Overfitting, Model Selection

CSC 461: Machine Learning

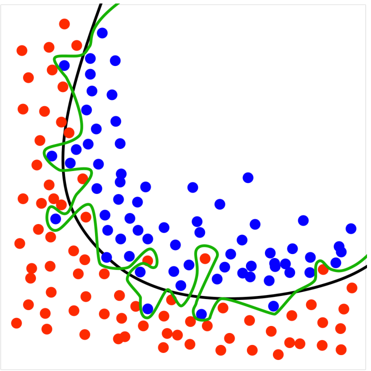
Fall 2021

Prof. Marco Alvarez
University of Rhode Island

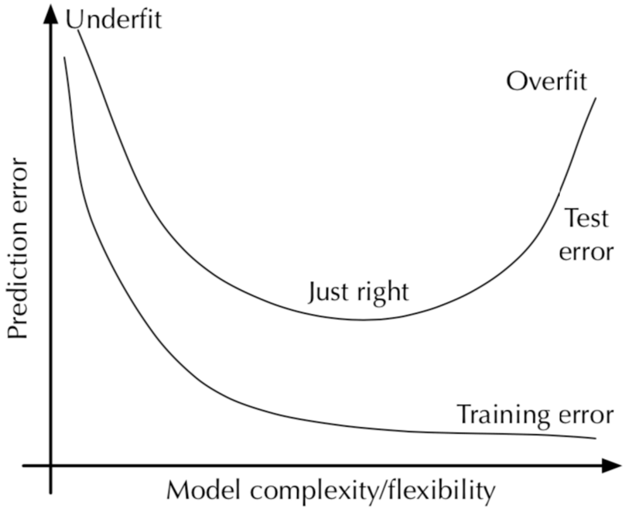
Overfitting

Overfitting

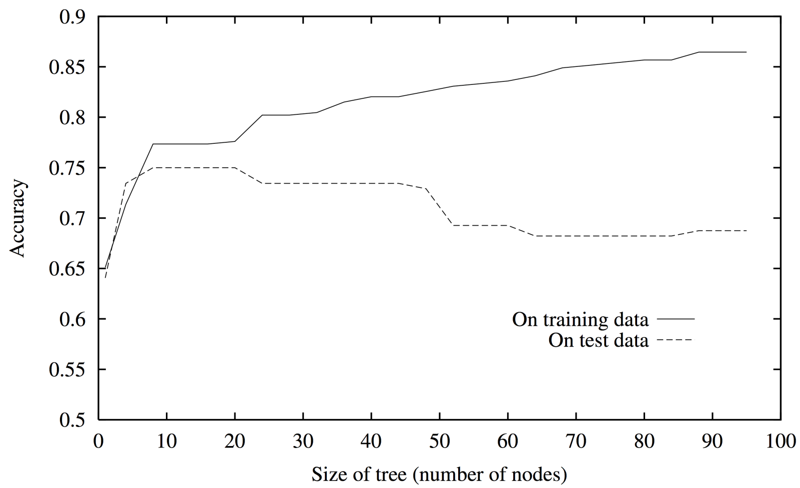
- Learning a **model** that "knows" the training data very well but does not **generalize**



Model complexity



Model complexity (DTs)

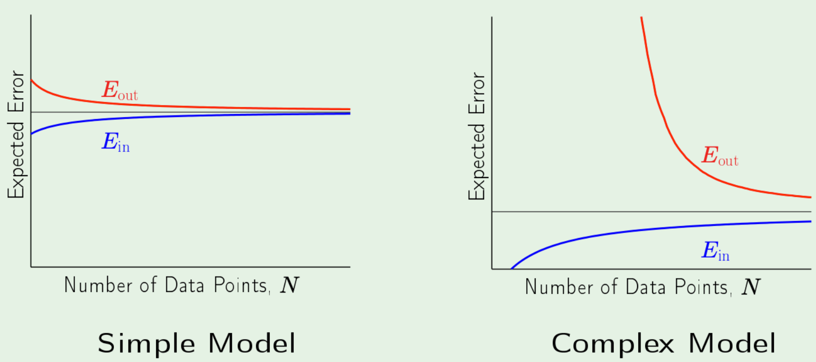


Machine Learning, Tom Mitchell, McGraw Hill, 1997

Overfitting

- Reasons
  - ✓ model is **too complex**
  - ✓ model is **fitting noise** present in the training data
  - ✓ training data is **not a representative sample** of the distribution
- How to prevent?
  - ✓ use **more training data**
  - ✓ use **fewer features**
  - ✓ **regularize** your model

