STA 414/2104: Machine Learning

Russ Salakhutdinov

Department of Computer Science
Department of Statistics
rsalakhu@cs.toronto.edu
http://www.cs.toronto.edu/~rsalakhu/

Lecture 4
Recap

• In our previous classes, we looked at:
  - Statistical Decision Theory
  - Linear Regression Models
  - Linear Basis Function Models
  - Regularized Linear Regression Models
  - Bias-Variance Decomposition

• We will now look at the Bayesian framework and Bayesian Linear Regression Models.
Bayesian Approach

• We formulate our knowledge about the world probabilistically:
  - We define the model that expresses our knowledge qualitatively (e.g. independence assumptions, forms of distributions).
  - Our model will have some unknown parameters.
  - We capture our assumptions, or prior beliefs, about unknown parameters (e.g. range of plausible values) by specifying the prior distribution over those parameters before seeing the data.
• We observe the data.
• We compute the posterior probability distribution for the parameters, given observed data.
• We use this posterior distribution to:
  - Make predictions by averaging over the posterior distribution
  - Examine/Account for uncertainly in the parameter values.
  - Make decisions by minimizing expected posterior loss.

(See Radford Neal’s NIPS tutorial on “Bayesian Methods for Machine Learning”)
Posterior Distribution

• The posterior distribution for the model parameters can be found by combining the prior with the likelihood for the parameters given the data.

• This is accomplished using Bayes’ Rule:

\[
P(\text{parameters} \mid \text{data}) = \frac{P(\text{data} \mid \text{parameters})P(\text{parameters})}{P(\text{data})}
\]

\[
p(w \mid D) = \frac{p(D \mid w)P(w)}{P(D)}
\]

This integral can be high-dimensional and is often difficult to compute.
The Rules of Probability

Sum Rule:

\[ p(X) = \sum_{Y} p(X, Y) \]

Product Rule:

\[ p(X, Y) = p(Y|X)p(X) \]
Predictive Distribution

• We can also state Bayes’ rule in words:

\[ \text{posterior} \propto \text{likelihood} \times \text{prior}. \]

• We can make predictions for a new data point \( x^* \), given the training dataset by integrating over the posterior distribution:

\[
p(x^* | \mathcal{D}) = \int p(x^* | w, \mathcal{D}) p(w | \mathcal{D}) \, dw = \mathbb{E}_{P(w | \mathcal{D})} \left[ p(x^* | w, \mathcal{D}) \right],
\]

which is sometimes called **predictive distribution**.

• Note that computing predictive distribution requires knowledge of the posterior distribution:

\[
p(w | \mathcal{D}) = \frac{p(\mathcal{D} | w) P(w)}{P(\mathcal{D})}, \quad \text{where} \quad P(\mathcal{D}) = \int p(\mathcal{D} | w) P(w) \, dw
\]

which is usually **intractable**.
Modeling Challenges

• The first challenge is in specifying suitable model and suitable prior distributions. This can be challenging particularly when dealing with high-dimensional problems we see in machine learning.

  - A suitable model should admit all the possibilities that are thought to be at all likely.
  - A suitable prior should avoid giving zero or very small probabilities to possible events, but should also avoid spreading out the probability over all possibilities.

• We may need to properly model dependencies between parameters in order to avoid having a prior that is too spread out.

• One strategy is to introduce latent variables into the model and hyperparameters into the prior.

• Both of these represent the ways of modeling dependencies in a tractable way.
Computational Challenges

The other big challenge is computing the posterior distribution. There are several main approaches:

• **Analytical integration**: If we use “conjugate” priors, the posterior distribution can be computed analytically. Only works for simple models and is usually too much to hope for.

• **Gaussian (Laplace) approximation**: Approximate the posterior distribution with a Gaussian. Works well when there is a lot of data compared to the model complexity (as posterior is close to Gaussian).

• **Monte Carlo integration**: Once we have a sample from the posterior distribution, we can do many things. The dominant current approach is Markov Chain Monte Carlo (MCMC) -- simulate a Markov chain that converges to the posterior distribution. It can be applied to a wide variety of problems.

