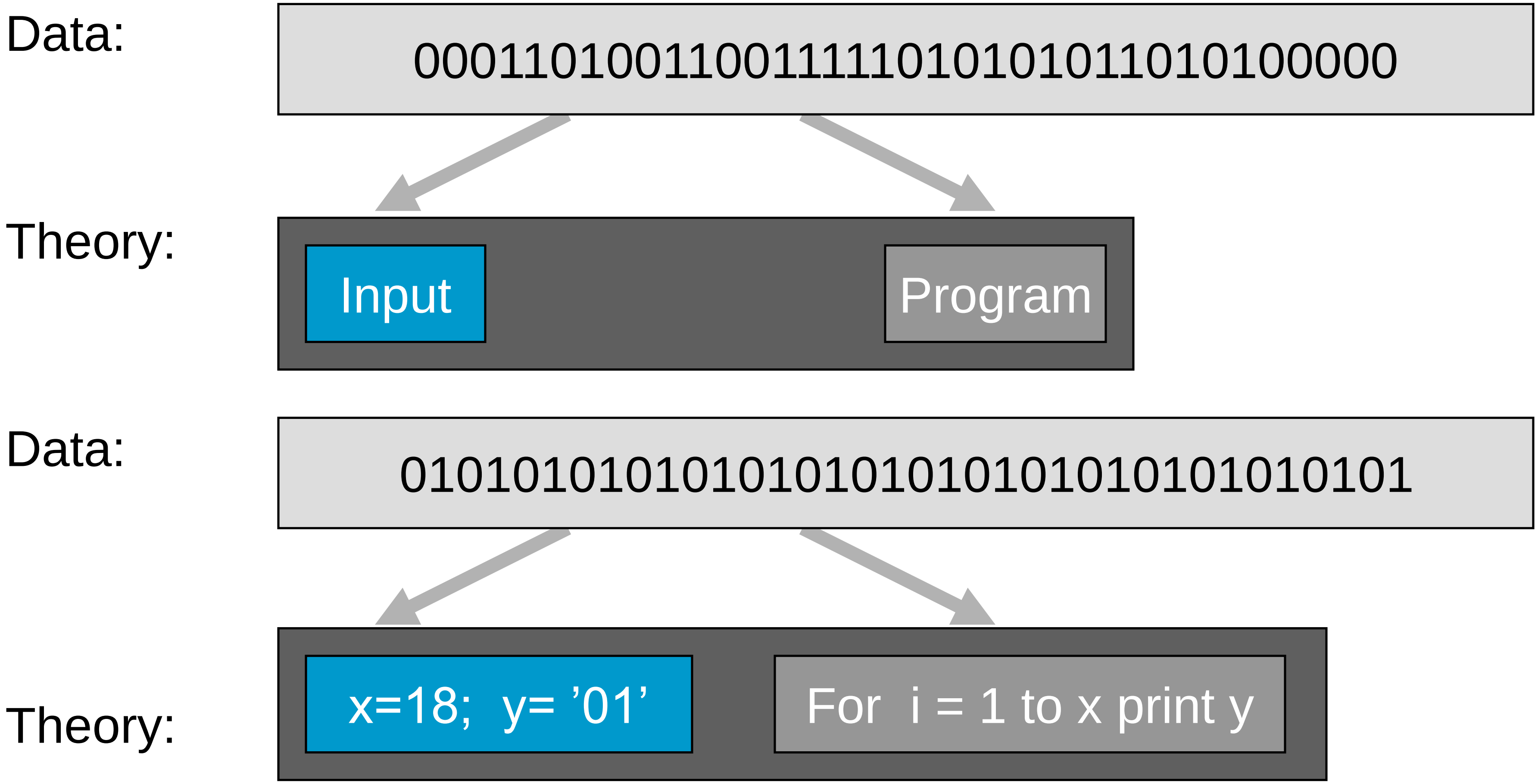
- Theory 1: very simple, elegant theory that explains the data almost perfectly
- Theory 2: significantly more complex theory that reproduces the data without mistakes
- Theory 1 is probably preferable