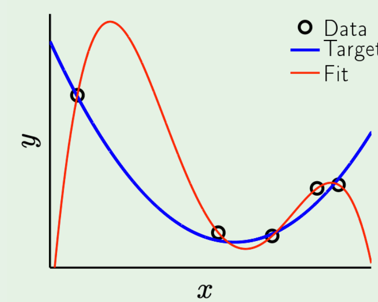
Number of data instances



https://work.caltech.edu/lectures.html

Restricting the model

- Imagine the target function below …
  - ✓ 5 noisy data points and a 4th order polynomial fit
  - ✓ what can you say about training error? test error?



https://work.caltech.edu/lectures.html

Restricting the model

https://work.caltech.edu/lectures.html

Model Evaluation

Confusion matrix (2 classes)

	POSITIVE (1)	NEGATIVE (0)
POSITIVE (1)	TP	FN
NEGATIVE (0)	FP	TN

Actual values? Predicted values?

Evaluation metrics (2 classes)

sensitivity, recall, hit rate, or true positive rate (TPR)

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{P}} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} = 1 - \mathrm{FNR}$$

specificity, selectivity or true negative rate (TNR)

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{N}} = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}} = 1 - \mathrm{FPR}$$

precision or positive predictive value (PPV)

$$\mathrm{PPV} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} = 1 - \mathrm{FDR}$$

negative predictive value (NPV)

$$\mathrm{NPV} = \frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FN}} = 1 - \mathrm{FOR}$$

miss rate or false negative rate (FNR)

$$\mathrm{FNR} = \frac{\mathrm{FN}}{\mathrm{P}} = \frac{\mathrm{FN}}{\mathrm{FN}+\mathrm{TP}} = 1 - \mathrm{TPR}$$

fall-out or false positive rate (FPR)

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{N}} = \frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}} = 1 - \mathrm{TNR}$$

https://en.wikipedia.org/wiki/Confusion_matrix

Evaluation metrics (2 classes)

accuracy (ACC)

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}$$

F1 score

is the harmonic mean of precision and sensitivity

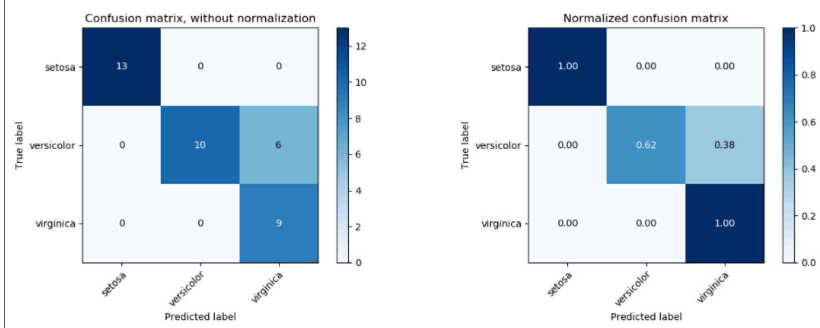
$$\mathrm{F}_1 = 2 \cdot \frac{\mathrm{PPV} \cdot \mathrm{TPR}}{\mathrm{PPV} + \mathrm{TPR}} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

Matthews correlation coefficient (MCC)

$$\mathrm{MCC} = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}$$

https://en.wikipedia.org/wiki/Confusion_matrix

Confusion matrix



https://scikit-learn.org/stable/auto_examples/model_selection/plot_confusion_matrix.html