• **Variational approximation**: A cleverer way to approximate the posterior. It often works much faster compared to MCMC. But often not as general as MCMC.
Bayesian Linear Regression

- Given observed inputs $X = \{x_1, x_2, ..., x_N\}$, and corresponding target values $t = [t_1, t_2, ..., t_N]^T$, we can write down the likelihood function:

$$p(t|X, w, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|w^T \phi(x_n), \beta^{-1}),$$

where $\phi(x) = (\phi_0(x), \phi_1(x), ..., \phi_{M-1}(x))^T$ represent our basis functions.

- The corresponding conjugate prior is given by a Gaussian distribution:

$$p(w) = \mathcal{N}(w|m_0, S_0).$$

- As both the likelihood and the prior terms are Gaussians, the posterior distribution will also be Gaussian.

- If the posterior distributions $p(\theta|x)$ are in the same family as the prior probability distribution $p(\theta)$, the prior and posterior are then called conjugate distributions, and the prior is called a conjugate prior for the likelihood.
Bayesian Linear Regression

- Combining the prior together with the likelihood term:

\[ p(w|t, X, w, \beta) \propto \prod_{n=1}^{N} \mathcal{N}(t_n|w^T \phi(x_n), \beta^{-1}) \mathcal{N}(w|m_0, S_0). \]

- The posterior (with a bit of manipulation) takes the following Gaussian form:

\[ p(w|t) = \mathcal{N}(w|m_N, S_N) \]

where

\[ m_N = S_N \left( S_0^{-1} m_0 + \beta \Phi^T t \right) \]
\[ S_N^{-1} = S_0^{-1} + \beta \Phi^T \Phi. \]

- The posterior mean can be expressed in terms of the least-squares estimator and the prior mean:

\[ m_N = S_N \left( S_0^{-1} m_0 + \beta \Phi^T \Phi w_{ML} \right). \]
\[ w_{ML} = (\Phi^T \Phi)^{-1} \Phi^T t. \]

- As we increase our prior precision (decrease prior variance), we place greater weight on the prior mean relative to the data.
Bayesian Linear Regression

• Consider a zero mean isotropic Gaussian prior, which is governed by a single precision parameter $\alpha$:

$$p(w) = \mathcal{N}(w|0, \alpha^{-1}I)$$

for which the posterior is Gaussian with:

$$m_N = \beta S_N \Phi^T t$$

$$S_N^{-1} = \alpha I + \beta \Phi^T \Phi.$$  

• If we consider an infinitely broad prior, $\alpha \rightarrow 0$, the mean $m_N$ of the posterior distribution reduces to maximum likelihood value $w_{ML}$.

$$w_{ML} = (\Phi^T \Phi)^{-1} \Phi^T t.$$  

• The log of the posterior distribution is given by the sum of the log-likelihood and the log of the prior:

$$\ln p(w|\mathcal{D}) = -\frac{\beta}{2} \sum_{n=1}^{N} (t_n - w^T \phi(x_n))^2 - \frac{\alpha}{2} w^T w + \text{const.}$$

• Maximizing this posterior with respect to $w$ is equivalent to minimizing the sum-of-squares error function with a quadratic regulation term $\lambda = \alpha / \beta$. 
Bayesian Linear Regression

• Consider a linear model of the form: \( y(x, w) = w_0 + w_1 x \).
• The training data is generated from the function \( f(x, a) = a_0 + a_1 x \) with \( a_0 = 0.3; a_1 = 0.5 \), by first choosing \( x_n \) uniformly from \([-1;1]\), evaluating \( f(x, a) \), and adding a small Gaussian noise.

• **Goal:** recover the values of \( a_0, a_1 \) from such data.

0 data points are observed:
Bayesian Linear Regression

0 data points are observed:

Prior

Data Space

1 data point is observed:

Likelihood

Posterior

Data Space
Bayesian Linear Regression

0 data points are observed.

