COMP 4613

Artificial Intelligence 1

Daniel L. Silver

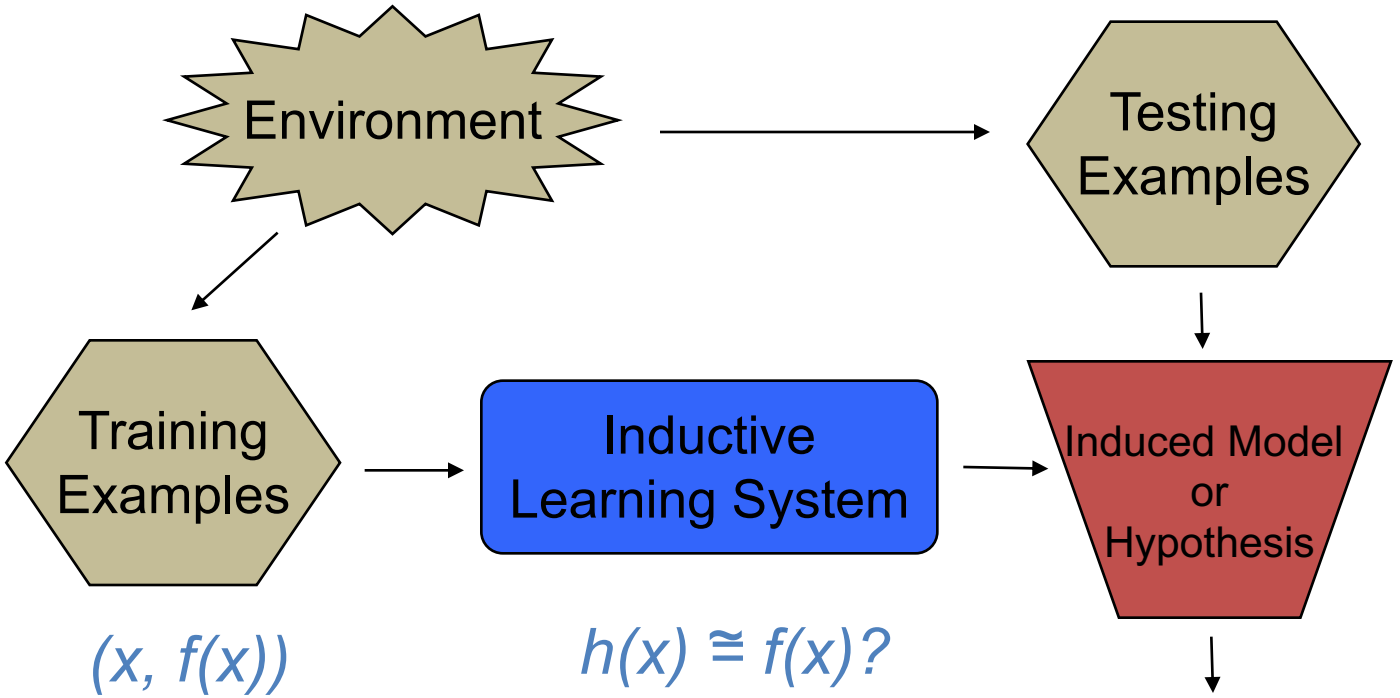
Model Evaluation and Selection

Outline

- Model performance and confidence
- Model selection
- Testing the difference between two models
- Testing the difference between two DM methods
- From Error rates to Loss functions
- Regularization - MDL – p.671
- Hyperparameters and tuning

Supervised Inductive Learning

Basic Framework for Inductive Learning



Focus is on developing models that can accurately classify new examples.

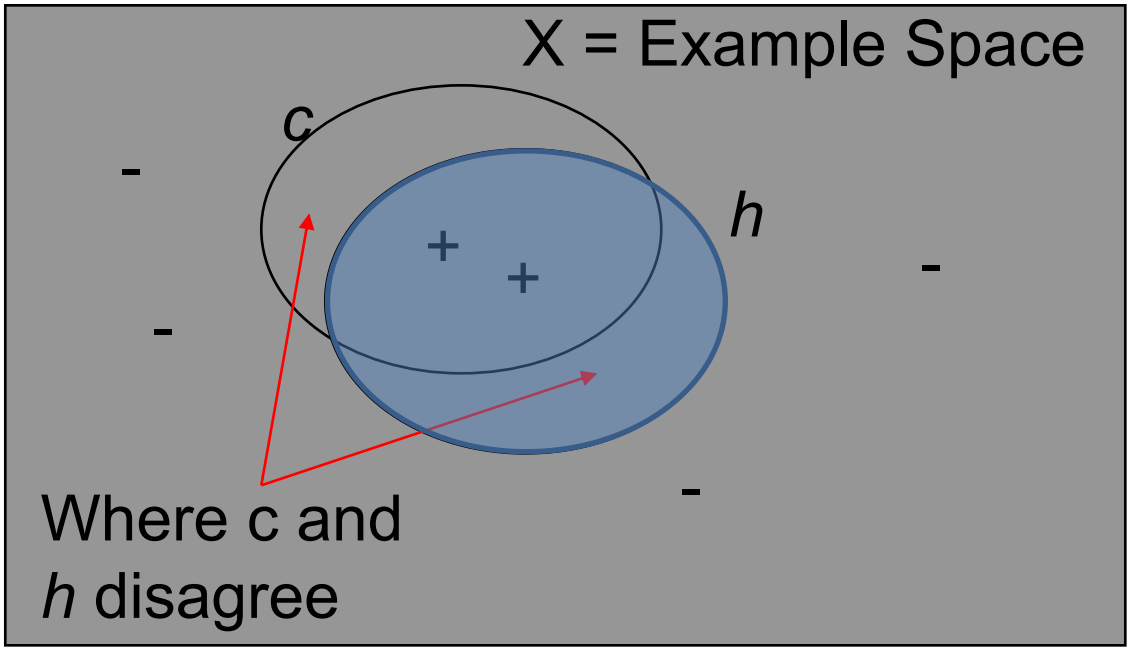
Model Evaluation

- ML Goal is to select a hypothesis (model) that will do well on future examples; it should "generalize" to future instances

