Lecture 12: Evaluation Part 2

COMP90049

Semester 2, 2022

Joseph West, CIS

Acknowledgement: Jeremy Nicholson, Tim Baldwin & Karin Verspoor

Roadmap

So far

- Supervised learning algorithms: Naive Bayes, Logistic Regression, Multi Layer Perceptron, Decision Trees
- Evaluation Part 1: Assess the effectiveness of the classifier

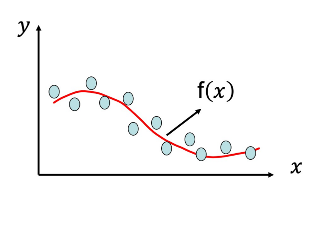
Today

- What's the problem of the model? Underfitting (high model bias) and overfitting (high model variance)
- How to know which problem the model suffers from? Learning curve
- How to correct the problems? Remedies for underfitting and overfitting
- Evaluation bias and variance

Underfitting and Overfitting

Given a training dataset $D = \{x_i, y_i\}$, $i = 1 \ldots n$ and $y \in \mathbb{R}$:

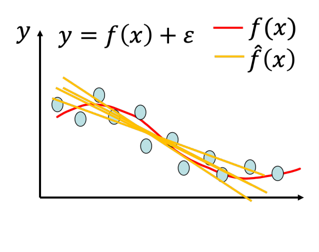
$$y = f(x) + \epsilon$$



- $f(\cdot)$: true function to generate data
- $\epsilon \in \mathcal{N}(0, \sigma)$: data noise

Generalization Error II

$$Err(x) = E\left[(y - \hat{f}(x))^2\right]$$



- $\hat{f}(x)$: estimation of $f(x)$
- Use multiple models (trained on different training sets) to remove data dependency
- E: expectation (average) operator over all possible training sets

Generalization Error III

- The generalization error can be decomposed to:

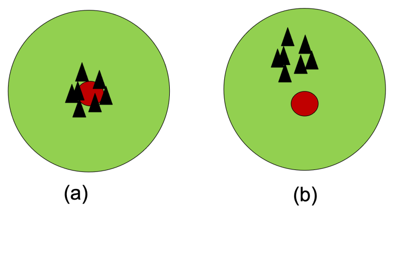
$$Err(x) = \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\left(\hat{f}(x) - E[\hat{f}(x)]\right)^2\right] + \sigma^2$$

- Or simply written as:

$$Err(x) = \text{Model Bias}^2 + \text{Model Variance} + \text{Irreducible Error}$$

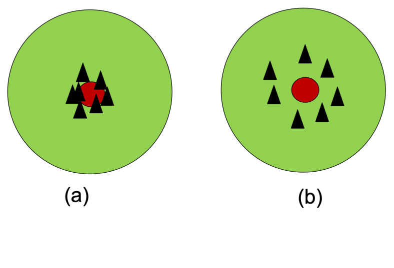
Bias Example

Which one has low bias?



Variance Example

Which one has low variance?



Underfitting I

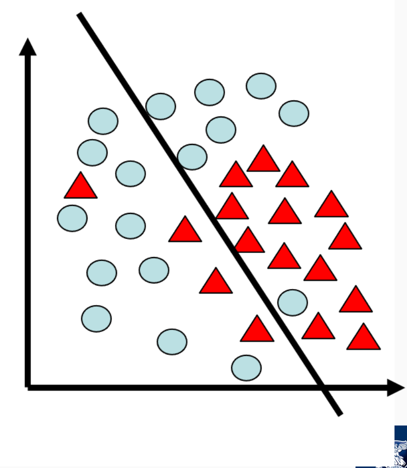
$$Err(x) = \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\left(\hat{f}(x) - E[\hat{f}(x)]\right)^2\right] + \sigma^2$$

Lazy model $\hat{f}(x) = c$: Extreme underfitting

- Model Variance: zero,
- Model Bias: large

Underfitting II

- does not fit the data
- Bad performance on training data
- Does not generalize to new data.