1 data point is observed.

2 data points are observed.

20 data points are observed.
Predictive Distribution

- We can make predictions for a new input vector \( \mathbf{x} \) by integrating over the posterior distribution:

\[
p(t|\mathbf{t}, \mathbf{x}, \mathbf{X}, \alpha, \beta) = \int p(t|\mathbf{x}, \mathbf{w}, \beta)p(\mathbf{w}|\mathbf{t}, \mathbf{X}, \alpha, \beta) d\mathbf{w}
\]

\[
= \mathcal{N}(t|\mathbf{m}_N^T \phi(\mathbf{x}), \sigma_N^2(\mathbf{x}))
\]

where

\[
\sigma_N^2(\mathbf{x}) = \frac{1}{\beta} + \phi(\mathbf{x})^T \mathbf{S}_N \phi(\mathbf{x}).
\]

- In the limit, as \( N \to \infty \), the second term goes to zero.
- The variance of the predictive distribution arises only from the additive noise governed by parameter \( \beta \).

Noise in the target values

Uncertainly associated with parameter values.
Predictive Distribution: Bayes vs. ML

Predictive distribution based on maximum likelihood estimates

\[ p(t|x, w_{ML}, \beta_{ML}) = \mathcal{N}(t|y(x, w_{ML}), \beta_{ML}^{-1}) \]

Bayesian predictive distribution

\[ p(t|x, t, X) = \mathcal{N}(t|m_N^T\phi(x), \sigma_N^2(x)) \]
Predictive Distribution

Sinusoidal dataset, 9 Gaussian basis functions.

Predictive distribution

Samples from the posterior
Predictive Distribution

Sinusoidal dataset, 9 Gaussian basis functions.
Gamma-Gaussian Conjugate Prior

• So far we have assumed that the noise parameter $\beta$ is known.

• If both $w$ and $\beta$ are treated as unknown, then we can introduce a conjugate prior distribution that will be given by the Gaussian-Gamma distribution:

$$p(w, \beta) = \mathcal{N}(w|m_0, \beta^{-1}S_0)\text{Gam}(\beta|a_0, b_0),$$

where the Gamma distribution is given by:

$$\text{Gam}(\beta|a, b) = \frac{1}{\Gamma(a)} b^a \beta^{a-1} \exp(-b\beta), \quad \Gamma(a) = \int_0^\infty u^{a-1} e^{-u} du.$$ 

• The posterior distribution takes the same functional form as the prior:

$$p(t|X, w, \beta) = \mathcal{N}(w|m_N, \beta^{-1}S_N)\text{Gam}(\beta|a_N, b_N).$$
Equivalent Kernel

• The predictive mean can be written as:

\[
y(x, m_N) = m_N^T \phi(x) = \beta \phi(x)^T S_N \Phi^T t
\]

\[
= \sum_{n=1}^{N} \beta \phi(x)^T S_N \phi(x_n) t_n
\]

\[
= \sum_{n=1}^{N} \phi(x, x_n) t_n.
\]

\[
\begin{align*}
m_N &= \beta S_N \Phi^T t \\
S_N^{-1} &= \alpha I + \beta \Phi^T \Phi.
\end{align*}
\]

The mean of the predictive distribution at a time \(x\) can be written as a linear combination of the training set target values.

Such regression functions are called linear smoothers.
Equivalent Kernel

- The weight of \( t_n \) depends on distance between \( x \) and \( x_n \); nearby \( x_n \) carry more weight.

- We can avoid the use of basis functions and define the kernel function directly, leading to Gaussian Processes.

- The kernel as a covariance function:

\[
\text{cov}[y(x), y(x')] = \text{cov}[\phi(x)^T w, w^T \phi(x')]
\]
\[
= \phi(x)^T S_N \phi(x') = \beta^{-1} k(x, x').
\]
Other Kernels

- Examples of kernels $k(x,x')$ for $x=0$, plotted as a function corresponding to $x'$.