x = input attributes

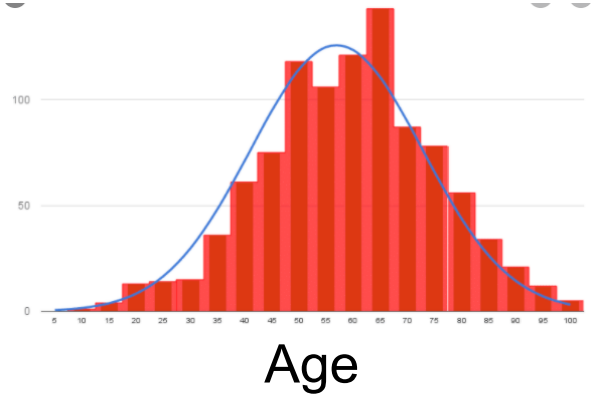
c = true class function

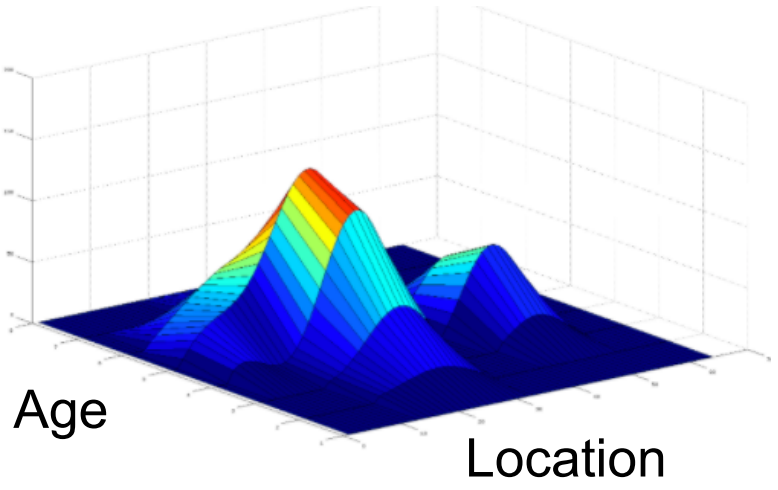
h = hypothesis (model)



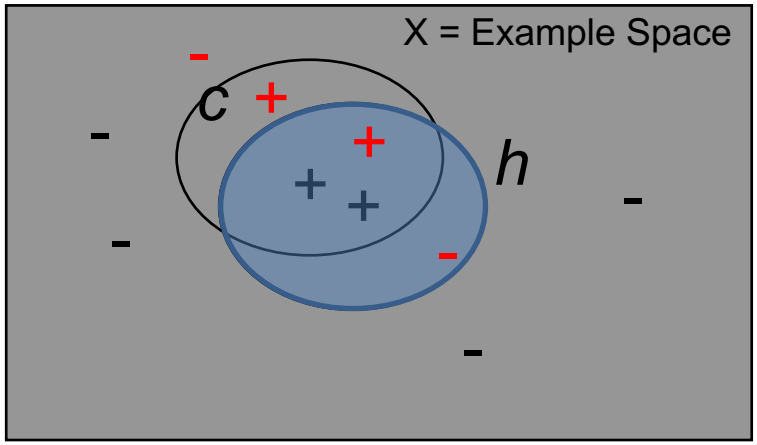
Model Evaluation

- **Stationarity** is assumed:
  - Future examples will be like past examples
  - Drawn from the same probability distribution
  - Are independent of each other
  - They are therefore i.i.d. = independent and identically distributed





