COMS30035, Machine learning: Combining Models 1, Model Selection and Averaging

Edwin Simpson

edwin.simpson@bristol.ac.uk

Department of Computer Science, SCEEM
University of Bristol

October 30, 2020

Agenda

- Model Selection
- Model Averaging
- Ensembles: Bagging and Boosting
- Tree-based Models
- Conditional Mixture Models
- Crowdsourcing

Textbook

We will follow Chapter 14 of the Bishop book: Bishop, C. M., Pattern recognition and machine learning (2006). Available for free here.

The Model Selection Problem

- Select a model h from a set of possible models, indexed $1, ..., H$,
- The models in H can differ in various ways, such as:
  - The structure of the model, e.g., differences in graphical models;
  - The choice of prior distribution $p(\theta)$ over model parameters;
  - The learning algorithm used, e.g., EM, MCMC, backpropagation, etc.; algorithm;
  - The features of the data used as inputs to the model.
  - The examples in the dataset the model is trained on.
  - Random initialisation, e.g., of parameters for EM or gradient descent for neural networks

Hyperparameters

- It's useful to characterise modelling decisions as a set of parameters called *hyperparameters*
- All parameters that are fixed before training are hyperparameters.
- Typical hyperparameters include:
  - Parameters of the prior distribution
  - Parameters of the learning algorithm, e.g., the learning rate for a neural network
  - Parameters of feature extractors.

Model Selection on a Validation Set

- Train the model with different combinations of features, hyperparameter values and random initialisations
- Choose the model h that maximises performance on a validation set.
  - As seen in week 1.
  - Validation set is obtained by setting aside some part of the labelled set.
  - Can't tune on the training set as it would lead to overfitting
- Advantage: optimises a performance metric we really care about.
- Disadvantage: we didn't use all the data in training;
- Disadvantage: if the validation set is small, we might choose the wrong model!

A Probabilistic View of Model Selection

- Suppose we have the following machine learning task:
  - Latent variables to predict (e.g., class labels in the test set): $\boldsymbol{z}$;
  - Training data: $\boldsymbol{X}$;
  - Model: h.
- We obtain the prediction of $\boldsymbol{z}$ from a chosen model h as follows:

$$p(\boldsymbol{z}|\boldsymbol{X}) \approx p(\boldsymbol{z}|\boldsymbol{X}, h) \tag{1}$$

Bayesian Model Selection

- How can we choose h in $p(\boldsymbol{z}|\boldsymbol{X}, h)$?
- Choose $h = h^*$ to *maximise* the marginal likelihood of the data:

$$h^* = \underset{h}{\operatorname{argmax}} p(\boldsymbol{X}|h) = \underset{h}{\operatorname{argmax}} \int p(\boldsymbol{X}|\boldsymbol{\theta}, h) p(\boldsymbol{\theta}|h) d\boldsymbol{\theta} \tag{2}$$

- Similar to maximum likelihood estimation, which we used before to optimise parameters $\boldsymbol{\theta}$.
  - Here, we use a Bayesian approach and integrate out (marginalise) $\boldsymbol{\theta}$.
  - Relies on finding a single, good model given our training set.

Bayesian Model Averaging (BMA)

- Even after computing marginal likelihood, we may be uncertain about which model h is correct
- We can express this by assigning a probability to each model given the training data, $p(h|\boldsymbol{X})$.

Bayesian Model Averaging (BMA)

- Rather than choosing a single model, we can now take an expectation.
- Our predictions now come from a *weighted sum* over models, where $p(h|\boldsymbol{X})$ are weights :

$$p(\boldsymbol{z}|\boldsymbol{X}) = \sum_{h=1}^{H} p(\boldsymbol{z}|\boldsymbol{X}, h)p(h|\boldsymbol{X}) \tag{3}$$

Bayesian Model Averaging (BMA)

- Apply Bayes' rule to estimate the weights:

$$p(h|\boldsymbol{X}) = \frac{p(\boldsymbol{X}|h)p(h)}{\sum_{h'=1}^{H} p(\boldsymbol{X}|h)p(h')} \tag{4}$$

- What happens as we increase the amount of data in $\boldsymbol{X}$? $p(h|\boldsymbol{X})$ becomes more focussed on one model.
- So BMA is soft model selection, it does not *combine* models to make a more powerful model.

Mixture of Experts

- Similar principle to BMA, except we soft-select a different model for each data point z_i that we wish to predict.
- Motivation: rather than design a single, complex model, each part of the input space is dealt with by a specialised expert model.
- Think of medical diagnosis: based on the patient's symptoms, a GP refers the patient to a specialist. If they are unsure what is causing the symptoms, they may send the patient to multiple specialists for examination.

Mixture of Experts

- Task: predicting z_i, e.g., a class label for data point i
- Model weights therefore depend on the input feature vector $\boldsymbol{x}_i$ for that data point.

$$p(z_i|\boldsymbol{X}, \boldsymbol{x}_i) = \sum_{h=1}^{H} p(z_i|\boldsymbol{x}_i, \boldsymbol{X}, h)p(h|\boldsymbol{X}, \boldsymbol{x}_i) \tag{5}$$

- This is the same idea as a mixture model, where one component is responsible for generating each data point.
- The weights can also be learned using EM.

Now do the quiz!

Please do the quiz for this lecture on Blackboard.