- Note that these are localized functions of $x'$.
Bayesian Model Comparison

• The Bayesian view of model comparison involves the use of probabilities to represent uncertainty in the choice of the model.

• We would like to compare a set of L models \( \{ \mathcal{M}_i \} \), where \( i = 1, 2, \ldots, L \), using a training set \( D \).

• We specify the prior distribution over the different models \( p(\mathcal{M}_i) \).

• Given a training set \( D \), we evaluate the posterior:

\[
p(\mathcal{M}_i|D) \propto p(\mathcal{M}_i)p(D|\mathcal{M}_i).
\]

• For simplicity, we will assume that all model are a-priori equal.

• The model evidence expresses the preference shown by the data for different models.

• The ratio of two model evidences for two models is known as Bayes factor:

\[
\frac{p(D|\mathcal{M}_i)}{p(D|\mathcal{M}_j)}
\]
Bayesian Model Comparison

• Once we compute the posterior $p(M_i|D)$, we can compute the predictive (mixture) distribution:

$$p(t|x, D) = \sum_{i=1}^{L} p(t|x, M_i, D)p(M_i|D).$$

• The overall predictive distribution is obtained by averaging the predictive distributions of individual models, weighted by the posterior probabilities.

• For example, if we have two models, and one predicts a narrow distribution around $t=a$ while the other predicts a narrow distribution around $t=b$, then the overall predictions will be bimodal:

• A simpler approximation, known as model selection, is to use the model with the highest evidence.
Bayesian Model Comparison

• Remember, the posterior is given by

\[ p(M_i | D) \propto p(M_i) p(D | M_i). \]

For a model governed by a set of parameters \( w \), the model evidence can be computed as follows:

\[ p(D | M_i) = \int p(D | w, M_i) p(w | M_i) \, dw. \]

• Observe that the evidence is the normalizing term that appears in the denominator in Bayes’ rule:

\[ p(w | D, M_i) = \frac{p(D | w, M_i) p(w | M_i)}{p(D | M_i)} \]

• The model evidence is also often called marginal likelihood.
Bayesian Model Comparison

• We next get some insight into the model evidence by making simple approximations.

• For a given model with a single parameter $w$, consider approximations:
  
  - Assume that the posterior is picked around the most probable value $w_{\text{MAP}}$, with width $\Delta w_{\text{posterior}}$
  
  - Assume that the prior is flat with width $\Delta w_{\text{prior}}$

$$p(\mathcal{D}) = \int p(\mathcal{D}|w)p(w)\,dw$$

$$\approx p(\mathcal{D}|w_{\text{MAP}}) \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}}$$
Bayesian Model Comparison

• Taking the logarithms, we obtain:

\[
\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|w_{\text{MAP}}) + \ln \left( \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}} \right). 
\]

Negative

• With M parameters, all assumed to have the same \( \Delta w_{\text{posterior}} / \Delta w_{\text{prior}} \) ratio:

\[
\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|w_{\text{MAP}}) + M \ln \left( \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}} \right). 
\]

Negative and linear in M.

• As we increase the complexity of the model (increase the number of adaptive parameters M), the first term will increase, whereas the second term will decrease due to the dependence on M.
• The optimal model complexity: trade-off between these two competing terms.
Bayesian Model Comparison

- The simple model cannot fit the data well, whereas the more complex model spreads its predictive probability and so assigns relatively small probability to any one of them.

- For the particular observed dataset $D_0$, the model $M_2$ with intermediate complexity has the largest evidence.

- The marginal likelihood is very sensitive to the prior used!
- Computing the marginal likelihood makes sense only if you are certain about the choice of the prior.
Evidence Approximation

• In the fully Bayesian approach, we would also specify a prior distribution over the hyperparameters $p(\alpha, \beta)$.

• The fully Bayesian predictive distribution is then given by marginalizing over model parameters as well as hyperparameters:

$$
p(t^*|\mathbf{x}^*, \mathcal{D}) = \int \int \int p(t^*|\mathbf{x}^*, \mathbf{w}, \beta)p(\mathbf{w}|\mathcal{D}, \alpha, \beta)p(\alpha, \beta|\mathcal{D}) d\mathbf{w} d\alpha d\beta.
$$

• However, this integral is intractable (even when everything is Gaussian). Need to approximate.