Model Evaluation

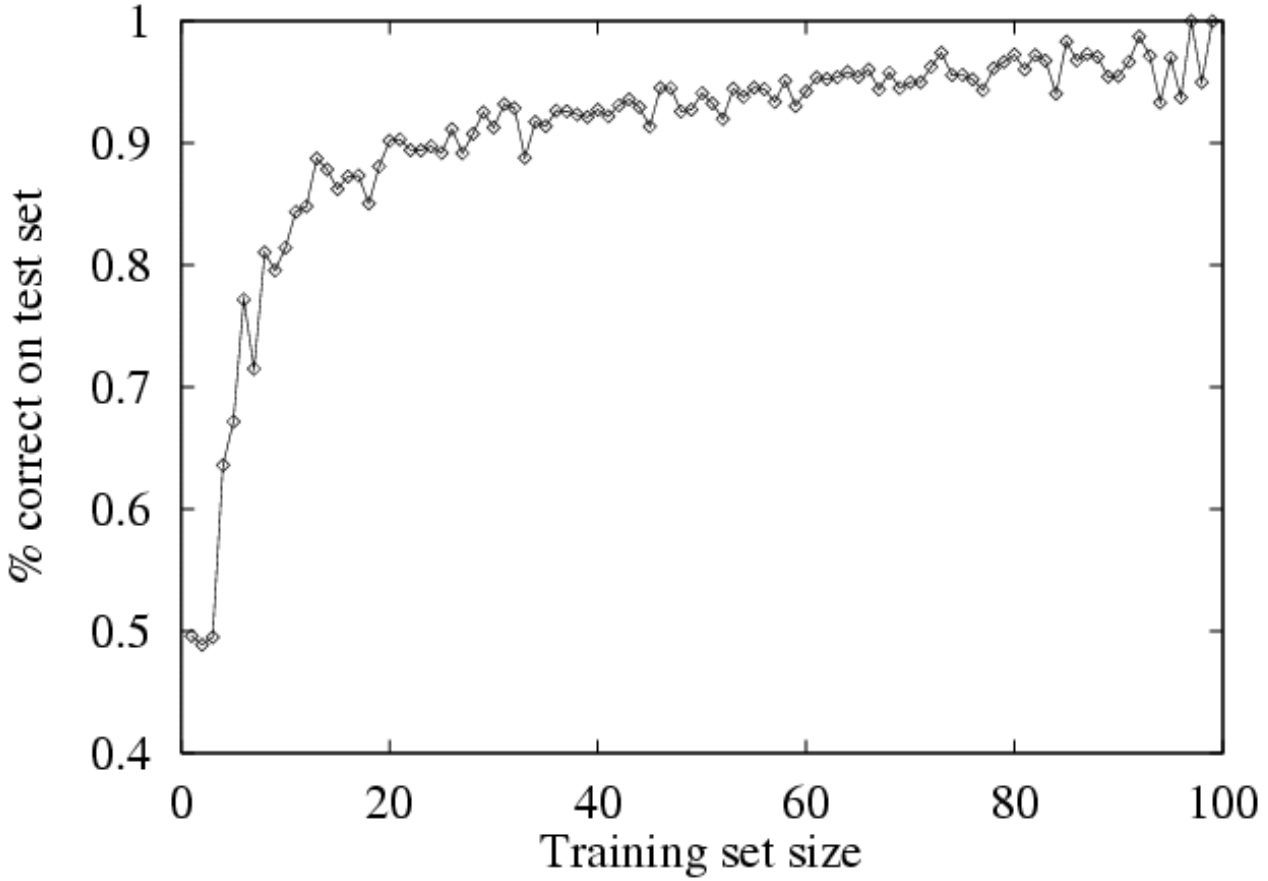


Three notions of error:

- ***Training Error*** **[-,+]**
  - Probability an example of the training set is misclassified
- ***Test Error*** **[-,+]**
  - Probability an example of the independent test set is misclassified
- ***True Error*** **[all of X]**
  - Probability an example of the entire population of possible examples, X, would be misclassified
  - Rarely can be measured
  - Must be estimated by the Test Error

Model Evaluation

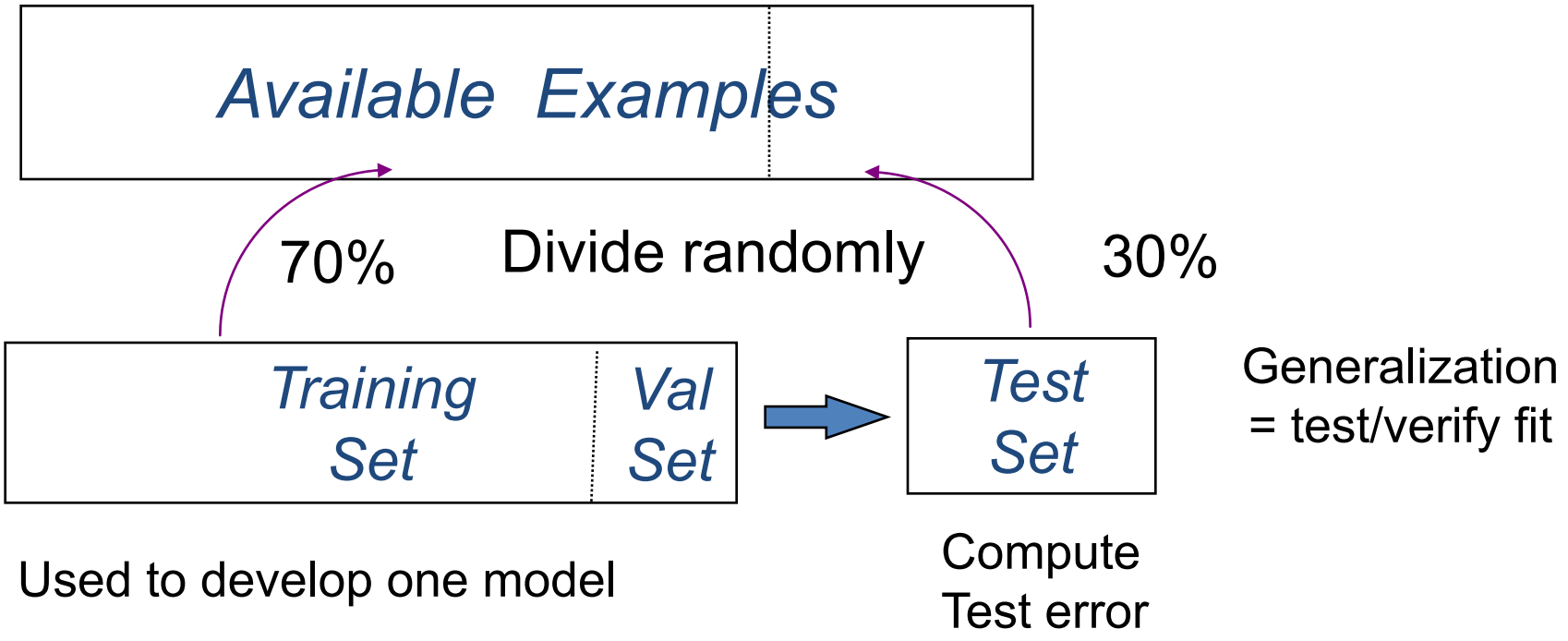
Learning curve = % error OR correct on test set as a function of training set size



