Foundations of Machine Learning
African Masters in Machine Intelligence

Model Selection

Marc Deisenroth

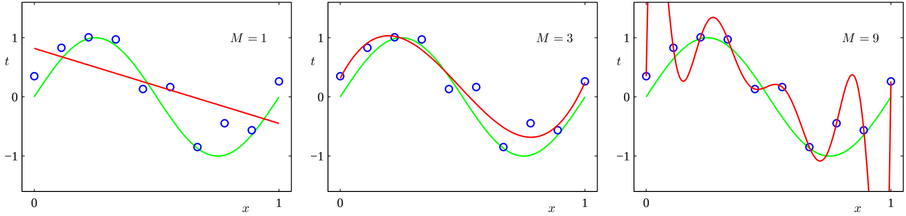
Quantum Leap Africa
African Institute for Mathematical Sciences, Rwanda

Department of Computing
Imperial College London

@mpd37
mdeisenroth@aimsammi.org

October 10, 2018

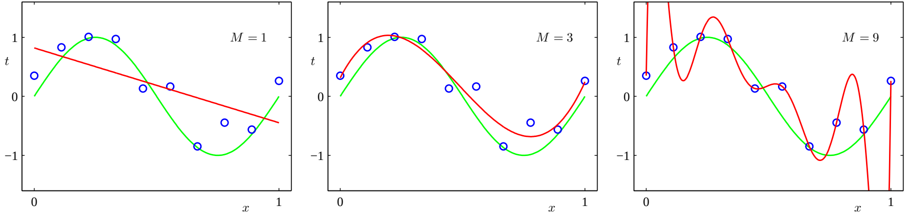
Model Selection



From PRML (Bishop, 2006)

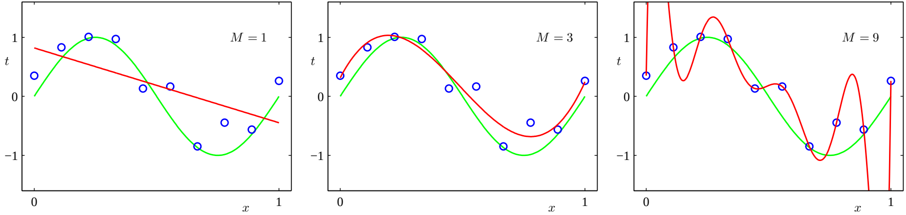
Sometimes, we have to make high-level decisions about the model we want to use:

- Number of components in a mixture model
- Network architecture of (deep) neural networks
- Type of kernel in a support vector machine
- Degree of a polynomial in a regression problem



From PRML (Bishop, 2006)

- For each high-level choice, we get a different set of parameters
- Rule of thumb: More parameters = more flexible model



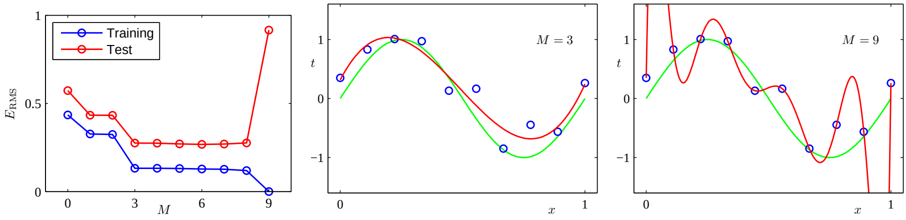
From PRML (Bishop, 2006)

- For each high-level choice, we get a different set of parameters
- Rule of thumb: More parameters = more flexible model

Problem

- At training time, we can only use the training data to evaluate the performance of the model
- We are generally interested in the test performance, not so much in the training performance

Training vs Test Error

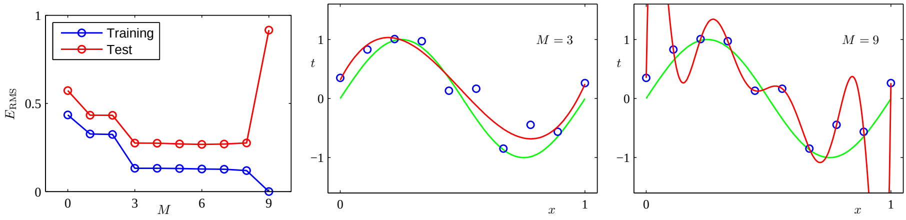


From PRML (Bishop, 2006)

General problem:

- Model fits training data perfectly, but may not do well on test data ⏩ Overfitting (especially with MLE)