Overfitting I

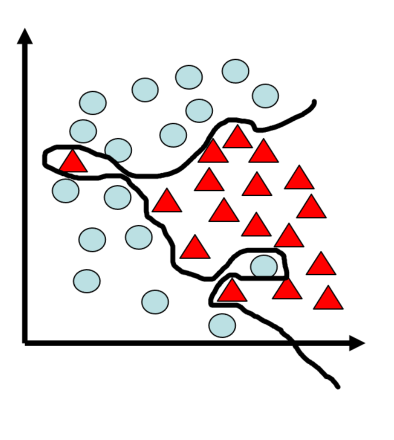
$$Err(x) = \left(E[\hat{f}(x)] - f(x)\right)^2 + E\left[\left(\hat{f}(x) - E[\hat{f}(x)]\right)^2\right] + \sigma^2$$

Hard-working model $\hat{f}(x) = y = f(x) + \epsilon$: Extreme Overfitting

- Bias: zero,
- Variance: large

Overfitting II

- Fit the training data too well
- Good performance on training data
- Unreliable Generalization.

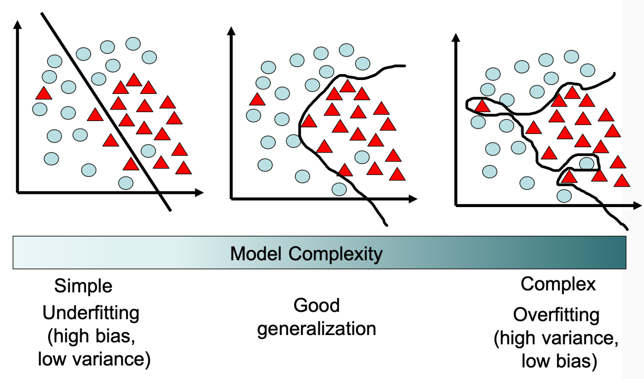


Overfitting III

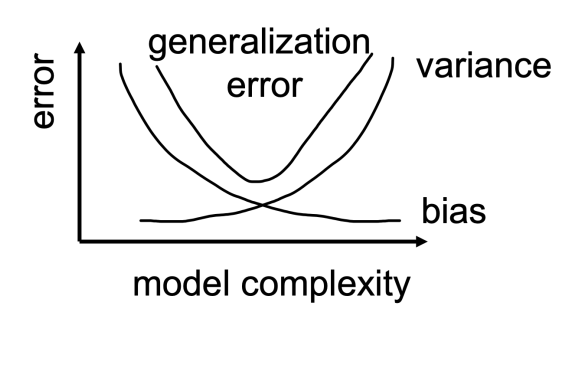
Which baseline has low variance?

- (a) Weighted random classifier
- (b) 0-R (majority classifier)

Underfitting vs Overfitting



Can we have a model with minimum bias and variance?

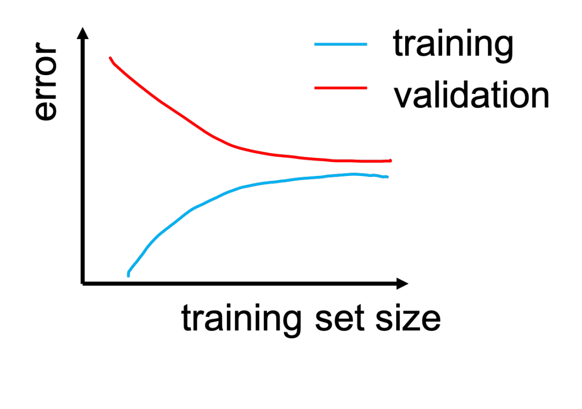

generalization
error
variance
error
bias
model complexity

Learning Curve

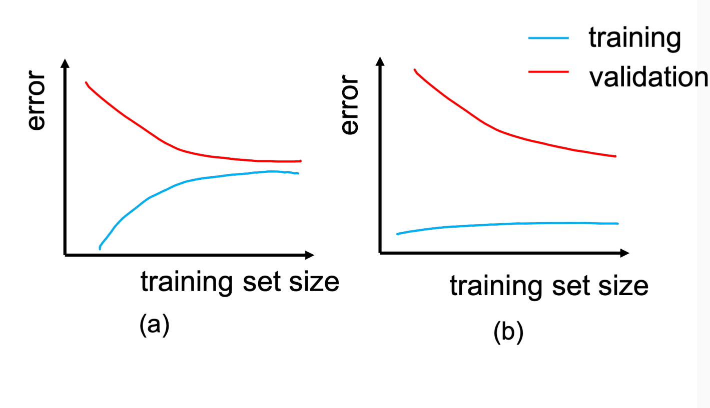
Learning Curve

Learning curve: plot of learning performance over increasing size of training dataset.