Model Accuracy and Confidence

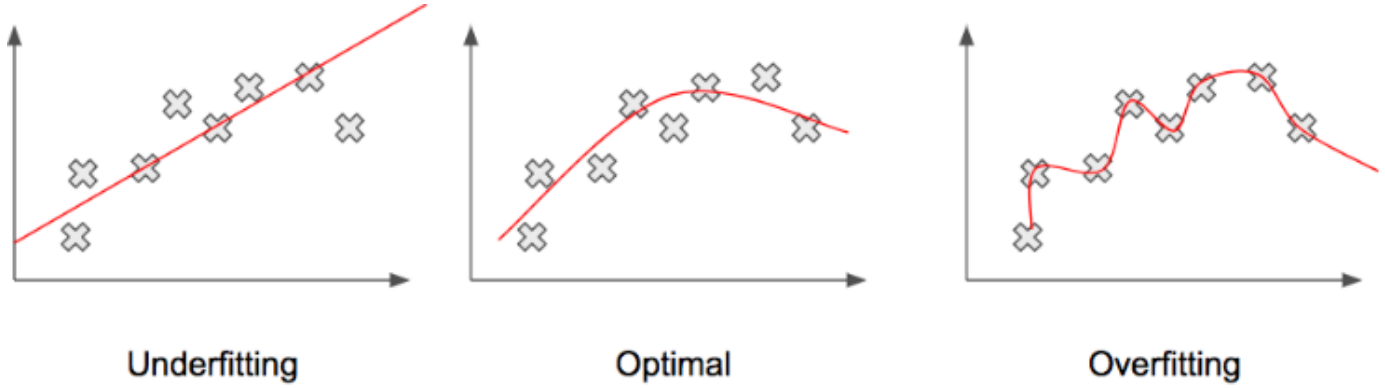
Approach #1: Large Sample

When the amount of available data is large ...



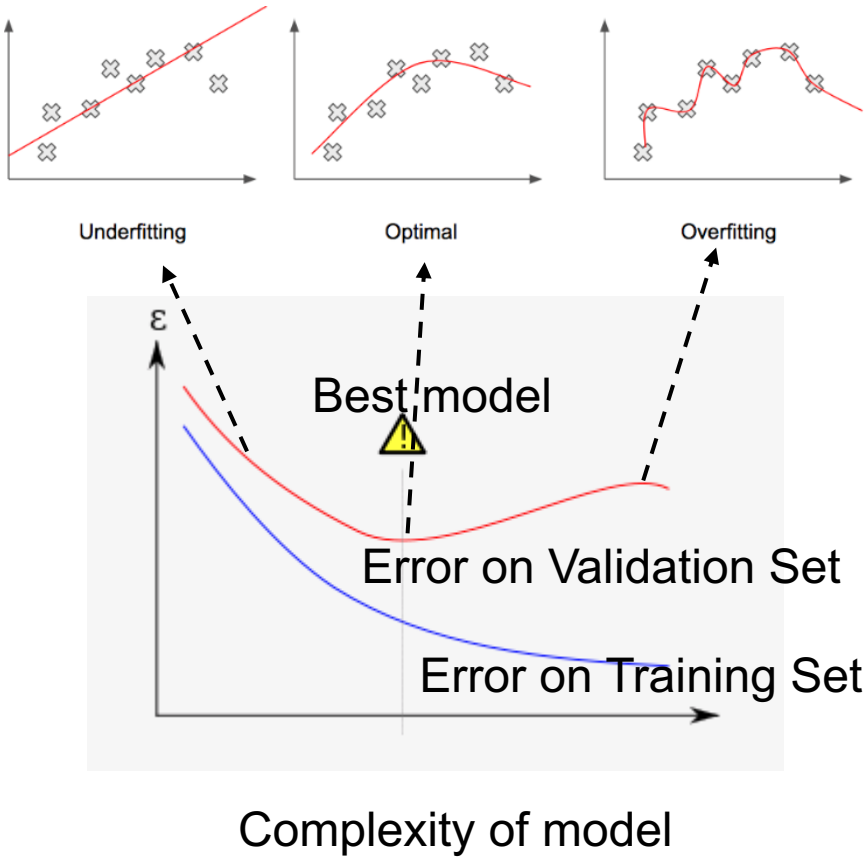
Model Selection

- The objective of learning is to achieve good *generalization* to new cases, otherwise just use a look-up table.