Training vs Test Error

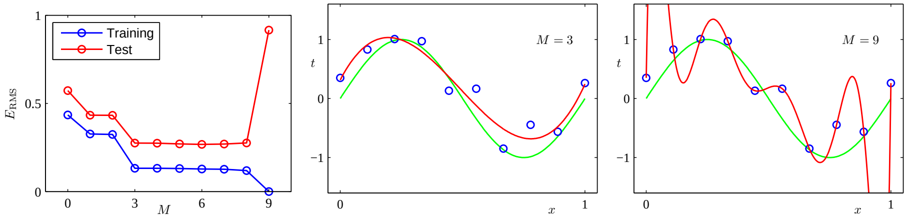


From PRML (Bishop, 2006)

General problem:

- Model fits training data perfectly, but may not do well on test data ⏩ Overfitting (especially with MLE)
- Training performance $\neq$ test performance, but we are mostly interested in test performance

Training vs Test Error



From PRML (Bishop, 2006)

General problem:

- Model fits training data perfectly, but may not do well on test data ⏩ Overfitting (especially with MLE)
- Training performance $\neq$ test performance, but we are mostly interested in test performance
- Need mechanisms for assessing how a model generalizes to unseen test data ⏩ Model selection

Training vs Test Error (2)

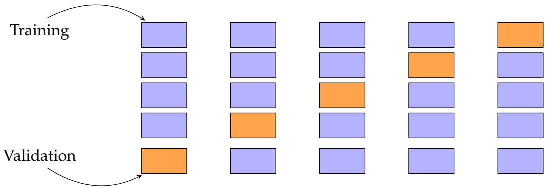
Model	L2	Train Accuracy	Test Accuracy
1 layer MLP		100.0	50.51
	✓	99.80	50.39
3 layer MLP		100.0	52.39
	✓	100.0	53.35
Alexnet (CNN)		100.0	76.07
	✓	100.0	77.36
Inception (CNN++)		100.0	85.75
	✓	100.0	86.03

Zhang, Chiyuan; Bengio, Samy; Hardt, Moritz; Recht, Benjamin; Vinyals, Oriol. "Understanding deep learning requires rethinking generalization", ICLR 2017

From Y. Dauphin's lecture at DL Indaba 2017

- What is suspicious here?

Cross Validation



- Heuristic to estimate the generalization performance of a model
- Partition your training data into K subsets
- Train the model on $K - 1$ subsets
- Evaluate the model on the other subset

Cross-Validation (2)

- Cross-validation effectively computes an empirical generalization error R on validation set $\mathcal{V}$:

$$\mathbb{E}_{\mathcal{V}}[R(f, \mathcal{V})] \approx \frac{1}{K} \sum_{k=1}^{K} R(f, \mathcal{V}^{(k)})$$

- R is a loss function (e.g., RMSE or NLL)
- To reduce variability, multiple rounds of cross-validation are performed using different partitions, and the validation results are averaged over the rounds.

Cross-Validation (2)

- Cross-validation effectively computes an empirical generalization error R on validation set $\mathcal{V}$:

$$\mathbb{E}_{\mathcal{V}}[R(f,\mathcal{V})] \approx \frac{1}{K}\sum_{k=1}^{K} R(f,\mathcal{V}^{(k)})$$

- R is a loss function (e.g., RMSE or NLL)
- To reduce variability, multiple rounds of cross-validation are performed using different partitions, and the validation results are averaged over the rounds.
- Train many models, compare test error

Cross-Validation (2)

- Cross-validation effectively computes an empirical generalization error R on validation set $\mathcal{V}$:

$$\mathbb{E}_{\mathcal{V}}[R(f, \mathcal{V})] \approx \frac{1}{K} \sum_{k=1}^{K} R(f, \mathcal{V}^{(k)})$$

- R is a loss function (e.g., RMSE or NLL)
- To reduce variability, multiple rounds of cross-validation are performed using different partitions, and the validation results are averaged over the rounds.
- Train many models, compare test error

Number of training runs increases with the number of partitions
Trivial to parallelize

Information Criteria

- Add penalty term to MLE to compensate for the overfitting of more complex models (with lots of parameters)

Information Criteria

- Add penalty term to MLE to compensate for the overfitting of more complex models (with lots of parameters)
- Maximize Akaike Information Criterion (Akaike 1974):

$$\ln p(\boldsymbol{x}|\boldsymbol{\theta}_{\mathrm{ML}}) - M$$

where M is the number of model parameters

Information Criteria

- Add penalty term to MLE to compensate for the overfitting of more complex models (with lots of parameters)
- Maximize Akaike Information Criterion (Akaike 1974):

$$\ln p(\boldsymbol{x}|\boldsymbol{\theta}_{\mathrm{ML}}) - M$$

  where M is the number of model parameters
- AIC estimates the relative information lost by a given model