• Note: the fully Bayesian approach is to integrate over the posterior distribution for $\{\alpha, \beta, \mathbf{w}\}$. This can be done by MCMC, which we will consider later. For now, we will use evidence approximation: much faster.
Evidence Approximation

• The fully Bayesian predictive distribution is given by:

\[ p(t^*|x^*, D) = \int \int \int p(t^*|x^*, w, \beta)p(w|D, \alpha, \beta)p(\alpha, \beta|D)dwd\alpha d\beta. \]

• If we assume that the posterior over hyperparameters \( \alpha \) and \( \beta \) is sharply picked, we can approximate:

\[ p(t^*|x^*, D) \approx p(t^*|x^*D, \hat{\alpha}, \hat{\beta}) = \int p(t^*|x^*, D, \hat{\beta})p(w|D, \hat{\alpha}, \hat{\beta})dw. \]

where \( (\hat{\alpha}, \hat{\beta}) \) is the mode of the posterior \( p(\alpha, \beta|D) \).

• So we integrate out parameters but maximize over hyperparameters.

• This is known as empirical Bayes, Type II Maximum Likelihood, Evidence Approximation.
Evidence Approximation

• From Bayes’ rule we obtain:

\[ p(\alpha, \beta|t, X) \propto p(t|X, \alpha, \beta)p(\alpha, \beta). \]

• If we assume that the prior over hyperparameters \( p(\alpha, \beta) \) is flat, we get:

\[ p(\alpha, \beta|t, X) \propto p(t|X, \alpha, \beta). \]

• The values \( (\hat{\alpha}, \hat{\beta}) \) are obtained by maximizing the marginal likelihood \( p(t|X, \alpha, \beta) \).

• This will allow us to determine the values of these hyperparameters from the training data.

• Recall that the ratio \( \alpha/\beta \) is analogous to the regularization parameter.
Evidence Approximation

• The marginal likelihood is obtained by integrating out parameters:

\[ p(t|X, \alpha, \beta) = \int p(t|X, w, \beta)p(w|\alpha)dw. \]

\[ m_N = \beta S_N \Phi^T t \]
\[ S_N^{-1} = \alpha I + \beta \Phi^T \Phi. \]

• We can write the evidence function in the form:

\[ p(t|X, \alpha, \beta) = \left( \frac{\beta}{2\pi} \right)^{N/2} \left( \frac{\alpha}{2\pi} \right)^{M/2} \int \exp \left( -E(w) \right) dw, \]

where

\[ E(w) = \beta E_D(w) + \alpha E_W(w) = \frac{\beta}{2} ||t - \Phi w||^2 + \frac{\alpha}{2} w^T w. \]

• Using standard results for the Gaussian distribution, we obtain:

\[ \ln p(t|\alpha, \beta) = \frac{M}{2} \ln \alpha + \frac{N}{2} \ln \beta - E(m_N) + \frac{1}{2} \ln |S_N| - \frac{N}{2} \ln(2\pi). \]
Some Fits to the Data

For $M=9$, we have fitted the training data perfectly.
Evidence Approximation

Using sinusoidal data, $M^{th}$ degree polynomial.

The evidence favours the model with $M=3$. 
Maximizing the Evidence

• Remember:

\[
\ln p(t|\alpha, \beta) = \frac{M}{2} \ln \alpha + \frac{N}{2} \ln \beta - E(m_N) + \frac{1}{2} \ln |S_N| - \frac{N}{2} \ln(2\pi).
\]

• To maximize the evidence \( p(t|X, \alpha, \beta) \) with respect to \( \alpha \) and \( \beta \), define the following eigenvector equation:

\[
\left( \beta \Phi^T \Phi \right) u_i = \lambda_i u_i.
\]

• Therefore the matrix:

\[
A = S_N^{-1} = \alpha I + \beta \Phi^T \Phi
\]

has eigenvalues \( \alpha + \lambda_i \).