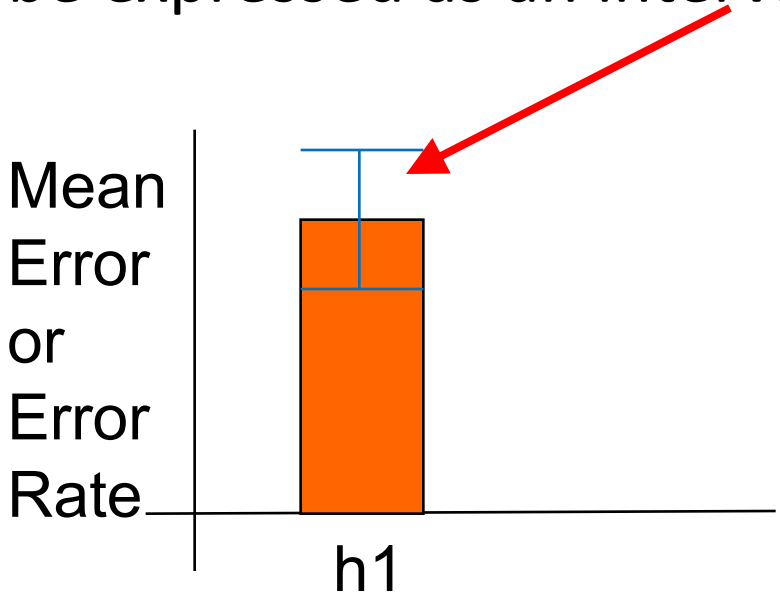
Model Selection

- Use a separate *validation* or *tuning set* of examples
- Monitor error on the val. set as model complexity increases
- Select best model at point of lowest validation set error. Just prior to over-fitting model to training data



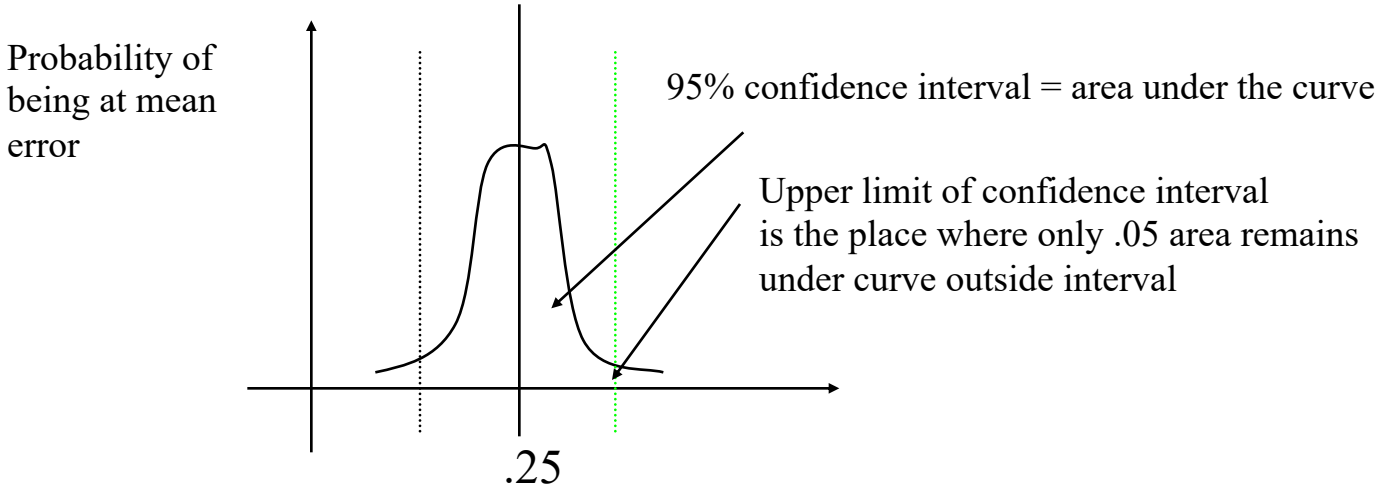
Model Accuracy and Confidence

- Preferably, a separate verification (test) set is used to judge model performance
- Statistical confidence in the accuracy of a model can be expressed as an interval



Model Accuracy and Confidence

- **Estimating True Error from Test Error:**
  - **If model has a mean test set error of 0.25**
  - **How well does this represent the true error if we desire 95% confidence? That is how badly can we expect it to do in future.**
  - **Requires statistics of a binomial distribution (nominal target) or normal distribution (continuous target) around the mean**



Model Accuracy and Confidence

- Confidence interval can be computed:
  - Nominal (binary) target variable - Given an error rate of P from a sample of *n* examples, then the 95% conf. interval