Inductive Learning Systems



|Theory| < |Data|

Example: Finite Binary String



Data:

000110100110011111010101011010100000

Theory:

Input

Program

Data:

01010101010101010101010101010101010101

Theory:

x=18; y= ’01’

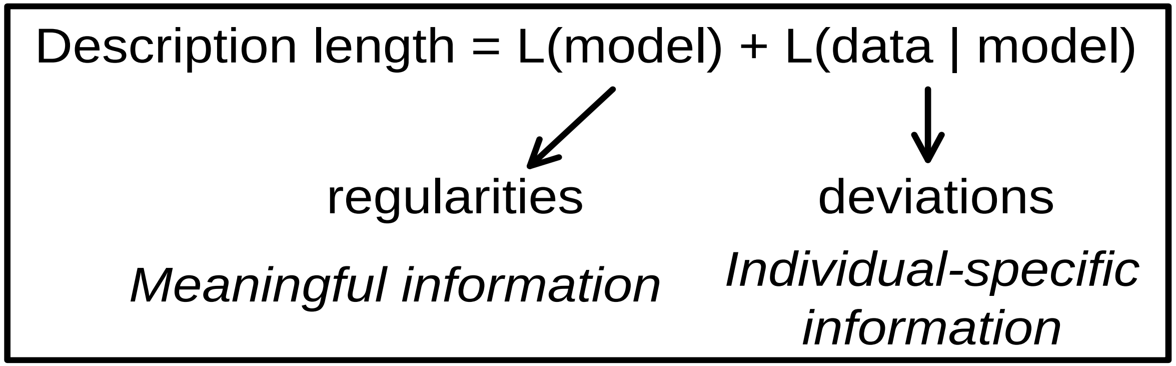
For i = 1 to x print y

|Theory| = |Program| + |Input| < |Data|

Randomness vs. Regularity

- 0110001101011010101
  - random string=incompressible=maximal information
- 010101010101010101
  - regularity of repetition allows compression
  - If the training set is ‘representative’ then regularities in the training set will be representative for regularities in an unseen test set

- A lot of forms of learning can be described as data compression in the following sense:



Description length = L(model) + L(data | model)

regularities

Meaningful information

deviations

Individual-specific information

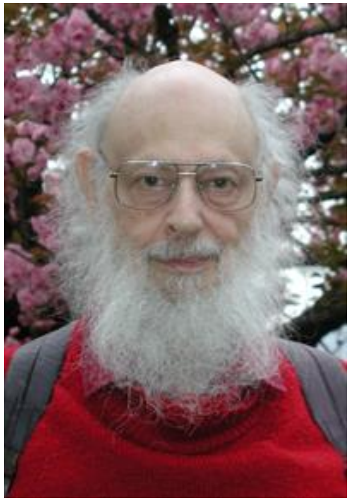
Kolmogorov Complexity

- The Kolmogorov complexity (K) of a binary object is the length of the shortest program that generates this object on a universal Turing machine
  - Random strings are not compressible
  - A message with low Kolmogorov complexity is compressible

- K as complexity measure is incomputable
  - So in practical applications it always needs to be approximated, e.g. by Lempel-Ziv (used in zip and unzip) compression or others

- Ray Solomonoff founded the *theory of universal inductive inference*, which draws on Kolmogorov complexity. Kolmogorov and Solomonoff invented the concept in parallel.

Kolmogorov



Solomonoff

Kolmogorov Complexity

- Let x be a finite length binary string and U a universal computer
- Let $U(p)$ be the output of U when presented with program p
- Let $l(p)$ be the length of program p
- The Kolmogorov complexity of string x is the minimal description length of x

$$K_U(x) = \min_{p:U(p)=x} l(p)$$

Kolmogorov Complexity is Not Computable

- Proof by contradiction (sketch):
  - Suppose there is a **function** KolmogorovComplexity(**string** s)
  - Then write a program that generates the shortest string w. Kolmogorov complexity $K(s) = n$

```
function GenerateComplexString(int n)
 for i = 1 to infinity:
     for each string s of length exactly i
       if KolmogorovComplexity(s) >= n return s
quit

function GenerateParadoxicalString()
     return GenerateComplexString($n_0$)
```

