Project Discussion

CS3244: Machine Learning

03 March 2022

CS3244 Project

Assessment Components

1. Application design & problem formulation (8%)
2. Model design & construction (6%)
3. Evaluation (6%)
4. Novelty (4%)
5. Instructions (1%)

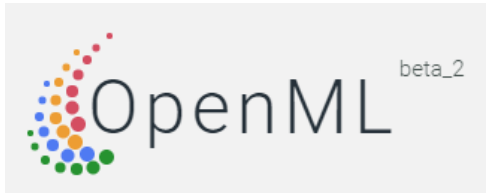
Content Overview

1. The Application
2. The Model
3. The Evaluation

Designing a Machine Learning (ML) Application

The Machine Learning Application

- Is it just about a dataset?



Designing Applications

- SDLC
  - Planning → Analysis → Design → Implementation → Maintenance → Planning → ...

- Planning and Analysis
  - Is there a need for ML in a particular system?
    - User ML needs / needs analysis
    - Use cases incorporating ML solutions

Main Issues

- Assume supervised learning

1. What objectives?
   - Model accuracy?
   - Performance measures
     - Quantifying the objectives

2. What data?
   - Use existing dataset or collecting data?
   - What do I know about the domain?
     - Features?
     - Hypothesis representation/space?

Objectives Apart from Accuracy

- What are other possible performance measures?

Gathering Data

- What do you intend to do about data?

Constructing a Good Predictor

Consistency with Training Data Versus Generalisation

- Consistency with training data is just the beginning ...

Real problems → massive instance spaces

UCI Machine Learning Repository
Center for Machine Learning and Intelligent Systems

Mushroom Data Set

Data Set Characteristics:	Multivariate	**Number of Instances:**	8124
Attribute Characteristics:	Categorical	**Number of Attributes:**	22
Associated Tasks:	Classification	**Missing Values?**	Yes

https://archive.ics.uci.edu/ml/datasets/Mushroom

Size of instance space:
1,638,333,457,367,040

... *generalisation* is more important (usually)

Consequence of Generalisation Objective

- Data is not enough

Example

D: 1,000,000 instances; 100 Boolean variables

$2^{100} - 10^6$ instances unlabelled

If all target functions are equally likely, any hypothesis cannot do better than random guessing...

No-Free-Lunch Theorems

Thankfully, real-world problems not drawn uniformly from the set of all possible function

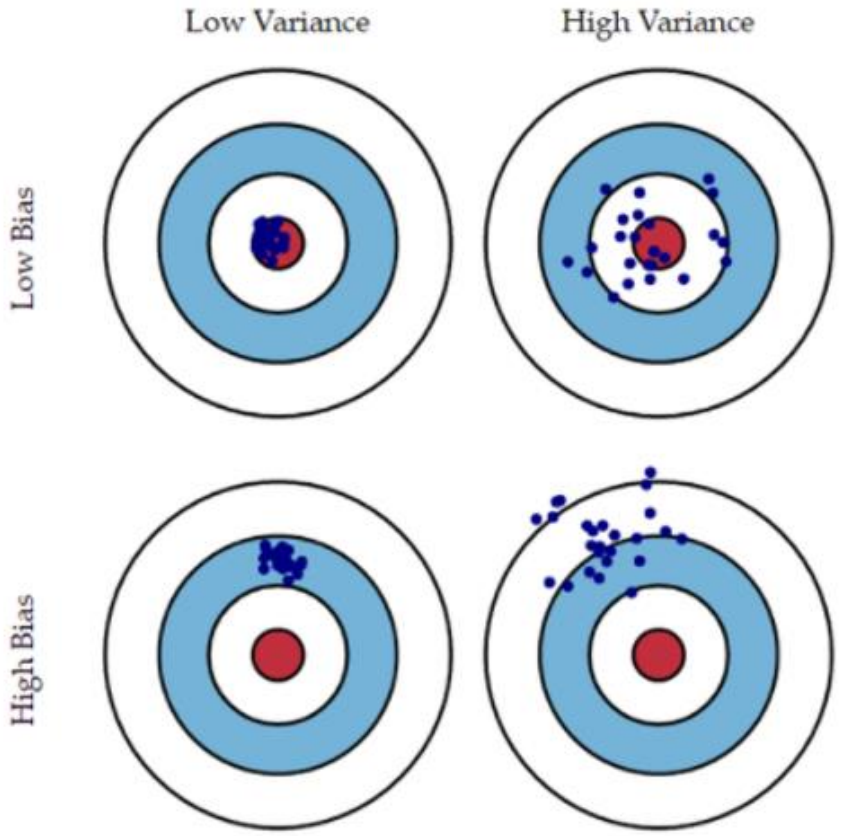
Picking the right Inductive Bias is essential

Inductive bias:
- Hypothesis Representation
- Hypothesis Preference

Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural computation*, *8*(7), 1341-1390