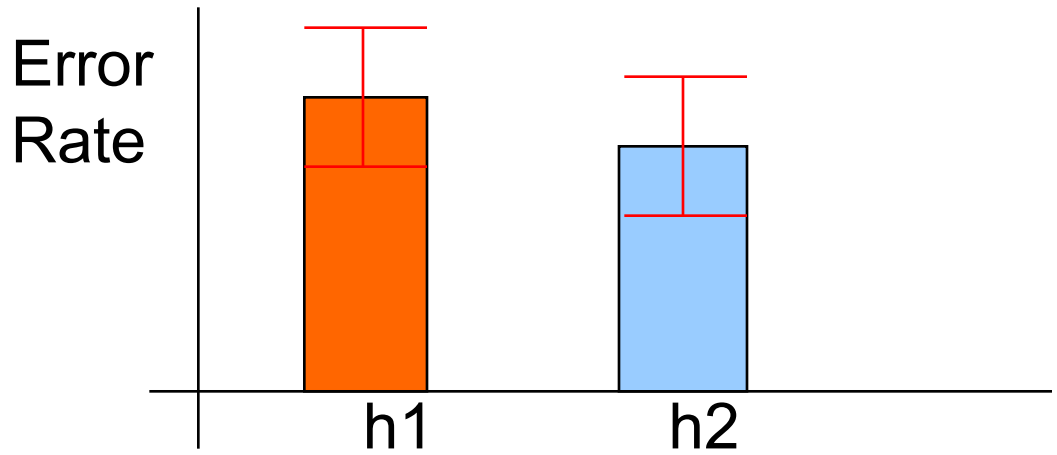
    = P ± 1.96 sqrt(P(1-P)/n) = P ± 1.96 * stdev

    = P ± 1.96 (sqrt[0.25(1-0.25)/100]) = P ± (1.96 * .043)

    where P = error rate = number incorrect / *n*
  - Continuous target variable - Compute mean error over *n* examples and confidence interval using Excel (evaluate_models.xls) (see ACORN webpage for download)
  - *Strictly speaking this is for n >= 30*

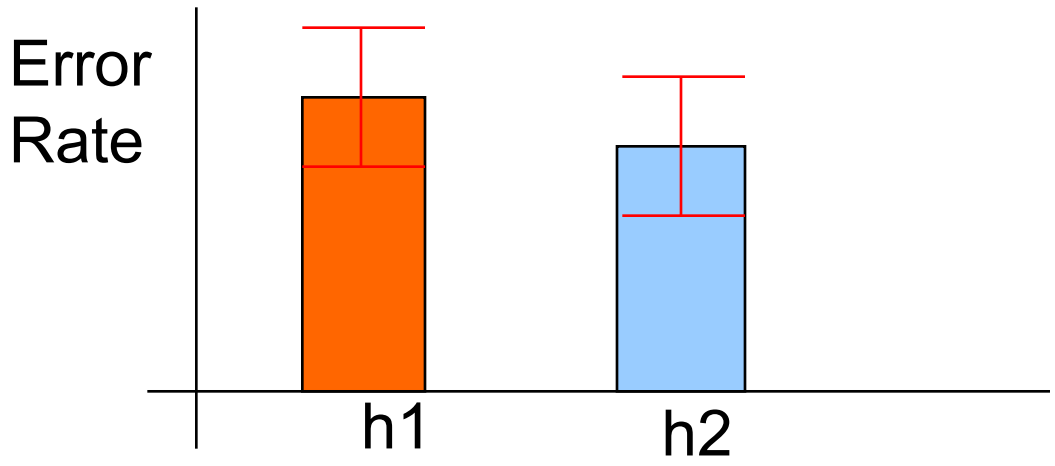
Testing the Difference Between Two Models

- Which of the following two hypotheses is the better? … h1 or h2 ?



Testing the Difference Between Two Models

- We can visually check the overlap of the error bars
- However, this is not terribly precise



Testing the Difference Between Two Models

- Assumption: If some measurable characteristic of the models is statistically different then we will consider the models different
- We will focus on the characteristics: mean error rate (proportion incorrect) which can be computed from the test results

Testing the Difference Between Two Models

- Nominal (binary) target variable
  - Use a Difference of Proportions Test
- Continuous target variable
  - Use a Difference of Means Test
- For 95% confidence in a difference then p-value statistic must be <= 0.05

  (see evaluate_models.xls)