Information Criteria

- Add penalty term to MLE to compensate for the overfitting of more complex models (with lots of parameters)
- Maximize Akaike Information Criterion (Akaike 1974):

$$\ln p(\boldsymbol{x}|\boldsymbol{\theta}_{\mathrm{ML}}) - M$$

  where M is the number of model parameters
- AIC estimates the relative information lost by a given model
- Bayesian Information Criterion/MDL (Schwarz 1978) (for exponential family distributions):

$$\ln p(\boldsymbol{x}) = \ln \int p(\boldsymbol{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \approx \ln p(\boldsymbol{x}|\boldsymbol{\theta}_{\mathrm{ML}}) - \frac{1}{2}M \ln N$$

  where N is the number of data points and M is the number of parameters.

Information Criteria

- Add penalty term to MLE to compensate for the overfitting of more complex models (with lots of parameters)
- Maximize Akaike Information Criterion (Akaike 1974):

$$\ln p(\boldsymbol{x}|\boldsymbol{\theta}_{\mathrm{ML}}) - M$$

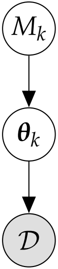
  where M is the number of model parameters
- AIC estimates the relative information lost by a given model
- Bayesian Information Criterion/MDL (Schwarz 1978) (for exponential family distributions):

$$\ln p(\boldsymbol{x}) = \ln \int p(\boldsymbol{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \approx \ln p(\boldsymbol{x}|\boldsymbol{\theta}_{\mathrm{ML}}) - \frac{1}{2}M \ln N$$

  where N is the number of data points and M is the number of parameters.
- BIC penalizes model complexity more heavily than AIC.

Bayesian Model Comparison

- Place a prior $p(M)$ on the class of models

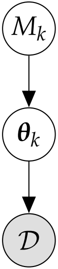


Bayesian Model Comparison

- Place a prior $p(M)$ on the class of models
- Given a training set $\mathcal{D}$, we compute the posterior distribution over models as

$$p(M_i|\mathcal{D}) \;\propto\; p(M_i)p(\mathcal{D}|M_i)$$

which allows us to express a preference for different models



Bayesian Model Comparison

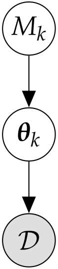
- Place a prior $p(M)$ on the class of models
- Given a training set $\mathcal{D}$, we compute the posterior distribution over models as

$$p(M_i|\mathcal{D}) \propto p(M_i)p(\mathcal{D}|M_i)$$

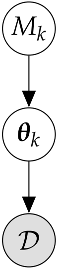
which allows us to express a preference for different models

- Model evidence (marginal likelihood):

$$p(\mathcal{D}|M_i) = \int p(\mathcal{D}|\boldsymbol{\theta}_{M_i})p(\boldsymbol{\theta}_{M_i}|M_i)d\boldsymbol{\theta}_{M_i}$$



Bayesian Model Comparison



- Place a prior $p(M)$ on the class of models
- Given a training set $\mathcal{D}$, we compute the posterior distribution over models as

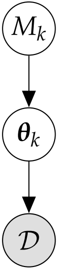
$$p(M_i|\mathcal{D}) \propto p(M_i)p(\mathcal{D}|M_i)$$

  which allows us to express a preference for different models
- Model evidence (marginal likelihood):

$$p(\mathcal{D}|M_i) = \int p(\mathcal{D}|\boldsymbol{\theta}_{M_i})p(\boldsymbol{\theta}_{M_i}|M_i)d\boldsymbol{\theta}_{M_i}$$

- Bayes factor for comparing two models: $p(\mathcal{D}|M_1)/p(\mathcal{D}|M_2)$

Bayesian Model Comparison



- Place a prior $p(M)$ on the class of models
- Given a training set $\mathcal{D}$, we compute the posterior distribution over models as

$$p(M_i|\mathcal{D}) \;\propto\; p(M_i)p(\mathcal{D}|M_i)$$

  which allows us to express a preference for different models
- Model evidence (marginal likelihood):

$$p(\mathcal{D}|M_i) = \int p(\mathcal{D}|\boldsymbol{\theta}_{M_i})p(\boldsymbol{\theta}_{M_i}|M_i)d\boldsymbol{\theta}_{M_i}$$

- Bayes factor for comparing two models: $p(\mathcal{D}|M_1)/p(\mathcal{D}|M_2)$
- Integral often intractable