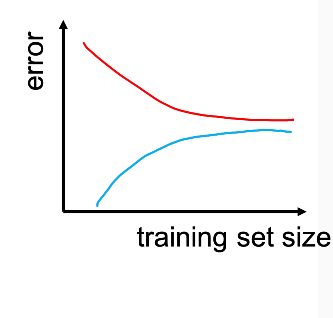
- x-axis: increasing number of training examples
- y-axis: scores like accuracy, error...



Which is underfitting ? Which is overfitting?

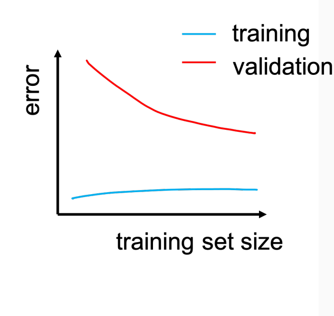


Underfitting vs Overfitting



Underfitting

- high training and validation error
- increasing data does not help



Overfitting

- low training error, high validation error
- increasing data can help

How to Plot Learning Curve

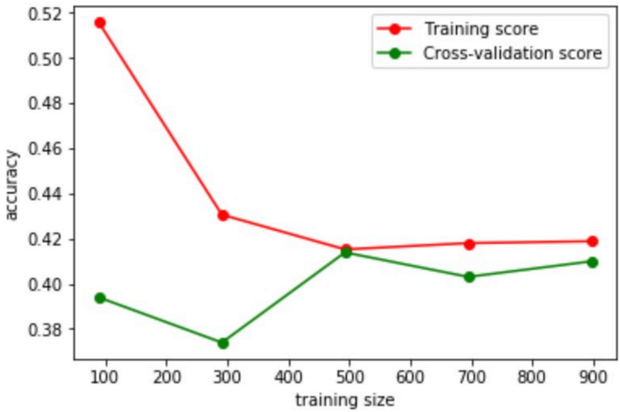
Learning curve:

- Choose various split sizes, and calculate effectiveness
  - For example: 90-10, 80-20, 70-30, 46-40, 50-50, 40-60, 30-70, 20-80, 10-90 (9 points)
  - Might need to average multiple runs per split size
- Plot % of training data vs training/test Accuracy (or other metric)

Example Code

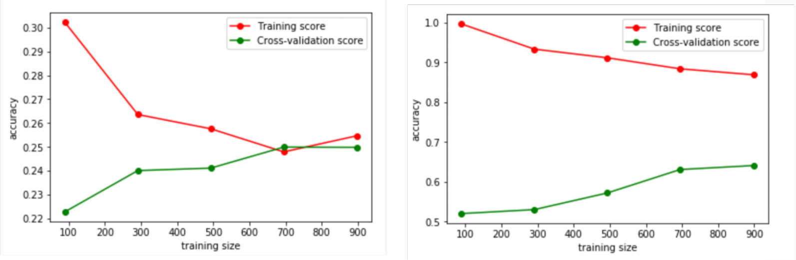
```
#https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.learning_curve.html#
#https://scikit-learn.org/stable/auto_examples/model_selection/plot_learning_curve.html#sphx-gl
from sklearn.model_selection import learning_curve
import matplotlib.pyplot as plt
from sklearn import tree
from sklearn.model_selection import StratifiedKFold

estimator = tree.DecisionTreeClassifier(max_depth=2)
train_sizes, train_scores, valid_scores= \
        learning_curve(estimator, X_train, y_train, scoring='accuracy',cv=StratifiedKFold(10),
                       train_sizes=np.linspace(.1, 1.0, 5))
plt.figure()
plt.xlabel("training size")
plt.ylabel("accuracy")
plt.plot(train_sizes,  np.mean(train_scores, axis=1), 'o-', color="r",
                 label="Training score")
plt.plot(train_sizes,  np.mean(valid_scores, axis=1), 'o-', color="g",
                 label="Cross-validation score")
plt.legend(loc="best")
plt.show()
```

Learning Curve Example I



Learning Curve Example II

Replacing the model with decision stump - (tree depth of 1) ?