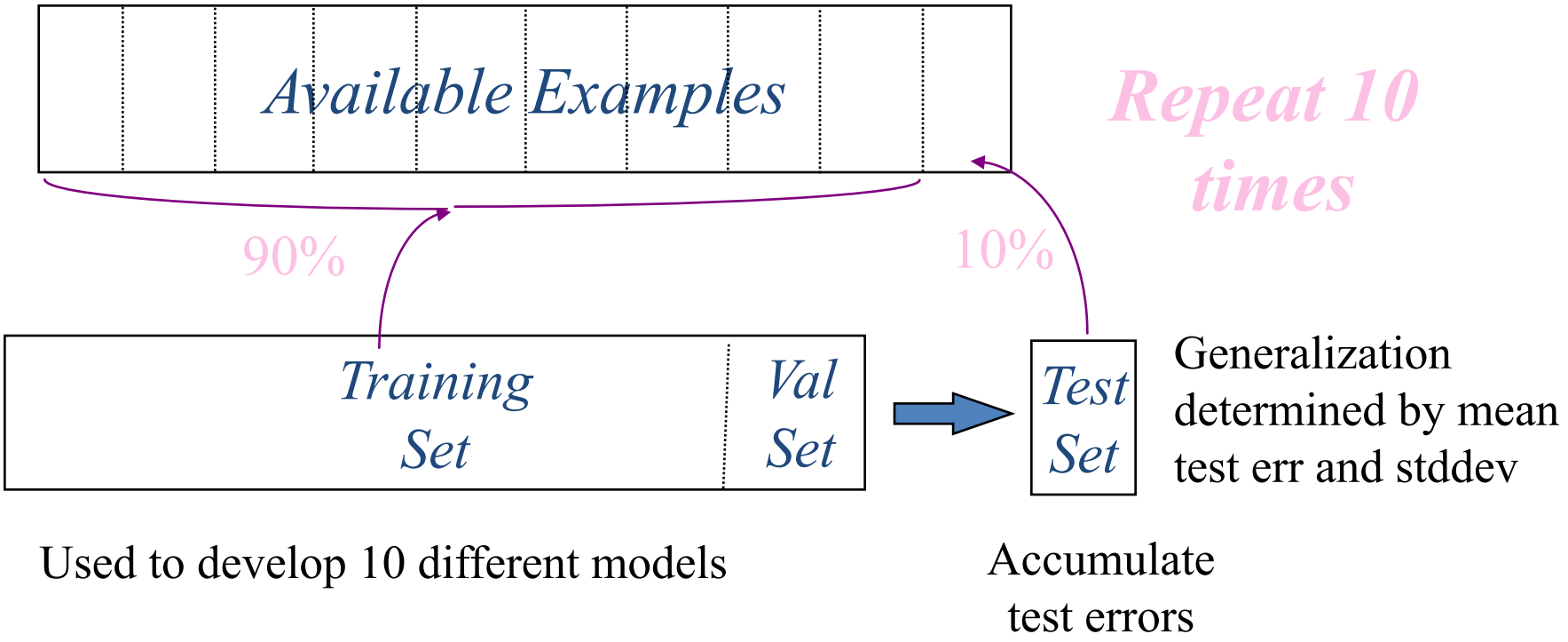
Testing with Small Data

- Cross-validation must be performed o get maximum use from available data
- Requires generating several models with different train and test sets
- Uses the error rate from each of the test sets

Testing with Small Data

Approach #2: Cross-validation

Builds multiple models from available data ...



Testing with Small Data

- A Difference of Means T-test can be used to determine a p-value statistic
- For 95% confidence in a difference then p-value statistic must be <= 0.05

  (see evaluate_models.xls)

Example: Using Census Data

- **Problem:** To identify males given census data
- **Performance measure:**
  - Test set error
- **Model generation:** IDT and ANN

Example: Using Census Data

- **Record results:** Goodness of fit stats on test set for 10 different models
  - Mean fitness: ANN= 26.6, IDT = 31.8
- **Test difference between methods**: Use a difference of means T-test (see evaluate_models.xls)
  - p-value = 0.00415
  - Since p-value < 0.05, the two models are significantly different

ANN Training

How do you ensure that a neural network has been well trained?

- Objective: To achieve good generalization accuracy on new examples/cases
- Establish a maximum acceptable error rate
- Train the tree using a method to prevent over-fitting by adjusting the
  - Architecture: number of hidden nodes
  - Hyperparameters: learning rate, momentum
- Test the trained network against a separate test set

ANN Training

- Try a reasonably wide range of C and M learning parameters [results on test set]

#H➔ LR	10	25	50
0.1	73%	78%	70%
0.01	75%	80%	72%
0.001	67%	71%	65%

ANN Training

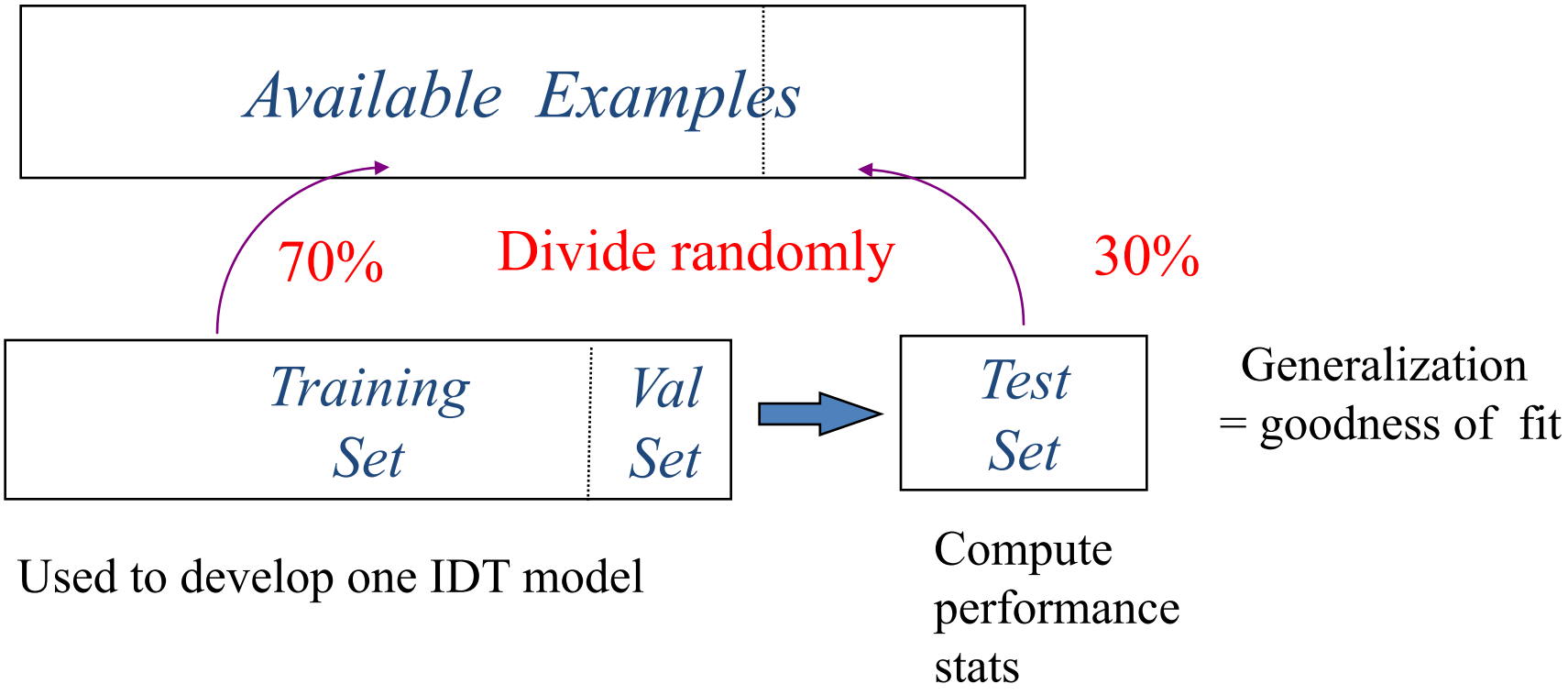
- Then narrow in on the best range and fine-tune the parameters [results on test set]