Length of GenerateComplexString: U
+ Number of bits to encode n_0: $\log_2(n_0)$
+ Overhead added by GenerateParadoxicalString
= $U + \log_2(n_0) + C$

$$U + \log_2(n_0) + C < n_0.$$

- TheGenerateComplexString generates the shortest string s with $K(s) = n_0$
- The length of this program is $< n_0$ if n_0 is large enough
- But s is not compressible, by definition of $K(s)$
- Contradiction!

The Minimum Description Length Principle

- MDL principle is a model selection criterion (like AIC)
- MDL restricts the set of allowed codes in such a way that it becomes possible (computable) to find the shortest codelength of the data, relative to the allowed codes
- The description length (DL) is defined as:

space required to describe a theory

+

space required to describe the theory’s mistakes

Learning = finding regularities = compression

Outline for Today

- Bias-Variance Tradeoff
- Resampling Methods
- Gain and ROC Curves
- Comparison Studies
- Algorithmic Information Theory
- **Computational Learning Theory (Outlook)**

“All models are wrong; some are useful.”

—George E. P. Box

Sample Complexity

- The sample complexity is the number of training-samples that we need to supply to the algorithm, so that the function returned by the algorithm is within an arbitrarily small error of the best possible function, with probability arbitrarily close to 1.
- *No Free Lunch theorem*: there is no algorithm that can learn the globally-optimal target function using a finite number of training samples.
- However, if we are only interested in a particular class of target functions (e.g., linear functions) then the sample complexity is finite and can be described by the *VC dimension* or *Rademacher complexity*.
- In *probabilistically approximately correct (PAC)* learning, one is concerned with whether the sample complexity is polynomial.
  - If VC dimension of a learner is finite, then it is PAC-learnable.

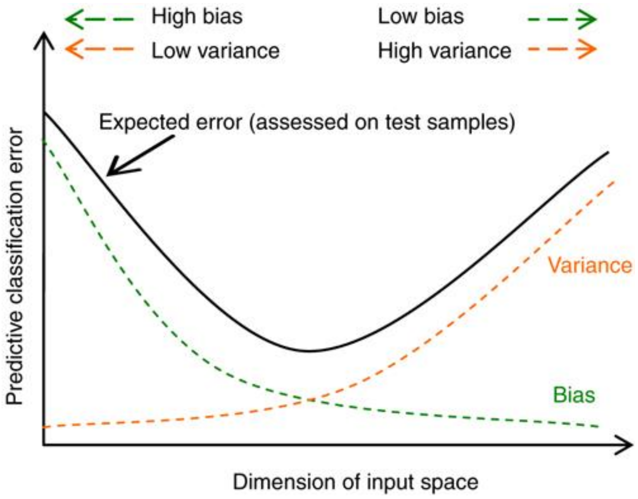
VC Dimension and Risk

Given some classifier, let H be its VC dimension, i.e. the *representational power* of a classifier (not defined here in detail)

- Risk $R(\theta)$ is the expected error on a test data set
- Empirical risk $R^{emp}(\theta)$ is the error on the training data

Look at the trade-off between test and training data

- Underfitting: with a simple learner, the difference will be small
- Overfitting: with a complex learner, the test error might be much worse



VC Dimension

The VC dimension is related to the capacity of a learner:

- It is the cardinality of the largest set of points that the learner can „explain“.
- A high value of the VC dimension gives good generalization.
- For a small VC dimension, a small training set may lead to good generalization.

VC dimension allows for a bound on the test error (without knowing the test data)

test error $\leq$ training error + $\sqrt{\frac{H \log\left(\frac{2n}{H}\right)+H-\log(\frac{\eta}{4})}{n}}$ (penalty)

η is a small number
n is the number of data points
H is the VC dimension

If H is small or n is large, then the bounding term is small.

Structural risk minimization: select n and H (model complexity) of a learner such that risk is minimized.