(a) (b)

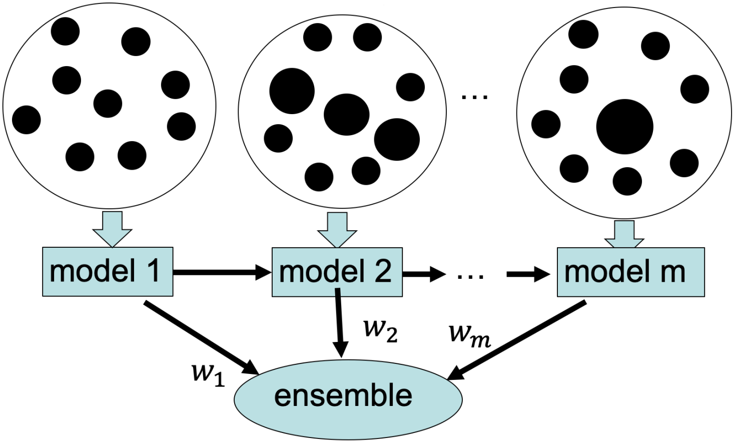
Remedy for Underfitting and Overfitting

Underfitting Remedy

- Use more complex model (e.g. use nonlinear models)
- Add features
- Boosting

Boosting

- training data: different weights (probabilities to be selected)
- Use multiple weak models → a stronger model; reduces bias (improves performance)

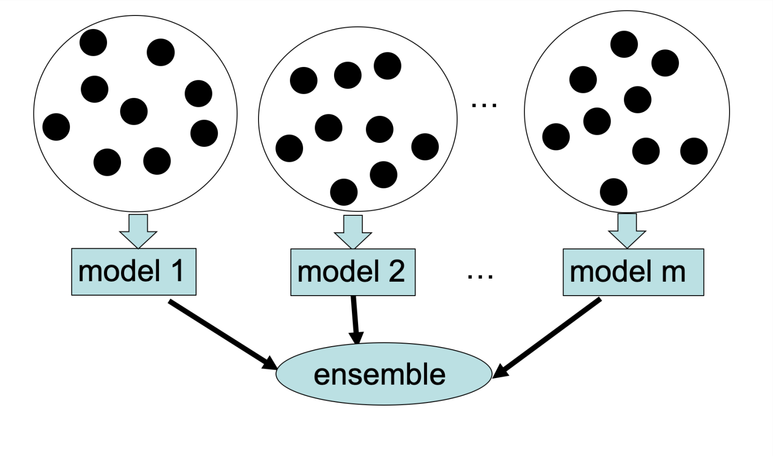


Overfitting Remedy

- Add more training data
- Reduce features
- Reduce model complexity – complex models are prone to high variance
- Bagging

Bagging

- Construct new datasets: randomly select the training data with replacement
- Combining multiple models$\rightarrow$ predictions are more stable; reduces variance of individual model.



Evaluation Bias and Variance

Evaluation Bias and Variance I

- We want to know the “true” error rate of a classifier, but we only have an estimate of the error rate, subject to some particular set of evaluation instances
    - **High evaluation Bias**: Our estimate of the effectiveness of a model is systematically too high/low
    - **High evaluation Variance**: Our estimate of the effectiveness of a model changes a lot, as we alter the instances in the evaluation set

Evaluation Bias and Variance II

How do we control bias and variance in evaluation?

- Holdout partition size
  - More training data: more evaluation variance
  - Less training (more test) data: less evaluation variance
- Repeated random subsampling and K-fold Cross-Validation
  - Less variance than Holdout
- Stratification: less evaluation bias
- Leave-one-out Cross-Validation
  - No sampling bias, lowest bias/variance in general

Summary

Summary

- How are underfitting and overfitting different?
- How are model bias and variance different?
- how to diagnose underfitting and overfitting using learning curve?
- How do we try to control for model bias and variance
- What is evaluation bias and variance?
- How do we try to control for bias and variance in evaluation?

References

- Sammut, Claude; Webb, Geoffrey I., eds. (2011). Bias Variance Decomposition. Encyclopedia of Machine Learning. Springer. pp. 100–101.
- Luxburg, Ulrike V.; Schölkopf, B. (2011). Statistical learning theory: Models, concepts, and results. Handbook of the History of Logic. 10: Section 2.4.
- Vijayakumar, Sethu (2007). The Bias–Variance Tradeoff. University of Edinburgh. Retrieved 19 August 2014.
- James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 112). New York: springer. Chapter 2.
- Jeremy Nicholson & Tim Baldwin & Karin Verspoor: Machine Learning