Confusion matrix

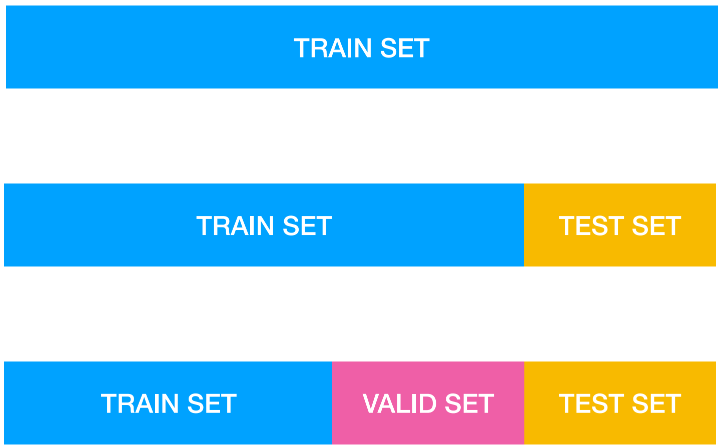
CIFAR-10 Confusion Matrix

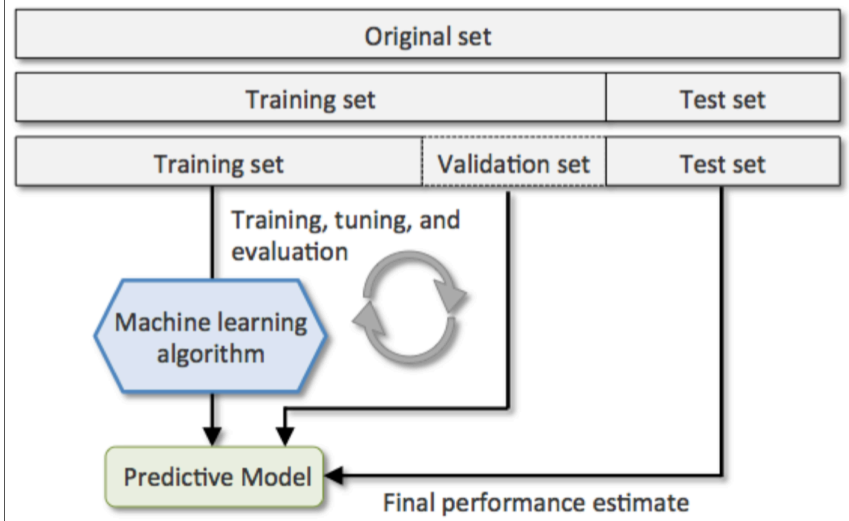
True Class \ Predicted Class	airplane	automobile	bird	cat	deer	dog	frog	horse	ship	truck		
airplane	923	4	21	8	4	1	5	5	23	6	92.3%	7.7%
automobile	5	972	2					1	5	15	97.2%	2.8%
bird	26	2	892	30	13	8	17	5	4	3	89.2%	10.8%
cat	12	4	32	826	24	48	30	12	5	7	82.6%	17.4%
deer	5	1	28	24	898	13	14	14	2	1	89.8%	10.2%
dog	7	2	28	111	18	801	13	17		3	80.1%	19.9%
frog	5		16	27	3	4	943	1	1		94.3%	5.7%
horse	9	1	14	13	22	17	3	915	2	4	91.5%	8.5%
ship	37	10	4	4		1	2	1	931	10	93.1%	6.9%
truck	20	39	3	3			2	1	9	923	92.3%	7.7%
	88.0%	93.9%	85.8%	79.0%	91.4%	89.7%	91.6%	94.1%	94.8%	95.0%		
	12.0%	6.1%	14.2%	21.0%	8.6%	10.3%	8.4%	5.9%	5.2%	5.0%		

https://www.mathworks.com/help/deeplearning/ref/confusionchart.html

Train, Validation, Test

Train, validation, and test sets





from INFO-4604: Applied Machine Learning, Fall 2017, Michael Paul, Univ. of Colorado

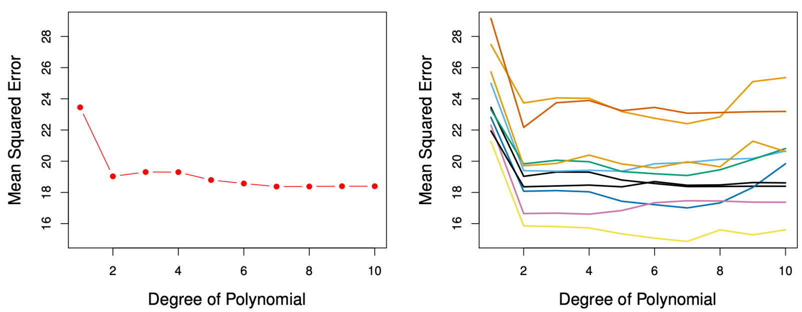
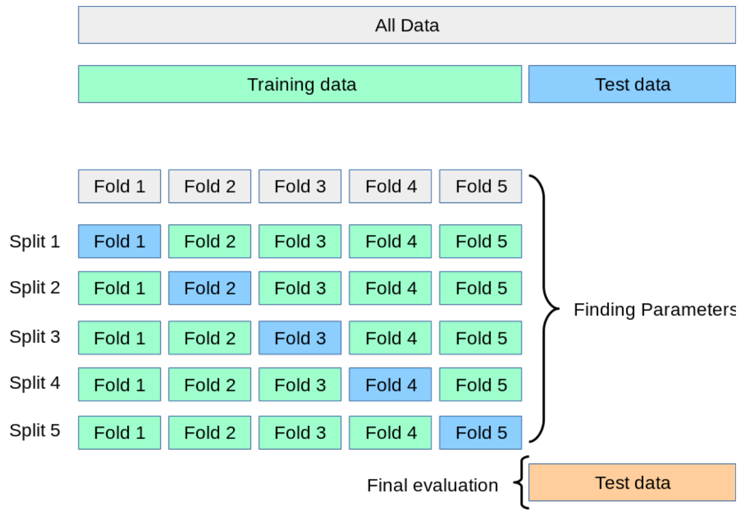


FIGURE 5.2. *The validation set approach was used on the* `Auto` *data set in order to estimate the test error that results from predicting* `mpg` *using polynomial functions of* `horsepower`. Left: *Validation error estimates for a single split into training and validation data sets.* Right: *The validation method was repeated ten times, each time using a different random split of the observations into a training set and a validation set. This illustrates the variability in the estimated test MSE that results from this approach.*

From An Introduction to Statistical Learning, 2nd Ed.