#H➔ LR	20	25	30
0.01	75%	79%	76%
0.03	79%	**81%**	80%
0.05	71%	78%	75%

IDT Evaluation

Approach #1: Large Sample

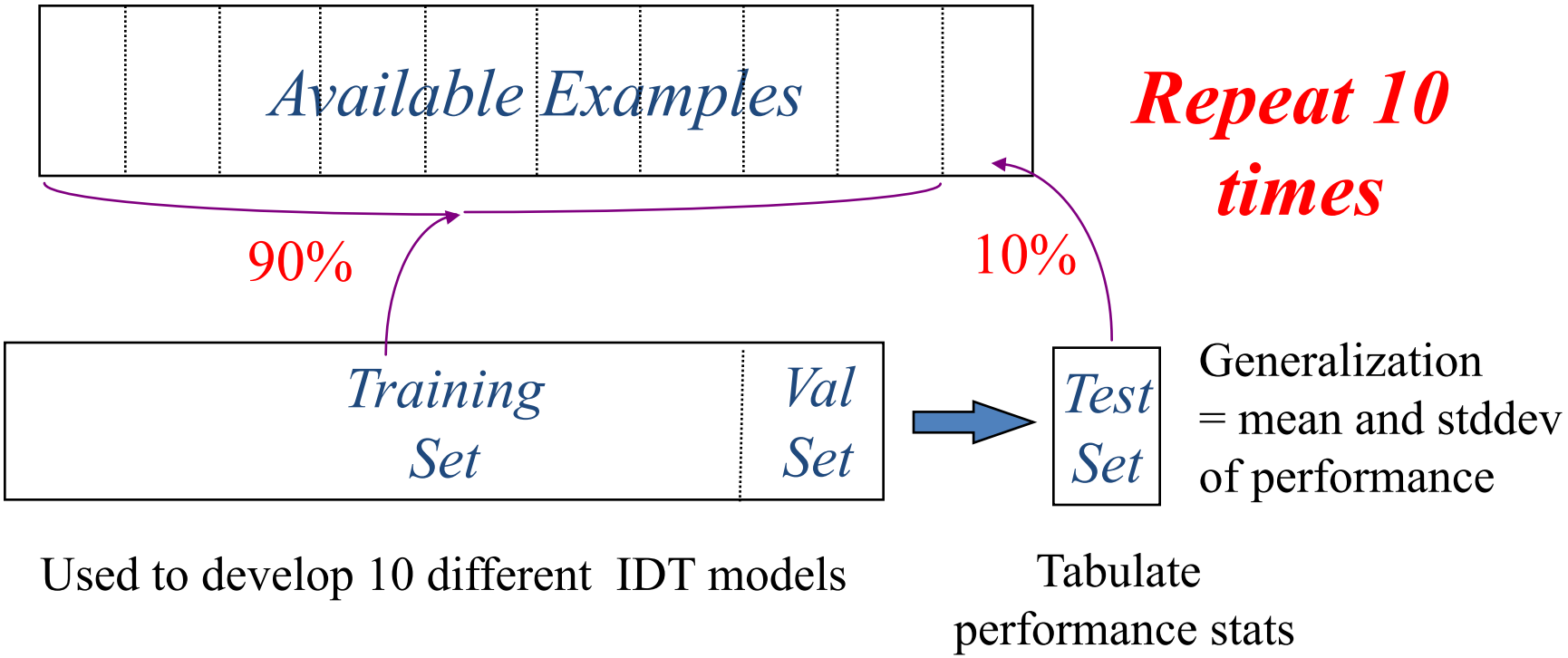
When the amount of available data is large ...



IDT Evaluation

Approach #2: Cross-validation

When the amount of available data is small ...



The IDT Application Development Process

Guidelines for inducting decision trees

1. IDTs are good method to start with
2. Get a suitable training set
3. Use a sensible coding for input variables
4. Develop the simplest tree by adjusting tuning parameters (significance level)
5. Use a method to prevent over-fitting
6. Determine confidence in generalization through cross-validation

References

- Cross-validation

  https://scikit-learn.org/stable/modules/cross_validation.html

THE END

danny.silver@acadiau.ca