Bayesian Model Averaging

- Place a prior $p(M)$ on the class of models
- Instead of selecting the "best" model, integrate out the corresponding model parameters $\boldsymbol{\theta}_M$ and average over all models $M_i, i = 1, \ldots, L$

$$p(\mathcal{D}) = \sum_{i=1}^{L} p(M_i) \underbrace{\int p(\mathcal{D}|\boldsymbol{\theta}_{M_i}) p(\boldsymbol{\theta}_{M_i}|M_i) d\boldsymbol{\theta}_{M_i}}_{=p(\mathcal{D}|M_i)}$$

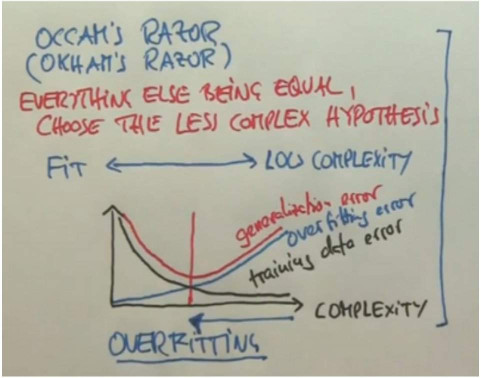
Bayesian Model Averaging

- Place a prior $p(M)$ on the class of models
- Instead of selecting the "best" model, integrate out the corresponding model parameters $\boldsymbol{\theta}_M$ and average over all models $M_i, i = 1, \ldots, L$

$$p(\mathcal{D}) = \sum_{i=1}^{L} p(M_i) \underbrace{\int p(\mathcal{D}|\boldsymbol{\theta}_{M_i}) p(\boldsymbol{\theta}_{M_i}|M_i) d\boldsymbol{\theta}_{M_i}}_{=p(\mathcal{D}|M_i)}$$

- Computationally expensive
- Integral often intractable (still...)

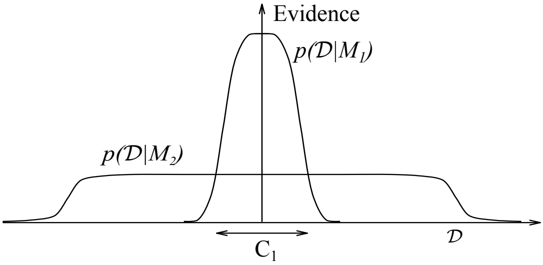
Occam's Razor



From `crowfly.net`

- Favor simpler models over complicated ones
- Very expressive models may be a less probable choice for modeling a given dataset

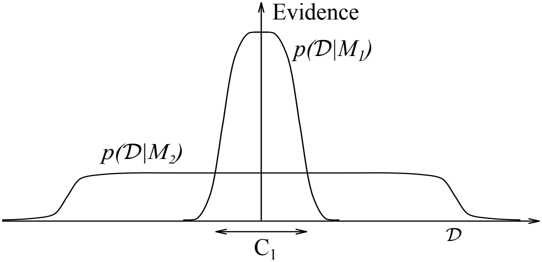
Occam's Razor (2)



From MacKay, ITILA (2003)

- Bayes' theorem rewards models in proportion to how much they predicted the data that occurred ⏩ Marginal likelihood (assuming a uniform prior over models)
- Simple model can predict only a small number of datasets

Occam's Razor (2)

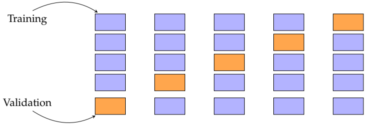


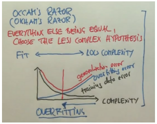
From MacKay, ITILA (2003)

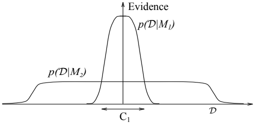
- Bayes' theorem rewards models in proportion to how much they predicted the data that occurred ▶▶ Marginal likelihood (assuming a uniform prior over models)
- Simple model can predict only a small number of datasets

▶▶ **Marginal likelihood automatically embodies Occam's razor**

Summary







- Objective: Achieve good generalization performance
- Assess generalization performance if only training data is available
  - Cross validation
  - Information criteria
- Occam's razor: choose the simplest model that explains the data
- Bayesian model selection and importance of the marginal likelihood

References I

[1] H. Akaike. A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer-Verlag, 2006.

[3] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, The Edinburgh Building, Cambridge CB2 2RU, UK, 2003.

[4] C. E. Rasmussen and Z. Ghahramani. Occam's Razor. In *Advances in Neural Information Processing Systems*, pages 294–300. The MIT Press, 2001.

[5] G. E. Schwarz. Estimating the Dimension of a Model. *Annals of Statistics*, 6(2):461–464, 1978.