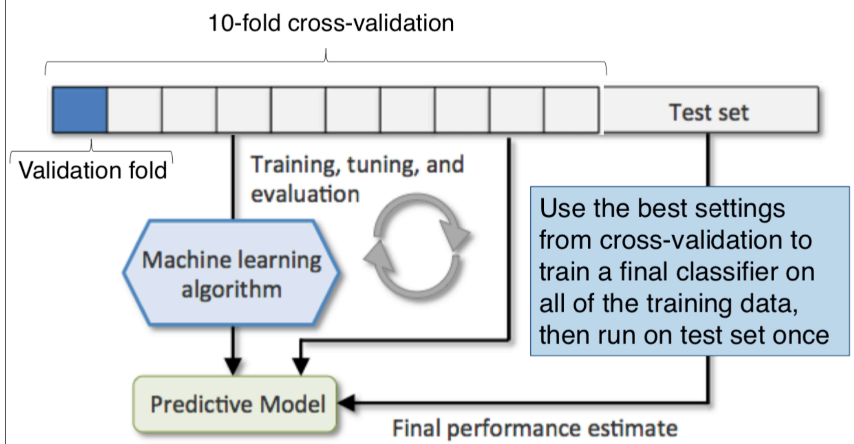
Cross Validation

What is k-fold Cross Validation?



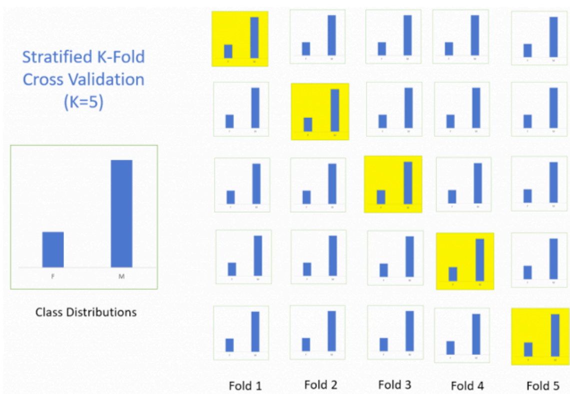
https://scikit-learn.org/stable/modules/cross_validation.html

Using Cross-Validation



from INFO-4604: Applied Machine Learning, Fall 2017, Michael Paul, Univ. of Colorado

Stratified cross validation

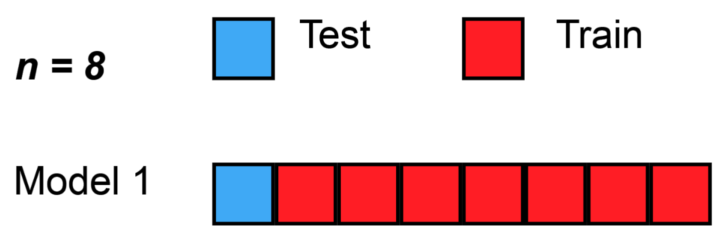


Stratified cross validation aims at having the same class distribution within each fold

https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

Leave-One-Out CV

- Special case of CV when **k = n**
- Can be expensive for large **n**



https://en.wikipedia.org/wiki/Cross-validation_(statistics)

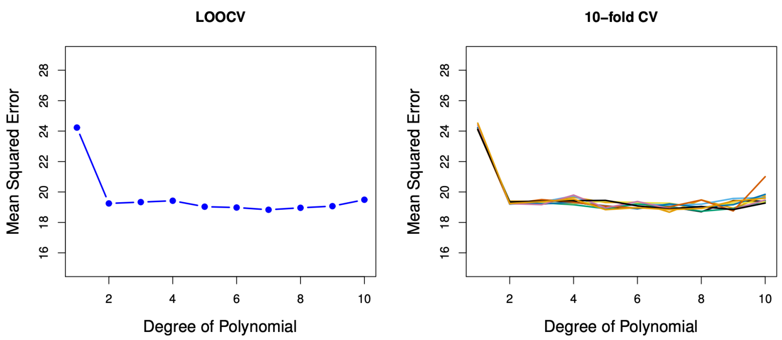


FIGURE 5.4. *Cross-validation was used on the* `Auto` *data set in order to estimate the test error that results from predicting* `mpg` *using polynomial functions of* `horsepower`. Left: *The LOOCV error curve.* Right: 10-*fold CV was run nine separate times, each with a different random split of the data into ten parts. The figure shows the nine slightly different CV error curves.*

From An Introduction to Statistical Learning, 2nd Ed.