• The derivative:

\[
\frac{d}{d\alpha} \ln |A| = \frac{d}{d\alpha} \ln \prod_i (\alpha + \lambda_i) = \frac{d}{d\alpha} \sum_i \ln(\alpha + \lambda_i) = \sum_i \frac{1}{\alpha + \lambda_i}.
\]
Maximizing the Evidence

- Remember:

\[ \ln p(t|\alpha, \beta) = \frac{M}{2} \ln \alpha + \frac{N}{2} \ln \beta - E(m_N) + \frac{1}{2} \ln |S_N| - \frac{N}{2} \ln(2\pi). \]

where

\[ E(m_N) = \frac{\beta}{2} ||t - \Phi m_N||^2 + \frac{\alpha}{2} m_N^T m_N. \]

- Differentiating \( \ln p(t|\alpha, \beta) \), the stationary points with respect to \( \alpha \) satisfy:

\[
\frac{M}{2\alpha} - \frac{1}{2} m_N^T m_N - \frac{1}{2} \sum_i \frac{1}{\alpha + \lambda_i} = 0.
\]

\[
\alpha m_N^T m_N = M - \alpha \sum_i \frac{1}{\alpha + \lambda_i} = \gamma,
\]

where the quantity \( \gamma \), effective number of parameters, can be defined as:

\[ \gamma = \sum_i \frac{\lambda_i}{\lambda_i + \alpha}. \]
Maximizing the Evidence

• The stationary points with respect to $\alpha$ satisfy:

$$\alpha m_N^T m_N = M - \alpha \sum_i \frac{1}{\alpha + \lambda_i} = \gamma, $$

where the quantity $\gamma$, effective number of parameters, is defined as:

$$\gamma = \sum_i \frac{\lambda_i}{\lambda_i + \alpha}. $$

• Iterate until convergence:

$$\alpha = \frac{\gamma}{m_N^T m}; \quad \gamma = \sum_i \frac{\lambda_i}{\lambda_i + \alpha}; \quad m_N = \beta S_N \Phi^T t \quad S_N^{-1} = \alpha I + \beta \Phi^T \Phi. $$

• Similarly:

$$\frac{1}{\beta} = \frac{1}{N - \gamma} \sum_{n=1}^N \left\{ t_n - m_N^T \phi(x_n) \right\}^2$$
Effective Number of Parameters

- Consider the contours of the likelihood function and the prior.

  • The eigenvalue $\lambda_i$ measures the curvature of the log-likelihood function.
  
  • The quantity $\gamma$ will lie $0 \leq \gamma \leq M$.

  • For $\lambda_i \gg \alpha$, the corresponding parameter $w_i$ will be close to its maximum likelihood. The ratio $\frac{\lambda_i}{\lambda_i + \alpha}$ will be close to one.

  • Such parameters are called well determined, as their values are highly constrained by the data.

  • For $\lambda_i \ll \alpha$, the corresponding parameters will be close to zero (pulled by the prior), as will the ratio $\frac{\lambda_i}{\lambda_i + \alpha}$.

  • We see that $\gamma$ measures the effective total number of well determined parameters.
Quick Approximation

- In the limit $N \gg M$, $\gamma = M$, and we consider to use the easy to compute approximations:

$$\alpha = \frac{M}{m_N^T m_N}$$

$$\frac{1}{\beta} = \frac{1}{N} \sum_{n=1}^{N} \left\{ t_n - m_N^T \phi(x_n) \right\}^2.$$
Limitations

• M basis function along each dimension of a D-dimensional input space requires $M^D$ basis functions: the curse of dimensionality.

• Fortunately, we can get away with fewer basis functions, by choosing these using the training data (e.g. adaptive basis functions), which we will see later.

• Second, the data vectors typically lie close to a nonlinear low-dimensional manifold, whose intrinsic dimensionality is smaller than that of the input space.