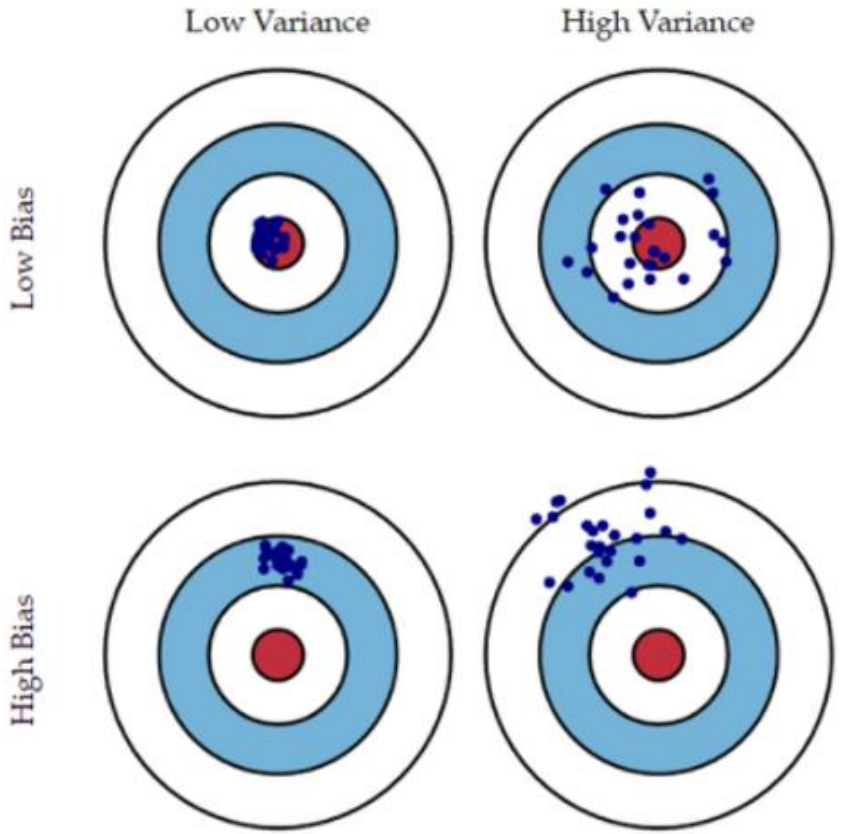
Bias-Variance Decomposition

- Generalisation error may be divided
  - Noise
    - Error inherent within the data
    - Typically cannot reduce this
  - Bias
    - Error from assumptions about target function
      - Appropriateness of hypothesis representation
      - Relevance of features / sufficient features
  - Variance
    - Error from sensitivity to small fluctuations in training data
      - More general hypotheses have lower variance



Overfitting & Underfitting

- High bias ⇒ underfitting
  - Model not expressive enough or not appropriately expressive to capture c
  - Higher training error
- High variance ⇒ overfitting
  - Model too expressive and sensitive to smaller changes in training sample
  - Requires more data to converge
  - Lower training error, higher testing error
- Examples
  - Decision trees
    - Larger/deeper tree ⇒ lower bias; higher variance
  - Neural Networks
    - More hidden units ⇒ lower bias; higher variance



Simple Ideas to Improve Generalisation Performance

Feature Selection

- Determine the “right” attributes to use
  - Remove redundant / irrelevant attributes
  - Filter approach – use a heuristic
    - Pearson correlation coefficient
    - Information gain
  - Wrapper approach
    - ML algorithm used to assess value of attribute sets
  - Embedded approach
    - Feature selection is part of the ML algorithm

Validation Using Cross-Validation

- k-Fold Cross-validation
- Divide training set, S, into k-folds, s_1, ..., s_k

  For each fold s_i

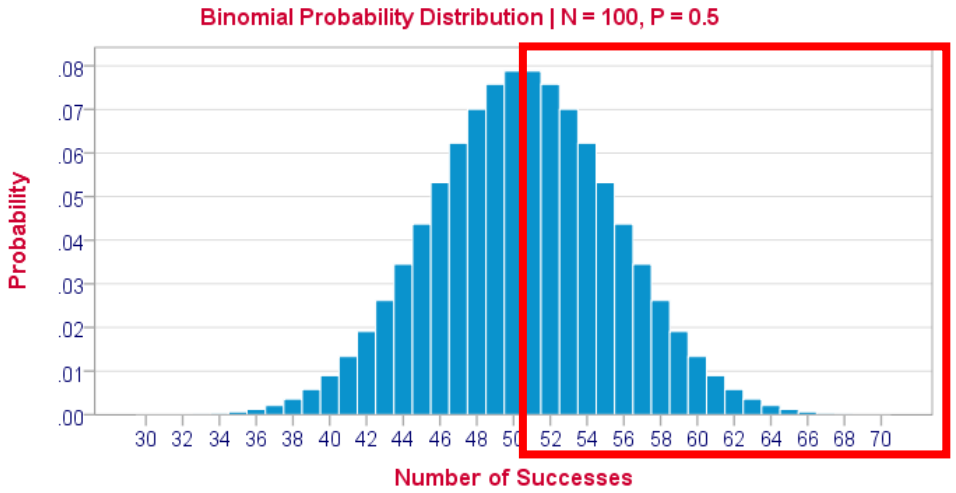
  Train model using $S \setminus s_i$

  Test model using s_i

  Take mean performance
- Wrapper-based approach
  - Selecting hyperparameters
  - Selecting attributes

Ensembles

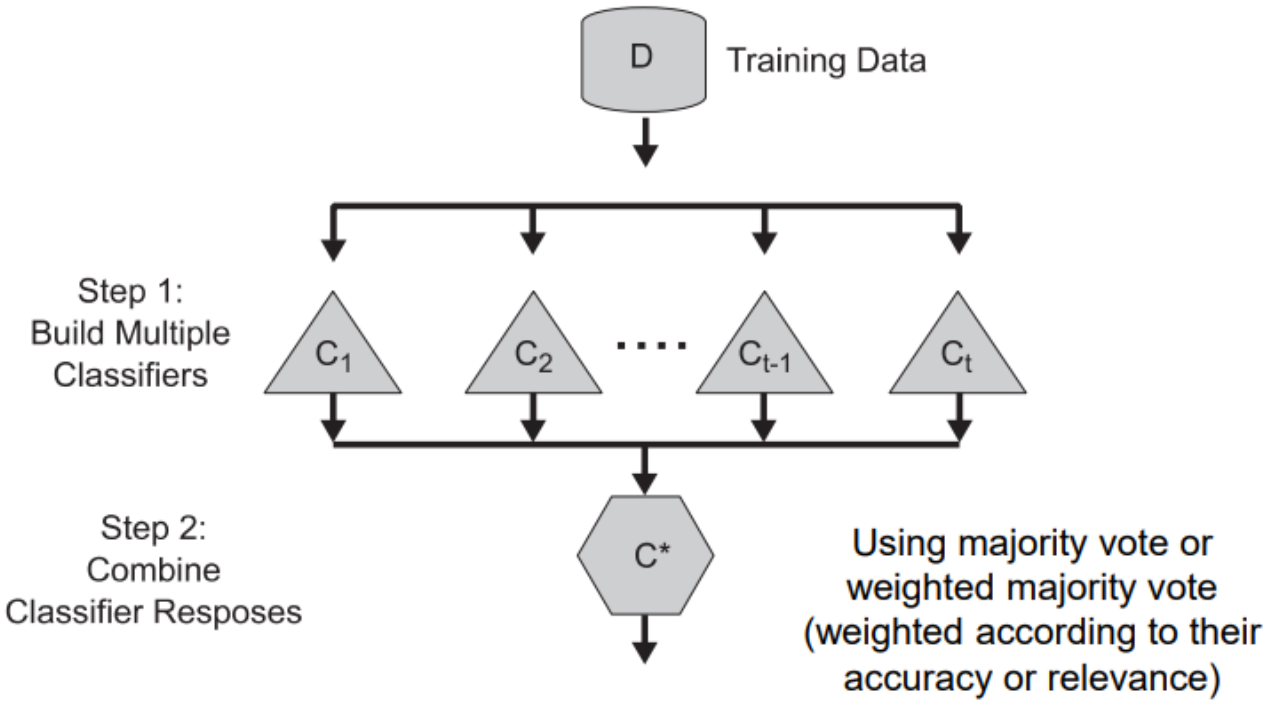
- General idea
  - Aggregate predictions of multiple hypotheses to generate an overall classification that is more accurate
- General motivation
  - Assume k independent (i.e., uncorrelated) hypotheses
  - Assume generalisation performance > 0.5



sum binomial terms with 51+ successes

much higher success as *p* tends to 1

Ensemble Framework



Example:
Random Forest

Evaluation

Model Evaluation

- How do you know you have succeeded?
- Determine a benchmark
  - Competing model
  - Threshold
- Form hypothesis test to see your model is significantly better than the benchmark
  - m × k-Fold Cross-Validation Performance
  - Each value is a mean (central limit theorem applies)
  - Apply t-test

Experimental Setup for Empirical Evaluation

- Example Walkthrough

Summary

- Determine appropriate user ML need
- Determine the important performance measures
  - For prediction, prioritise generalisation performance
- Determine your sources of data
- Apply domain knowledge and feature selection
- Consider performing validation to choose hyperparameters
- Consider ensemble methods
- Evaluate model against a benchmark via a valid hypothesis test

Questions?